Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# maxAI 280 Design Studio Software Guide

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

#### **Revision History**

Ver	Change Description	Date	Author	Approver
1.0	Initial release	Sep 28, 2022	Zaid N	
2.0	Changes made in LED Section 7.4, 7.4.6, 7.4.7, 7.4.9, 7.4.10 Digital IO 7.2.2, 7.2.3, 8.2.4.1 Keypad 7.1.6, 7.1.3.3 In each section of the platform service regarding the time out configuration and task priority, added the text "Recommended value" Common sections included from AI430 user manual into the AI280 user manual.	Oct 17, 2022	Zaid N	
3.0	Overall corrections. See attached excel file for details: maxAl280 SDK Manual Corrections	Oct 21 <sup>st</sup> , 2022	Luis Figueroa	Victor Rios

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

#### References

No	Document Name	Ver.	Lotion
01	touchgfx- documentation-4.16.pdf	1.0	https://support.touchgfx.com/4.20/docs/introduction/welcome
02	STM 32 document	1.0	http://www.st.com/en/microcontrollers- microprecessors/stm32-32-bit-arm-cortex-mcus.html
03	Free RTOS	1.0	https://controllerstech.com/free-rtos-tutorial-2-0-with-stm32/
04	J1939	1.0	https://copperhilltech.com/a-brief-introduction-to-the-sae- j1939-protocol/

Table of Contents	

REFERENCES	3
TABLE OF CONTENTS	4
SCOPE OF DOCUMENT	9
ACRONYMS & ABBREVIATIONS	
1 MAXIMATECC SOFTWARE OVERVIEW	
1.1.1 maxAI™ Configurator	11
1.1.2 maxAl™ Design Studio	
1.1.2 maxAr Design stadio 1.1.3 maxAr™ Specialized	
3 SDK SETUP AND INSTALLATION	13
3.1 SDK Development Environment	13
3.1.1 Hardware Requirements	
3.1.2 Software Requirements	
3.2 IDE INSTALLATIONS	
3.3 S/W RELEASE PACKAGE DETAILS	
3.3.1 AI280 Project Structure	
3.3.2 Demo Project File	
3.3.3 Sample Application Project File	
3.3.4 Blank Project File	27
4 SDK APPLICATION DEVELOPMENT PROCEDURE	
4.1 BLANK PROJECT INSTALLATION AND LOADING	29
4.1.1 STM32 Cube IDE Setup	
4.2 Build and Flash Procedure	
4.3 CAN MODULE DEMO	41
4.3.1 Adding new GUI elements in the TouchGFX Screen	41
4.3.2 Edit the DB Variables	44
4.3.3 Configurations	49
4.3.4 Output	50
4.4 LED DEMO	54
4.5 User Task Edit Details	63
5 SDK OVERVIEW	73
5.1 SDK Architecture	73
5.2 SDK INTERFACES	75
5.3 SDK BOOT FLOW	76
6 APPLICATION AND SDK INTERACTION	77
6.1 SDK MODULE DEFAULT CONFIGURATION	77
6.2 RUN TIME CONFIGURATION:	

6.3	DBLayer USER APIS	79
6.4	FUNCTION NAME: GET_DL	-
6.5	Function Name: SET_DL	
7 (0	 DK MODULES	
7 SD	JK MODULES	82
7.1	Keypad module	-
7.1	1.1 Keypad module Enable/Disable	
	1.2 Keypad BackLight ON/OFF	
7.1	1.3 Keypad Time Out Configuration	
7.1	1.4 Keypad task Priority	
	1.5 Keypad Keys Enable/Disable	
	1.6 Keypad Keys read status	
	1.7 Keypad sample Configuration	
7.2	DIGITAL OUTPUT MODULE	
	2.1 Digital Output module Enable/Disable	
	2.2 Digital Output Configuration	
	2.3 Digital Output ON/OFF	
	2.4 Digital Output Time Out Configuration	
	2.5 Digital Output task Priority	
	2.6 Digital Output Sample Configuration	
7.3	Configurable Inputs Module	
7.3	3.1 Configurable Inputs module Enable/Disable	
7.3	3.2 Configurable Inputs task Priority	
7.3	3.3 Configurable Inputs Task Time Out Configuration	
7.3	3.4 Configurable Inputs – Configure the number of samples	
7.3	3.5 Configurable Inputs configuration	
7.3	3.6 Configurable Inputs Default Configuration	
7.4	LED MODULE	
7.4	4.1 LED module Enable/Disable	
7.4	4.2 LED Time Out Configuration	
7.4	4.3 LED task Priority	
7.4	4.4 Maximum LED'S Configuration	
7.4	4.5 Configuring RED LED Enable/Disable	
7.4	4.6 Configuring RED LED State	
7.4	4.7 Configuring RED LED blinking	
7.4	4.8 Configuring AMB LED Enable/Disable	
7.4	4.9 Configuring AMB LED State	
7.4	4.10 Configuring AMB LED blinking	
7.4	4.11 LED Sample Configuration	
7.5	Power Monitor Module	
	5.1 Power Monitor module Enable/Disable	
7.5	5.2 Power Monitor Time Out Configuration	
7.5	5.3 Power Monitor task Priority	
7.5	5.4 Power Monitor Functionality Support	
7.5	5.5 Power Monitor sample configuration	
7.6	BLUETOOTH LOW ENERGY (BLE) MODULE	118

7 ( 1	BLE module Enable/Disable	110
7.6.1 7.6.2	BLE Time Out Configuration	
7.6.3	BLE Monitor task Priority	
7.6.4	BLE module device Name configuration	
-	BLE module device Nume conjugaration BLE module RX/TX	
7.6.5		
7.6.6	BLE sample configuration	
7.7.1	Timer Module Enable/Disable	
7.7.2	Timer Module Time Out Configuration	
7.7.3	Timer Module Task Priority	
7.7.4	Timer Start or Stop	
7.7.5	Timer Mode Configuration	
7.7.6	, ,	
	Timer sample configuration	
7.8	RTC MODULE	
7.8.1	RTC Module Enable/Disable	
7.8.2	RTC Timeout Configuration	
7.8.3	RTC Task Priority	129
7.8.4	RTC Date and Time Configuration	
7.8.5	RTC Time Format	131
7.8.6	RTC Alarm Date and Time	132
7.8.7	RTC Alarm Time Format	135
7.8.8	RTC Sample Configuration	135
7.9	EEPROM Module	136
7.9.1	EPROM Module Enable/Disable	136
7.9.2	EEPROM Time Out Configuration	136
7.9.3	EEPROM Module Task Priority	136
7.9.4	EEPROM Placeholder	137
7.9.5	EEPROM Sample Configuration	139
7.10	WATCH DOG MODULE	141
7.10.	1 Watch dog module Enable/Disable	141
7.10.	-	
7.10.		
7.10.		
7.10.		
7.10.		
7.10.	5 5 ,	
-	Power Mode Module	
7.11.		
7.11.		
7.11.		
7.11.		
7.11.		
7.11.		
7.11.		
	LCD MODULE	
7.12		152

7.12.1	LCD mode module Enable/Disable	
7.12.2	LCD Module Timeout Configuration	
7.12.3	LCD task Priority	
7.12.4	LCD State	
7.12.5	LCD Brightness	
7.12.6	LCD Sample Configurations	
	N MODULE	
7.13.1 7.13.2	CAN Module Configuration Support	
/12012	CAN Enable/Disable	
7.13.3	CAN Module Timeout Configuration	
7.13.4	CAN benderate	
7.13.5	CAN Identifier Confirmations	
7.13.6	CAN Identifier Configurations	
7.13.7	CAN Films Configurations	
7.13.8	CAN Preside Track Delay	
7.13.9	CAN Receive Task Delay	
7.13.10	CAN Channel Modes and States	-
7.13.11	CAN Channel Reset	
7.13.12	CAN module RX/TX	
7.13.13	CAN Sample Configuration.	
	39	
7.14.1	J1939 Module Configuration Support	
7.14.2	J1939 Module Timeout Configuration	
7.14.3	J1939 task Priority	
7.14.4	J1939 Claim Address Enable/Disable	
7.14.5	J1939 CAN Enable/Disable	
7.14.6	J1939 Claim Address	
7.14.7	J1939 CAN Bit Rate	
7.14.8	J1939 Diagnostics Support	
7.14.9	J1939 Dynamic Address Claim	
7.14.10	J1939 Dynamic Address Claim Next Address Configuration	
7.14.11	J1939 Configure Number of PGN's supported	
7.14.12	J1939 PGN and SPN Configuration	
7.14.13	J1939 Diagnostic Message Configuration	
	J1939 Sample Configuration	
	ROUGHPUT MODULE	
7.15.1	Throughput Enable/Disable	
7.15.2	Throughput Absolute Time	
7.15.3	Throughput Percentage Time	
7.15.4	Fhroughput Sample Configuration	187
8 APPLICA	TION DETAILS	
8.1 SAM	IPLE APPLICATION PROJECT DETAILS	
8.1.1	Introduction	188
8.1.2	Home Screen Navigation:	189
	Keypad Module	

	8.1.4	Power Monitor	
	8.1.5	RTC	
	8.1.6	LCD	
	8.1.7	Digital Output	
	8.1.8	Software Timer	
	8.1.9	Configurable Inputs	
	8.1.10	) LED	207
	8.1.1	Power Mode	211
	8.1.12	2 EEPROM	213
	8.1.13	8 Watchdog	215
	8.1.14	BLE	217
	8.1.15	5 CAN	
	8.1.16	5 J1939	
	8.1.17	7 Throughput	232
8	3.2	DETAILS OF DEMO APPLICATION	234
	8.2.1	Difference between Sample Application and Demo Application	234
	8.2.2	Panel Button Functionality	234
	8.2.3	Demo App Screen 1	235
	8.2.4	Demo App Screen 2	237
	8.2.5	Demo App screen 3	240
	8.2.6	Demo App Screen 4	242
9		IOBILE TEST APPLICATION	245
9	BLE IV		
ç	9.1	SCAN SCREEN	245
ç	9.2	Connect Screen	247
ç	9.3	GUI SCREEN	250
ç	9.4	Read/Write DB Variable Screen	252
ç	9.5	Read/Write by Memory Address Screen	255
ç	9.6	GENERIC DATA TO SEND	258
ç	9.7	CLEAR LIST AND STOP TESTING	260

Dept: Software	<b>User Manual Document</b>	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# **Purpose of document**

The purpose of this document is to enable an application developer to write TouchGFX applications for the maxAI280 SDK using the features, modules, interfaces, and possible configurations that is available with the maxAI280 SDK hardware platform.

# Scope of document

The scope of the document is to list all the features and functionalities of the maxAI280 SDK which are of relevance to the Touch GFX Application Developer using the SDK.

Dept: Software	<b>User Manual Document</b>	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

#### Acronyms & Abbreviations

No	Acronyms	Definition
01	SDK	Software development kit
02	IDE	Integrated development Environment
03	DL	Data Layer
05	RTC	Real Time Clock
06	BLE	Bluetooth low energy
08	DB	Data Base
09	API	Application Programming Interface
10	LCD	Liquid crystal Diode
11	CAN	Control area network

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

### **1** Maximatecc Software Overview

#### 1.1.1 maxAI<sup>TM</sup> Configurator

For quick and easy setup, use the Configurator Tool to automatically populate your engine monitoring data with preset options and layouts. No need for complex coding or additional resources.

#### 1.1.2 maxAI<sup>TM</sup> Design Studio

The Design Studio is a Software Development Kit (SDK) that provides a higher level of flexibility and control. You choose the advanced engine monitoring parameters to equip your display with all the information you need to know.

#### 1.1.3 maxAI<sup>TM</sup> Specialized

The Specialized provides you with access to the maximatecc engineering team, who develop a custom interface that meets your specific application needs. The team supports all elements of the engineering and setup process for ease and flexibility.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

## 2 Introduction

The AI280 SDK platform is an embedded software solution for custom applications based on the AI280 hardware only. This platform provides a set of software components to reduce the development effort to create a complete embedded application compliant with all the customer requirements. The SDK solution potentiates the scope of the AI280 platform.

The user can explore all the possibilities to cover the requirements and needs by using the AI280 peripherals and by creating their own custom graphical applications.



The SDK platform has the following benefits:

- Short development time.
- > Pre-established low level driver administration.
- Portable software components.
- > Low technical development skills required.
- > Pre-configured and stable SW architecture.
- Secure custom and private algorithms implementation.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

## **3** SDK Setup and installation

To get started with the AI280 SDK, you will need to setup the right environment. Please follow the procedure described in this section to install the necessary tools required to use the SDK and create a Touch GFX application.

#### 3.1 SDK Development Environment

The AI280 SDK allows TouchGFX Applications to be custom built on the AI280 platform. Please ensure the below hardware and software setup is available.

Host PC	WINDOWS (64-bit OS)
RAM Size	4 GB RAM required minimum
Disk Space	2 GB disk space required minimum
Board with power supply	MAXAI280 kit
Debugger	ST Link V2 in-Circuit debugger with USB cable

#### 3.1.1 Hardware Requirements

#### 3.1.2 Software Requirements

Development IDE	STM32 Cube (1.8.0)
Development IDE	TOUCH GFX (4.18.1)
SW Package	S/W package released with the MAXAI280 kit

#### **3.2 IDE Installations**

To get started with the AI280 SDK, please follow the below links to install the STM32 Cube IDE and the Touch GFX IDE.

- 1) Install the STM32 Cube IDE following the instructions in the document <u>https://www.st.com/resource/en/user\_manual/um2563-stm32cubeide-installation-guide-stmicroelectronics.pdf</u>
- 2) Install the Touch GFX IDE following the instructions listed in the document https://support.touchgfx.com/4.18/docs/introduction/installation

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

3) Once the STM32 Cube IDE is installed to open one of the SDK projects (integration test project, demo project or blank template project) open the Cube IDE and from the File menu select "open project from file system..." option

For reliability within TouchGFX, please ensure the project files are located in a file path directory with no spaces. i.e.,

"C:\Users\91822\Desktop\AI280\Project\_Build\AI280\_Project\_Integration"

<u>F</u> ile	<u>E</u> dit <u>Source</u> Refactor <u>N</u> avigate	Search <u>P</u> roject <u>R</u> u
	New	Alt+Shift+N >
	Open File	
	Open Projects from File System	
	Recent Files	>
	Close Editor	Ctrl+W
	Close All Editors	Ctrl+Shift+W
	Save	Ctrl+S
	Save As	
0	Save All	Ctrl+Shift+S
	Revert	
	Move	
	Rename	F2
8	Refresh	F5
	Convert Line Delimiters To	>
	Print	Ctrl+P
è	Import	
2	Export	
	Properties	Alt+Enter
	Switch Workspace	>
	Restart	
	Exit	

To avoid conflicts only open one SDK project at the time.

Select the option "Directory..." to look for the folder where the SDK project is located.

Dept: Software	t: Software User Manual Document	Rev No.	Rev 3.0
F		Date:	Oct 17, 2022

mport source:		~ Di	rectory	Archive
type filter text			] [	Select All
Folder	Import as		D	eselect All
			0 of 0 selec	ted eady open proje
Close newly imported projects upon	completion			
lse <u>installed project configurators</u> to:				
Search for nested projects				
Search for nested projects Detect and configure project natures				New
Search for nested projects Detect and configure project natures Working sets			~	New Select

After some seconds the option "Finish" will be enable and need to be clicked to finish the process.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
T		Date:	Oct 17, 2022

This wizard ana	yzes the content of your folder o	r archive file to find projects and import them	in the IDE.	
Import source:	C:\Users\91822\Desktop\Al280	\Project_Build\AI280_Project_Integration	<ul> <li>✓ Directory</li> </ul>	Archive
type filter text				Select All
Folder		Import as		eselect All
Jse <u>installed pr</u> ✓ Search for n	imported projects upon completi oject configurators to: ested projects configure project natures	ion	Hide alr	eady open projec
∠ Detect and of	configure project natures			
	oningure project natures		-	
Working sets	t to working sets		[	New
Working sets	t to working sets			New Select
Working sets	t to working sets		Show other specia	Select

Once the process to import the project is done the files will be available in the left section of the IDE.

Dept: Software		Rev 3.0	
1		Date:	Oct 17, 2022

Pro	oject Explorer ⊠		8
. DE	AI280_Project_Integration		
>	😽 Binaries		
>	🔊 Includes		
>	😕 Core		
>	Drivers		
>	Middlewares		
>	TouchGFX		
>	🗁 Debug		
>	🗁 Release		
	Al280_Project_Integration.ic	DC	
	Al280_Project_Integration D	ebug.launch	
	STM32H743IITX_FLASH_DE	BUG.Id	
	STM32H743IITX_FLASH_RE	LEASE.Id	
	STM32H743IITX_RAM.Id		

4) Once the SDK project is imported an external loader need to be copied into the installation folder.

Inside the AI280 project in Tools folder the external loader is located.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
<b>k</b>		Date:	Oct 17, 2022

Type: Connect under re	tartup 🦻 Source 🗖 Comr eset 🗸 🗸		
Device settings			
Debug in low power me	odes: Enabl	e	~
suspend watchdog cou	inters while halted: Enabl	e	~
Serial Wire Viewer (SWV Enable	n	RTOS Kernel Awareness Enable RTOS Proxy	
Core Clock (MHz):	16.0	Driver settings	
Limit SWO clock		Driver: ThreadX	
taximum SWO clock (	(Hz) auto detect	Fort: cortex_m0	
ort number.	61235	Port number. 60000	
Misc			
∠ Verify flash downloa ∠ Enable live expression			
Log to file:	C:\Users\91822\Deskto	p\Al280\Project_Build\Al280_Project_Integration\Debug\st-link_gdbserver_log.txt	Browse
External Loader:	maxAl_qspi_loader_stm	i32h743, 0x90000000, NOR_FLASH, maxAI_qspi_loader.stldr v	Scan 🗌 Initialize
Shared ST-LINK			
Max halt timeout(s):	2		

The maxAI\_qspi\_loader. stldr need to be copied in the following path:

 $\label{eq:c:STSTM32CubelDE_1.8.0\STM32CubelDE\plugins\com.st.stm32cube.ide.mcu.externaltools.cubeprogrammer.win32_2.0.100.202110141430\tools\bin\ExternalLoader$ 

#### 3.3 S/W Release Package Details

The MAXAI280 SDK kit comes with the below S/W release package. It has 3 project files released.

- 1) Demo Project File
- 2) Application Project File
- 3) Blank Project File

#### 3.3.1 AI280 Project Structure

The MAXAI280 project files are integrated source code which include the TouchGFX application integrated with the AI280 SDK. These applications leverage the hardware capabilities of the AI280 platform via the SDK interface.

In this section we will describe to you the variations in the three project files released with the MAXAI280 SDK which will enable you to write full-fledged applications using the AI280 SDK.

#### **3.3.2 Demo Project File**

The Demo Project File is a fully graphical pre-built project file which leverages all the functionality of the AI280 SDK. This application is an integrated example which communicates with different modules in the SDK in a single UI screen. This project can be used as a reference for all users who are working on creating integrated applications for their specific needs. This project has 5 UI screen and the details of how-to setup and test the same are described in section 7.2. The below image shows the folder structure of the AI280 demo project file. The Main folders of interest are

- a. Core
- b. Drivers
- c. Touch GFX

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

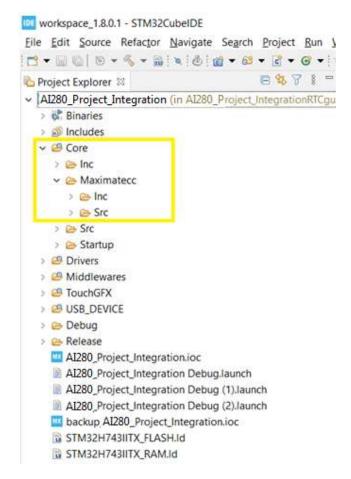
workspace\_1.8.0.1 - STM32CubeIDE

Project Explorer 😂	E 🛠 🎖 🕴 🗖 E
Al280_Project_Integration (in Al	280_Project_IntegrationRTCgui22
> 🖑 Binaries	
> 🞒 Includes	
- 🖾 Core	
> 🗁 Inc	
> 🕞 Maximatecc	
> 🗁 Src	
2 G Startup	
🛩 😂 Drivers	
> 🕞 CMSIS	
Maximatecc	
STM32H7xx_HAL_Driver	
> 😕 Middlewares	
V 🖉 TouchGFX	
> 🗁 App	
assets	
> 🔁 build	
Config	
> 🕞 generated	
> 🗁 gui	
> > src	
> > simulator	
> 🗁 target	
AI280_Project_Integration	backup.touchofx
AI280_Project_Integration	
application.config	
ApplicationTemplate.tout	:hgfx.part
> 🥴 USB_DEVICE	
> 🗁 Debug	
> 🗁 Release	
AI280_Project_Integration.io	
AI280_Project_Integration De	ebug.launch
AI280_Project_Integration De	ebug (1).launch
AI280_Project_Integration De	ebug (2).launch
backup_AI280_Project_Integr	ation.ioc

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

#### 3.3.2.1 Core Directory

The core directory includes the files which form the core of the SDK architecture which include the platform service files for all the modules. (Platformservice.h and Platformservice.c) They are located under the Core\Maximatecc\Inc and Core\Maximatecc\Src directories.



#### **3.3.2.2** Driver Directory

The Driver files for all the modules in the MAXAI280 are located under the folder structure Drivers\Maximatecc\inc and Drivers\Maximatecc\src.

Dept: Software	Software User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

workspace\_1.80.1 - STM32CubeIDE File Edit Source Refactor Navigate Search Project Run Wind 🗂 • 🔛 🐚 | 🖲 • 🐔 • 🚔 🖉 🙆 🔞 • 🚳 • 🗗 • 🚱 • 🕸 • 8\$78 - 0 Project Explorer 😂 AI280\_Project\_Integration (in AI280\_Project\_IntegrationRTCgui22) > 🗟 Binaries > D Includes > 😂 Core > CMSIS Maximatecc > 🕞 inc > 😂 SIC > > STM32H7xx\_HAL\_Driver > 😕 Middlewares > 🐸 TouchGFX USB\_DEVICE > 🗁 Debug > 🕞 Release AI280\_Project\_Integration.ioc AI280\_Project\_Integration Debug.launch AI280 Project Integration Debug (1).launch AI280\_Project\_Integration Debug (2).launch backup\_AI280\_Project\_Integration.ioc STM32H743IITX\_FLASH.Id STM32H743IITX\_RAM.Id

#### 3.3.2.3 TouchGFX Directory

The GUI files for Demo purpose and understanding are available under the folder structure TouchGFX\gui\src. The screens that are available as demo are scr\_io\_screen,scr\_main\_screen and scr\_timer\_screen.

As a TouchGFX developer any GUI code that is developed by you would go into this directory.

Dept: Software	<b>User Manual Document</b>	Rev No.	<b>Rev 3.0</b>
		Date:	Oct 17, 2022

```
workspace_1.8.0.1 - STM32CubeIDE
Eile Edit Source Refactor Navigate Search Project Run Winde
🗂 + 🔄 🐚 🖲 + % + 🙀 🔌 🙆 + 🚳 + 🙆 + 🞯 + 🕸 +
                                        E $ 7 8 - 0
Project Explorer 🖾
AI280_Project_Integration (in AI280_Project_IntegrationRTCgui22)
   > 🐇 Binaries
   > 🔊 Includes
  > 🥴 Core
  > 😕 Drivers
   > 🥴 Middlewares
  > 🕞 App
    > 🗁 assets
     > 🕞 build
    Config
     > 😂 generated
     v 🕞 gui
       > 🕞 include
       V B STC
         > > common
         > 🕞 model
         > 🗁 scr_io_screen
         > 🕞 scr_main_screen
         > 🗁 scr_timer_screen
     > 🕞 simulator
     > 😂 target
       AI280_Project_Integration_backup.touchgfx
       AI280_Project_Integration.touchgfx
       application.config
       ApplicationTemplate.touchgfx.part
   > 🥴 USB_DEVICE
   > 🕞 Debug
   > 🕞 Release
```

Date: Oct 17, 2022	Dept: SoftwareUser Manual DocumentRev No.Rev 3	Rev 3.0	

#### 3.3.3 Sample Application Project File

The Application Project File is a semi graphical pre-built project file which details each module which is available in the AI280 hardware. References for all the functionality of the AI280 SDK can be found in the application project file. This sample application has standalone screens for each and every module in the SDK and elaborates in detail the possible ways you can interact with each module in the SDK. This project can be used as a reference for all users to understand in detail the individual modules of the SDK and get sample reference of how to use the various functionalities in the individual modules. The below image shows the folder structure of the AI280 sample application project file. The Main folders of interest are

- a. Core: Similar to the Demo project
- b. Drivers: Similar to the demo project.
- c. Touch GFX

	avigate Search Project Run Win
3 • 🖩 🕼   🛛 • % • 🖷   •	<b>1</b> ⊕ 1 ∰ ▼ 63 ▼ 63 ▼ 63 ▼ 1 ∰
Project Explorer 😫	■ 🎋 🎖 🕴 🗖 🗖
🗸 🚾 .AI280_Project_Integration (	in AI280_Project_IntegrationRTCgui22
> 🗱 Binaries	
> 🔊 Includes	
- 🖾 Core	
> 🗁 Inc	
> 😂 Maximatecc	
> 🗁 Src	
Startup	2
✓ 🥴 Drivers	
> 😁 CMSIS	
🖌 🗁 Maximatecc	
> 🗁 STM32H7xx_HAL_Driv	ver
> Biddlewares	
V 😬 TouchGFX	
> 🗁 App	
> 🗁 assets	
> 🗁 build	
> 😁 config	
> 😂 generated	
👻 🗁 gui	
> 😂 include	
> 🗁 src	
> 🗁 simulator	
> 🗁 target	
AI280_Project_Integra	ition_backup.touchgfx
AI280_Project_Integra	ition.touchgfx
application.config	
ApplicationTemplate	.touchgfx.part
> 🥴 USB_DEVICE	
> 🗁 Debug	
> 😂 Release	
AI280_Project_Integratio	n.ioc
AI280_Project_Integratio	n Debug.launch
AI280_Project_Integratio	n Debug (1).launch
AI280_Project_Integratio	n Debug (2).launch
Backup_ AI280_Project_In	itegration.ioc

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

#### 3.3.3.1 TouchGFX Directory

The GUI files for application purpose and understanding are available under the folder structure TouchGFX\gui\src. The screens for all the possible user applications are available under mentioned folder as shown in the below diagram.



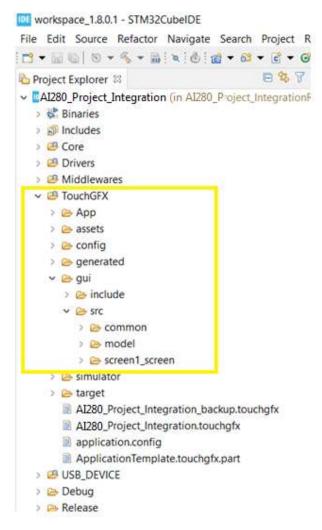
Dept: Software	User Manual Document	Rev No.	Rev 3.0
· · · · · · · · · · · · · · · · · · ·		Date:	Oct 17, 2022

#### 3.3.4 Blank Project File

The Blank Project File template is provided as a convenience for the end user to begin the firmware development. The Blank Project file can be unzipped to the desired location (folder) and renamed to a name as per users' choice.

#### 3.3.4.1 TouchGFX Directory

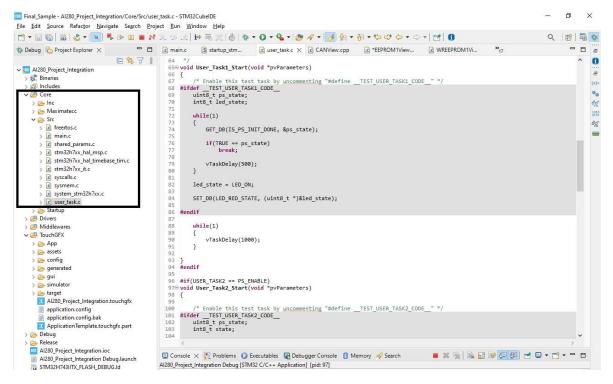
The GUI files for creating new applications should be added under the folder structure TouchGFX\gui\src. Section 3 describes in detail how a sample application can be written and integrated with the SDK and tested on a MAX AI280 board.



Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

#### 3.3.4.2 Blank User Task Files

The SDK has included some blank user tasks that can be used by the application developers if they would like to create some tasks that run in the platform independent of the Touch GFX.As shown in the image below, you can find the user tasks in the path core/src/user\_task.c as shown below.



Please refer to section 3.5 to understand how to edit these user tasks and integrate with the SDK and tested on a MAX AI280 board.

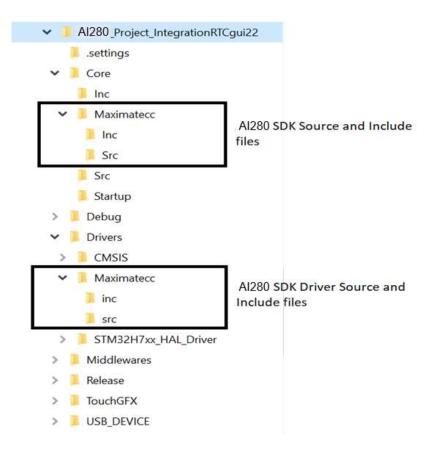
Dept: Software	User Manual Document	Rev No.	<b>Rev 3.0</b>
		Date:	Oct 17, 2022

# 4 SDK Application Development Procedure

In this section we will walk you through the procedure to write a simple Touch GFX application using the blank project file provided in the S/W release package and compile the same with the STM Cube IDE and flash it on the MAXAI280 hardware and test it. We will also provide you the details on how you can debug the application using the ST link debugger.

#### 4.1 Blank Project Installation and loading

The MaxAI280 SDK release comes with 3 project files as described in the Section 2.3. Please go to the folder with the zip file (AI280\_GettingStart.zip) containing the blank project file is available and unzip the same. You will get the directory, AI280\_Project\_IntegrationRTCgui22 after unzip the file. The below image shows the contents and path of the blank project after it has been unzipped.



Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

#### 4.1.1 STM32 Cube IDE Setup.

Click on **STM32CubeIDE Launcher** and provide the Workspace name on the pop-up window given below and then click on **Launch** option given at the bottom right corner of the Pop-up notification to open the IDE with any desired workspace. We will then be adding our project into this workspace.

	ectory as workspace DE uses the workspace directory to store its preferences and d	evelopment artifacts.	
Workspace:	C:\Users\Rajesh\STM32CubeIDE\SAMPLEapk22	•	Browse
🗖 l lse this a	s the default and do not ask again		
_ ose tins o			

Once the user enters the Workspace, he can see the below screen.

1 Information Center 🕄		1 1 1
STM32CubeIDE H	ome	
Start a project	Welcome to STM32Cub	eIDE Support & Community
Ea	What's new	
Start new STM32		Twitter
project	STM32	Youtube
Ea Start new project from	MPU Linux Bootloorder Deployment Update MPU Linux	ST Home ST Community ST Longevity Commitment
STM32CubeMX file (.loc)	Cootloader on running target & test	Standalone STM32 Tool
Import project	How to access it : Right-click on a Cortex-A project > Run As	
	> STM32 Cortex-A Bootloader Deployment	© STM32CubeMon-Pwr
Ē	Quick links	STM32CubeMon-RF     STM32CubeMon-UCPD     STM32CubeProg
import STM32Cube example	Access to Videos	
	Read STM32CubeIDE Documentation	Application Tools
	Getting Started with STM32CubeIDE	eDesignSuite     AlgoBuilder

The user must now import the blank project into this workspace. Click file menu and select the option "Open Projects from File System". Refer the image given below.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

New	Alt+Shift+N>		 000 × ×
Open File			
Open Projects from File System Recent Files	h		
Close Editor Close All Editors	Ctrl+W Ctrl+Shift+W	Welcome to STM32CubeIDE	Support & Community
Save Save As Save All	Ctrl+S Ctrl+Shift+S	What's new	Twitter
Revent		STM32 CubeIDE	Facebook Youtube
Move Rename	F2		ST Home ST Community
Refresh Convert Line Delimiters To	F5 ,	Video tutorials	ST Longevity Commitment
Print	Ctrl+P	Available through	
Import Export		<pre> the new video     browser</pre>	Standalone STM32 Too
Properties	Alt+Enter		Calife
Switch Workspace Restart	>		STM32CubeMX
nestart Exit		How to access it : Help > Information Center > Access videos	STM32CubeMonitor
			 STM32CubeMon-Pwr     STM32CubeMon-RF
67		Quick links	STM32CubeMon-RF     STM32CubeMon-UCPD     STM32CubeProg
Import STM32Cube		Ccess to Videos	Updates Available
example		Read STM32CubeIDE Documentation	Updates are available for your soft Active click to review and install updates. Go to S You will be reminded in 4 Hours. Set reminder preferences

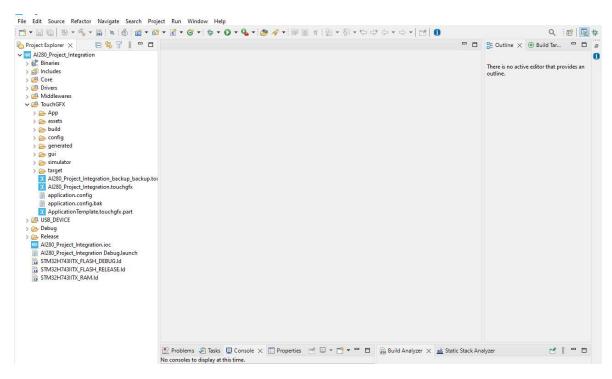
Clicking on the "Open Projects from File System" will bring up the below window. Please provide the path of the unzipped blank folder in the import source option.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Information Center × STM32CubeIDE	Hop III Import Projects fro	m File System or Archive		>	
S IM32CubelDE		•		/	
Start a project		n File System or Archive he content of your folder or archive file to find projects and	import them in the IDE.		Support & Community
E₀ Start new	Import source:	AI280_latest_source_code\AI280_Project_Integration	~	Directory Archive	witter
STM32 project	type filter text			Select All	acebook
	Folder		Import as	Deselect All	outube T Home
	Al280_Project	Integration	Eclipse project	Descreet Mu	T Community
E₀					T Longevity Commitment
Start new project from				1 of 1 selected Hide already open projects	
STM32CubeMX file (.ioc)	Close newly impor	ted projects upon completion			
. 50 M.					
	Use installed project of	configurators to:			Standalone STM32 Tool
	Search for nested p	<u>configurators</u> to: projects			Standalone STM32 Tool
Cr) Import	Search for nested p	<u>configurators</u> to: projects			STM32CubeMX
Survey and State	Search for nested p Detect and configu Working sets	configurators to: projects are project natures			STM32CubeMX STM32CubeMonitor
Import	Search for nested p	configurators to: projects are project natures		New	STM32CubeMX
Import project	Search for nested p Detect and configu Working sets	configurators to: projects are project natures		New	STM32CubeMX STM32CubeMonitor STM32CubeMon-Pwr STM32CubeMon-RF STM32CubeMon-UCPD
Import project	Search for nested p  Detect and configu  Working sets  Add project to v	configurators to: projects are project natures	Sho		STM32CubeMX STM32CubeMonitor STM32CubeMon-Pwr STM32CubeMon-FF STM32CubeMon-UCPD STM32CubeProc
Import project	Search for nested p  Detect and configu  Working sets  Add project to v	configurators to: projects are project natures	Shor	✓ Select	STM32CubeMX STM32CubeMonitor STM32CubeMon-Pwr STM32CubeMon-FF STM32CubeMon-UCPD STM32CubeProc
Import project	Search for nested p  Detect and configu  Working sets  Add project to v	configurators to: projects are project natures	Shor	✓ Select	STM32CubeMX STM32CubeMonitor STM32CubeMon-Pwr STM32CubeMon-PcP STM32CubeMon-UCPD STM32CubeProg
Import project	Search for nested i Detect and configu Working sets Working sets	configurators to: projects are project natures		v Select	STM32CubeMX STM32CubeMonitor STM32CubeMon-Pwr STM32CubeMon-FF STM32CubeMon-UCPD STM32CubeProc
Import project	Search for nested p  Detect and configu  Working sets  Add project to v	configurators to: projects are project natures	Shor < Back: Next >	✓ Select	STM32CubeMX STM32CubeMonitor STM32CubeMon-Pwr STM32CubeMon-Pwr STM32CubeMon-UCPD STM32CubeProg Application Tools
Import project	Search for nested i Detect and configu Working sets Working sets	configurators to: projects ure project natures working sets		v Select	STM32CubeMX STM32CubeMonFWr STM32CubeMonFWr STM32CubeMonFVF STM32CubeMonFUCPD STM32CubeMonFUCPD STM32CubeMonFUCPD sTM32CubeMonFUCPD ebesignSube
Import project	Search for nested i Detect and configu Working sets Working sets	configurators to: projects are project natures		v Select	STM32CubeMX STM32CubeMonitor STM32CubeMon-Pwr STM32CubeMon-Pwr STM32CubeMon-UCPD STM32CubeProg Application Tools
Import project	Search for nested i Detect and configu Working sets Working sets	configurators to: projects ure project natures working sets		v Select	STM32CubeMX STM32CubeMonitor STM32CubeMon-Fwr STM32CubeMon-FF STM32CubeMon-UCPD STM32CubeMon-UCPD STM32CubeProg
Import project	Search for nested i Detect and configu Working sets Working sets	configurators to: projects ure project natures working sets		v Select	STM32CubeMX STM32CubeMonitor STM32CubeMon-Fwr STM32CubeMon-FF STM32CubeMon-UCPD STM32CubeMon-UCPD STM32CubeProg

Now click on Finish once it recognizes the project file. You will get the below screen once you click on Finish.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022



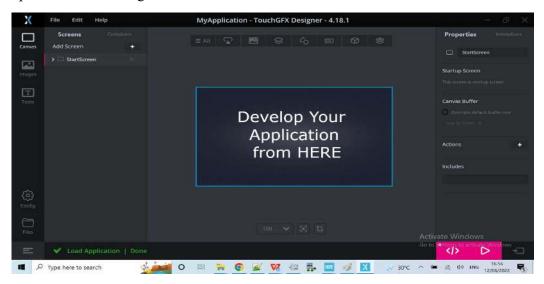
#### 4.2 Build and Flash Procedure

Now we will create our own TouchGFX UI and compile and test it on the board. Expand the TouchGFX folder from the STM32CubeIDE and double click on the AI280\_Project\_Integration.touchgfx file. Refer the image given below.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

	・ 🕄 ・ 🎯 ・ 🏇 ・ 🕖 ・ 🎥 🥖 ・ 🗟 🗐 🖷 🗄 ・ 🖗 ・ や や や ・ や ・ 🔂 🚺	Q 😰 🖬
- Project Explorer 🗙 📄 🖶 😭 🖓 🖇 📟 🗖		🗄 Outline 🗙 🛞 Build Tar 📟 🗖
Implet topionet     Implet topionet       Al280, Project_Integration       > ∰ Includes       > ∰ Includes </td <td></td> <td>Counter &amp; Counter and Counter</td>		Counter & Counter and Counter

Once you double click the AI280\_Project\_Integration.touchgfx file, TouchGFX IDE will be opened with our designed UI as shown below.



Now we will modify some text and test it on the board. Changed the text with "Modified this Text". Click "</>" button for generating the TouchGFX code. Refer the below image.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

X	File Edit Help	MyApplication - TouchGFX Designer - 4.18.1	
	Screens Containine Add Screen +		Properties Hiteractions
2	→ 🗆 StartScreen 🛛 🕫		StartScreen
agen			Startup Screen This screen is itertagescreen
T			Canvas Buffer
		Modified this Text	Actions +
			Includes
		39 <b>2</b> 14	Activate Windows
			Go to ethogs to actuate Windows

Go to STM32CubeIDE Screen. And click Project ===> Build All. Refer the below image.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Project Explorer X Project Suplorer X   Project Suplorer X Project Suplorer X   Image: Support Suppor		) Build Tar
Al280 Project_Integration     Build All     Ctrl+B       > % Binaries     Build Configurations     >       > @ Includes     Build Ornigurations     >       > @ Core     Build Project     >       > @ Drivers     Build Working Set     >       > @ Middlewares     Clean     >       > @ Assets     Build Automatically     >       > @ assets     Build Automatically     >       > @ build     Generate Report     >	There is no active en	
> Image: Second Sec		ditor that provides a
<ul> <li>&gt; Michaddes</li> <li>&gt; Source</li> <li>&gt; Sou</li></ul>		ditor that provides a
> ≥ Core     Build Working Set →       > ≤D hivers     Clean       > ≥D touchGFX     Build Automatically       > ≥→ sasets     Build Targets →       > ≥→ build     Generate Report       > ⇒→ config     Generate Code		
≥ Middlewares     Clean       ≥ Middlewares     Build Automatically       ≥ ⇒ App     Build Targets       > ⇒ sasets     Build Targets       > ⇒ build     Generate Report       > ⇒ config     Generate Code		
Yeig     Build Automatically       > ≥ App     Build Automatically       > ≥ assets     Build Targets       > ≥ build     Generate Report       > ≥ config     Generate Code		
> ≥ App Second		
> De sasets Build Fargets > > De build Constant Report > De config de constant Code		
> Config Constant Code		
> 🍅 config		
> E generated		
Comparison      Comparison      Properties     Comparison      Properties		
> 🦻 target		
X Al280_Project_Integration_backup_backup.tox		
X Al280_Project_Integration.touchgfx		
📄 application.config		
📄 application.config.bak		
X ApplicationTemplate.touchgfx.part		
B USB_DEVICE		
> 😂 Debug		
Delaura		
Al280_Project_Integration.ioc		
Al280_Project_Integration.ioc Al280_Project_Integration Debug.launch		
III Al280 Project_Integrationaice		
Al280_Project_Integration.ioc       Al280_Project_Integration Debuglaunch       STM32H743IITX_FLASH_DEBUGJd       G_STM32H743IITX_FLASH_RELEASE.id		
III Al280 Project_Integrationaice		
Al280_Project_Integration.ioc         Al280_Project_Integration Debug.launch         TM32H743IITX_FLASH_DEBUG.ld         TM32H743IITX_FLASH_RELEASE.ld		
Al280_Project_Integration.ioc Al280_Project_Integration Debug.lsunch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld		
Al280_Project_Integration.ioc       Al280_Project_Integration Debuglaunch       STM32H743IITX_FLASH_DEBUG.Id       STM32H743IITX_FLASH_RELEASE.Id		
Al280_Project_Integration.ioc Al280_Project_Integration Debug.launch STM32H743HTX_FLASH_DEBUG.ld STM32H743HTX_FLASH_RELASE.id		
Al280_Project_Integration.ioc         Al280_Project_Integration Debug.launch         TM32H743IITX_FLASH_DEBUG.ld         TM32H743IITX_FLASH_RELEASE.ld		
Al280_Project_Integration.ioc         Al280_Project_Integration Debug.launch         TM32H743IITX_FLASH_DEBUG.ld         TM32H743IITX_FLASH_RELEASE.ld		
Image: Al280 Project_Integration.ioc         Al280 Project_Integration.ioc         STM32H741ITX_FLASH_DEBUG.id         STM32H743ITX_FLASH_RELEASE.id         STM32H743ITX_RAM.id		
A220 Project_Integration.icc     A230 Project_Integration.bebug.lsunch     STM32H743IITX_FLASH_BEUG.id     STM32H743IITX_FLASH_RELEASE.id     STM32H743IITX_RAM.id	ack Analyzer	1 00 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Image: Al280 Project_Integration Debug.launch         Image: TM32H743ITX_FLASH_DEBUG.ld         Image: TM32H743ITX_FLASH_RELEASE.ld         Image: TM32H743ITX_RAM.id		
Image: Al280 Project_Integration.ioc         Al280 Project_Integration.oc         Al280 Project_Integration.oc         STM32H743IITX_FLASH_DEBUG.id         STM32H743IITX_FLASH_RELEASE.id         Image: STM32H743IITX_RAM.id		
Image: Al280 Project_Integration.loc         Image: Al280 Project_Integration.bebug.lsunch         STM32H743IITX_FLASH_DBUG.ld         STM32H743IITX_FLASH_RELEASE.ld         STM32H743IITX_RAM.ld	/AI280_Project_Integrati	
Image: Al280 Project_Integration.oic         Al280 Project_Integration.oic         STM32H743IITX_FLASH_DEBUG.id         STM32H743IITX_FLASH_RELEASE.id         Image: Stm102H743IITX_FLASH_RELEASE.id         Image: St	/Al280_Project_Integrati	ion/Debug - Sep
Image: Alt280 Project_Integration.icc         Alt280 Project_Integration Debug.launch         STM32H743IITX_FLASH_BEUG.id         STM32H743IITX_FLASH_BEUG.id         STM32H743IITX_RAM.id	Al280_Project_Integrati	ion/Debug - Sep Free
A280_Project_Integration Debug Jaunch     STM32H743IITX_FLASH_DEBUGJd     STM32H743IITX_FLASH_RELASE.Id     STM32H743IITX_RAM.id      Problems Tasks Console × Properties * • • • • • • • • • • • • • • • • • •	/Al280_Project_Integrati	ion/Debug - Sep Free 638.20
Image: Al280 Project_Integration.ioc         Al280 Project_Integration.beuglaunch         STM32H743IITX_FLASH_DEBUG.Id         STM32H743IITX_FLASH_RELEASE.Id         Image: STM32H743	Final address         Size           0x08200000         2 MB           0x20020000         128 K	ion/Debug - Sep Free 8 638.24 KB 128 K
Image: Alt280 Project_Integration.icc         Alt280 Project_Integration Debug.launch         STM32H743IITX_FLASH_BEUG.id         STM32H743IITX_FLASH_BEUG.id         STM32H743IITX_RAM.id             Image: Problems @ Tasks @ Console × [] Properties [] + [] + [] + [] + [] + [] + [] + []	Image: Algorithm         End address         Size           0x08200000         2 MB         0x0200000         128 K           0x02020000         128 K         0x0208000         512 K	ion/Debug - Sep Free 8 638.21 (B 128 K (B 82.03
Image: Alt280 Project_Integration.icc         Alt280 Project_Integration Debug.launch         STM32H743IITX_FLASH_BEUG.id         STM32H743IITX_FLASH_BEUG.id         STM32H743IITX_RAM.id             Image: Problems @ Tasks @ Console × [] Properties [] + [] + [] + [] + [] + [] + [] + []	Final address         Size           0x08200000         2 MB           0x20020000         128 K	ion/Debug - Sep Free 8 638.21 (B 128 K (B 82.03

Once the build is success, you will get the console windows with "0 errors, 0 warnings". As shown below

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

• 🔟 🔟   🕲 • 💊 • 📖 ! 🖉 : 🕲 : 🔯 • 🖬	・ 🗟 • 🚱 •   ね • 🔕 •   🎥 🛷 •   副 🖩   日 • 初 • や 中 や • や •	1 🗹   🚺				Q	E	1 42
Project Explorer 🗙 📄 😫 🍟 🖬		-	- 0	BE Outline	× 🖲 B	Build Tar.		- 0
Al280 Project_Integration Al280 Project_Integration Al280 Project_Integration Al280 Project_Integration_backup_backup.to TouchGFX Duild				There is no a outline.				
	👔 Problems 🧔 Tasks 🗳 Console 🗙 🛄 Properties	X & & &	58 5	1 = B. C	1 🖻 🕻	3 - 6	} - =	- C
	CDT Build Console [Al280_Project Integration] 15:55:51 **** Build of configuration Debug for project Al288_Project_integration		-					
	make -j4 all -size AI280 Project_Integration.elf arm-none-eabl-size AI280 Project_Integration.elf text data bss dec hex filename 1135342 308236 132808 1375666 1808f2 AI280_Project_Integration.elf Finished building: default.size.stdout							

Now we will flash the binary using the ST JTAG, click debug [\*] icon to flash the code. Refer the highlighted ( $\square$ ) part in the below screen.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
F		Date:	Oct 17, 2022

1 두 🛄 🐘 등 두 🗞 두 📓 🦉 🖗 🗃 두 🚳 두 🕃 두 🧭 두 🌘 두 💁 주 🧶 두 🧶 두 🧶 두 🧶 🥔		E Outline × 🖲	Duga Tas		1
Project Explorer X E V V 8 E	- 0	E Outline X ()	Build Iar	- 0	
> Binaries					
> 🗊 Includes		There is no active ec outline.	litor that prov	rides an	
> 🥵 Core		outime.			Ī
> 😕 Drivers					
> 🥝 Middlewares					
🗸 😂 TouchGFX					
> 🗁 App					
> 📴 assets					
> 😕 build					
> 🗁 config					
> 🗁 generated					
> 👝 gui					
> 👝 simulator					
> 🗁 target					
X Al280_Project_Integration_backup_backup.toi					
X Al280_Project_Integration.touchgfx					
📄 application.config					
application.config.bak					
X Application Template.touchgfx.part					
> 😕 USB_DEVICE					
> 👝 Debug					
> 🔁 Release					
Al280_Project_Integration.ioc					
Al280_Project_Integration Debug.launch					
STM32H743IITX_FLASH_DEBUG.Id					
STM32H743IITX_FLASH_RELEASE.Id					
STM32H743IITX_RAM.Id					
💦 Problems 🧔 Tasks 📮 Console 🗙 🥅 Pro	perties 🗶 😽 🚱 🗔 🖥		📮 🕶 📑 🔻	- 0	

While flashing the code, you will get the below screen. Please click "Switch" button to continue.

Project Explorer 🛛 📑 🕫 🖓 🦷 🕴 🕾	□ B main.c B Ai280_Config.h B user_task.c B Watchdog_Platformservice.c B startup_stm32h743iitx.s =	-
A1280 Project Integration	61 ldr sp, =_estack /* set stack pointer */	
# Binaries	62 63/* Call the clock system initialization function.*/	
Includes	64 bl SystemInit	
* 10 Core	65	
🥴 🍅 Inc	66/* Copy the data segment initializers from flash to SRAM */	
> G Maximatecc	68 1d Confirm Perspective Switch	
> 😕 Src > 😂 Startup	69 14	
Orivers	70 md Diskind of launch is configured to open the Debug perspective when it suspends.	
Middlewares	71 b	
@ TouchGFX	72 73 Copy 73 Copy debug stack, variables and breakpoints.	
e Debug	74 Id	
© Release	75 st 76 ad Switch to this perspective?	
Al280 Project Integration.ioc	77 77	
# AI280 Project Integration Debug launch	78Loop	
# STM32H743IITX_FLASH_DEBUG.Id	79 ad Remember my decision	
STM32H743IITX_FLASH_RELEASE.id	80 cm 81 bc	
3 STM32H743IITX_RAM.Id	82 /* 2	
ButtonThread (in ButtonThread2)	83 ldr r2, =_sbss	
F Demo	84 ldr r4, =_ebss	
Keypad_testing	č	3
maxAl280_gauge	😫 Problems 🖉 Tasks 💷 Console 🖾 🗁 Properties 🗟 Build Analyzer 🚄 Static Stack Analyzer 🛹 Search 🖷 Progress	■ # % [ k & > (20 20 ) 년 • 년 •
	AI280 Project Integration Debug (STM32 Cortex-M C/C++ Application)	
	InitWhile : Enabled	
	Waiting for debugger connection	
	Debugger connected	
	Waiting for debugger connection	
	Debugger connected	1 12 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
	Whiting for debugger connection	
	Waiting for debugger connection	Activate Windows Go to Settings to activate Windows

Once the flashing is completed you will get the below screen.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
T		Date:	Oct 17, 2022

	\$ • 0 • \$ • \$ * \$ * \$ * \$ * \$ * \$ * \$ * \$ * \$	۹ ( <u>ه</u> )	EL 🖬 🗘
Debug II Deroject Explorer     E II 1 0     Construction Debug (STM32 Cortex-M C/C+     S Al280 Project Integration Debug (STM32 Cortex-M C/C+     S Al280 Project Integrational (cores: 0)     V 3 Al280 Project Integrational (cores: 0)     V 3 Al280 Project Integrational (cores: 0)     If main) at maint at mainth at maint	(# manc.*) I# A280 Confight # user taskc # Watchdog Platformservice.c # startup stm32h743Hoxs 170 #fifded DEBUG 171 VectorBase_Config(); 172 #endif 173 /* USER CODE END 1 */		^
■ manip at mancrite Scouldscele af arm-none-eab-ight (3.1.2019)211) af ST-LINK (ST-LINK GDB server)	<pre>174 175 /* MCU Configuration*/ 176 177 /* Reset of all peripherals, Initializes the Flash interface and the Systick. */ 178 HAL_Init(); 179 179 179 179 179 179 179 179 179 179</pre>		-
	<pre>194 /* Initialize all configured peripherals */ 195 MX_GPIO_Init();</pre>		~
	196 MX CRC Toit().		3
	© Console ≅ ≋Progress & Problems © Executables ®Debugger Console ® Memory I Search M280_Project_Integration Debug [STM32 Cortex-M C/C++ Application]	● # 등  등 중 위원의 ㅋ 0 • 11	• • •
	Download verified successfully		
		Activate Windows Go to Settings to activate Windows	
< >	Writable Smart Insert 178 : 1 : 4981		20
Type here to search	🕵 H 🖬 🚱 💉 💘 🕮 🍱 🔣 📢 ^ B	☑ 🛥 🬾 ⊄× ENG 16:33	<b>R</b> 4)

Click Resume button (Highlighted 🗖 ) in the above image) to run the application. You will get the below Screen on the AI280 board.



Date: Oct 17, 2022	Dept: Software	pt: Software User Manual Document	Rev No.	Rev 3.0
	Depti Soltmare		Date:	Oct 17, 2022

We were successfully able to compile and flash the blank project with minimal changes to the MAX AI280 board. In the next section we will elaborate with an example on how you can leverage the features of the SDK.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

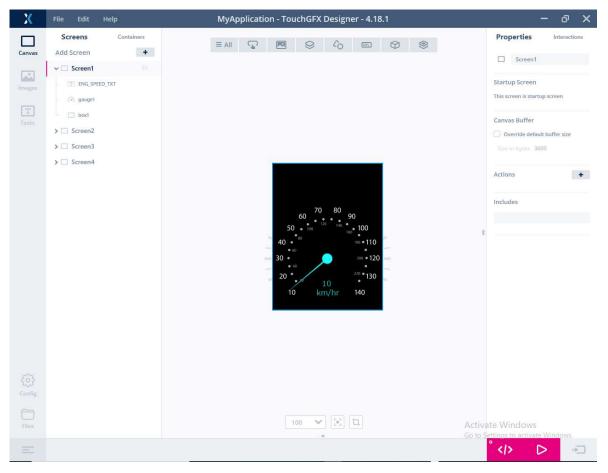
# 4.3 CAN Module Demo

This section elaborates the procedure on how the Touch GFX application interacts with the SDK. We have used the CAN module demo to explain this procedure.

### 4.3.1 Adding new GUI elements in the TouchGFX Screen

In this section, we will elaborate how we can add new elements in the touch GFX screen and then link them with the SDK.

Select the "Text Area" under "All". Now you can type the text in the text box. You need to select wildcard1 option for Gauge value. Gauge will fetch the data from CAN analyser and display it on the screen. Refer the below screen.



To create the gauge, select gauge under the section "All". The initial value can be from 0(Min) to 140(Max). You can assign the initial value under Properties section. Refer the below screen.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Х	File Edit	Help	MyA	pplicati	on - Tou	chGFX	Design	er - 4.18	.1			ł	- 0
Canvas Canvas Images Texts	Add Screen Add Screen Content Screen Content Screen Content Screen Content Screen Content Screen Content Conte	Containers		ي م	m - Tou			E7 - 4.10	P			Properties W 276 Lock Visible Style Background Background Image Background Image Background Image	Interaction H 228
				ç	50 40 30 20 10	20	80 9 140 160 140 n/hr	0 100 180 • 110 200 • 120 220 • 130 140				× 0 Gauge Rotation Center × 138 Angle Start -130 Value Range Min 10 Initial Value 40	Y 138 End 130 Max 140
ې anfig												Animation Animate move Easing Linear Easing Option In O Needle	ment Iut InO
files					1	00 🗸		12		A	ctiva o to Se	t www.inclows ettings to activate	e Windows.

Select the "Text Area" under "All". Now drag the Text Area and place the text box over the circle. You need to select wildcard1 option for gauge value. Gauge will fetch the data from CAN and display it on the screen. Refer the below screen.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

File Edit Help		MyApplica	ion - TouchGl	X Designe	er - 4.18.1			- O
Screens 0	Containers			<u>^</u>			Properties	Interaction
Add Screen	+	= All •1.		6		8	X -18	Y 70
Screen1							W 276	H 228
T ENG_SPEED_TXT							C Lock	
- (?) gauge1							Visible	
box1							Style	
> 🗌 Screen2								
>  Screen3								
>  Screen4							Background	
							Background Im	
								background.png
			p		-		Background Of	
			60	70 80 90	)			Y O
			50 - 100	120 140 160	100		Gauge	
			40	. /	180 • 110		Rotation Cente	r:
			30 •		200 • 120		X 138	Y 138
			20		<sup>220</sup> 130		Angle	
			10	100	140		Start -130	End 130
				KIII/III	140		Value Range	
					e e e e e e e e e e e e e e e e e e e		Min 10	Max 140
							Initial Value	100
							Animation	
							Animate m	ovement
							Easing	
							Linear	
							Easing Option	
			100	<ul> <li>E</li> </ul>			Activate Windov	Out InO

Click "</>" button for generating the TouchGFX code. Refer the highlighted part (□) in the below screen.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Soltware		Date:	Oct 17, 2022

Easing Linear Easing Option V In Activate Windows		elp	ign	ner - 4.1	8.1				– o
d Screent       ×	iners	Containers	~	1222		(		Properties	Interac
Image: Second image: Secon	+	+	9		Ŷ	0		X -18	Y 70
i boxin       I boxi         Screen2       Background Ima         Screen4       I boxin         I boxin       I boxin         I boxin       I boxin         I boxin       I boxin         Screen4       I boxin         I boxin       I boxin								W 276	H 228
uppe1       Style         en2       en3         en4       Image: Style         Image: Style       Ima		TXT						Lock	
Screen3 Screen4 Scr								Visible	
Screen3 Screen4  Background Ima Bac								Style	
n3 n4									
Screen4 Sc									
baceground ma baceground ma baceground ma baceground offs × 0 Gauge Rotation Center × 138 Angle Start -130 Value Range Min 10 Indial Value 0 Animation Earing Linear Earing Option ↓ 10 ↓ 10									
Background Offs X 0 Gauge Rotation Center X 138 Ange Start -130 Value Range Min 10 Initial Value To Yold Participant Initial Value To Yold Participant Initial Value Initial									
60       70       80       90       Gauge         60       50       100       100       Gauge         10       100       100       100       130         10       100       140       140       100         10       Initial Value       Animation         Gasing       Linear       Casing         100       100       140       Activate Windows									
60 V V V V V V V V V V V V V V V V V V V									
40 10 10 10 10 10 10 10 10 10 10 10 10 10			0	90				(0.2	
x 138 Ange Start -130 Value Range Min 10 Initial Value Animation □ Animation □ Animation			40 1					Gauge	
Ange 20 0 20 130 10 km/hr 140 Value Range Min 10 Initial Value * Animation Animate mo Easing Linear				180 •110	-0			Rotation Center	
20       20       130       130       130       5tart       130         10       km/hr       140       140       10       Initial Value         Min       10       Initial Value       Animation       Animation         Initial Value       Initial Value       Initial Value       Initial Value         10       Initial Value       Initial Value       Initial Value         Initial Value       Initial Value       Initial Value       Initial Value					0				Y 138
10 km/hr 140 Value Range Min 10 Initial Value Animate mo Easing Linear Easing Option ✓ In				220 130	-				
tinar t									End
Initial Value * Animation □ Animate me Easing Linear Easing Option ✓ In 100 ▼ 💽 🖽 Activate Windows					<u> </u>				Max
Animation Animate mo Easing Unear Easing Option V In Activate Windows									
□ Animate mo Easing Linear Easing Option ✓ In 100 ▼ 20 12 Activate Window?								Initial Value 13	30
Easing Linear Easing Option ~ In Activate Windows								Animation	
Libear Easing Option Vin Activate Windows								Animate move	ement
Tan Activate Windows									
100 V 🕃 🖪 Activate Windows									
100 🗸 😥 🗔 Activate Windows									Dut
			C	17,			Activat	e Windows	
Go to SetNeedle, active							Go to Set	Needle, activat	e Windo

After the code is generated, please navigate to the STM 32 Cube IDE to edit the code and link the graphical elements to the SDK.

### 4.3.2 Edit the DB Variables

Now we are going to enable the graphical elements added in the last section to configure the CAN module. To do the same we will need to link it with the SDK DB.

In this section, we will describe how to link the DB Variables to the TouchGFX elements.

Go to STM32CubeIDE Screen. Add the following function declarations in

"Screen1Presenter.hpp" file under the "TouchGFX/gui/include/gui/Screen1\_screen" folder structure.

- " virtual void keyUpdate(uint8\_t val); "
- " virtual void handleTickEvent();"
- " virtual void trigger();"

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

<pre>Project Explore X</pre>		- 6 - 6 - * - 0 - 4					144	Q :		
<pre>v Brains v Brains v Brains v Brains v Brains v Constructions v Brains v Brains</pre>		A REAL PROPERTY AND A REAL		C J1939Data_L	🔝 main.c	C Screen 1Pres	ⓑ Screen1Pres ×	"1		, E
<pre>includes Core Drivers Drivers Drivers Divers Drivers D</pre>										1
<pre>include <gu modellisterer.hpp="" rodel=""> include <gu <gu="" include="" modellisterer.hpp="" rodel=""> include <gu <gu="" include="" modellisterer.hpp="" modellisterer<="" rodel="" td=""><td></td><td></td><td>N_IIFF</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></gu></pre>			N_IIFF							
■ Mindlewers       ■ Mindlewers         ● Mindlewers       ■ using namespace touchgfx;         ● InduchDFX       ■ class ScreenIVers         > ● assets       ■ class ScreenIVersenter ; public touchgfx::Presenter, public ModelListener         > ● assets       ■ intilize Class ScreenIVersenter ; public touchgfx::Presenter, public ModelListener         > ● assets       ■ intilize Class ScreenIVersenter (ScreenIView& v);         > ● assets       ■ intilize Class ScreenIVersenter (ScreenIView& v);         > ● assets       ■ intilize Class ScreenIVersenter (ScreenIView& v);         > ● assets       ■ intilize Class ScreenIVersenter (ScreenIView& v);         > ● assets       ■ intilize Class ScreenIVersenter (ScreenIView& v);         > ● assets       ■ intilize Class ScreenIVersenter (ScreenIView& v);         > ● assets       > intilize Class ScreenIVersenter (ScreenIView& v);         > ● assets       > intilize Class ScreenIVersenter(ScreenIView& v);         > ● assets       > intilize Class ScreenIVersenter();         > ● assets       > virtual void activate function is called automatically when this screen is "switched out"         > ● ScreenIView.hpp       /**         > ● ScreenIview.hpp       /**         > ● ScreenIview.hpp       /**         > ● ScreenIview.hpp       /**         > ● ScrenIview.hpp       /** <td></td>										
Middlewares       7       using namespace touchgfx;         0       Class ScreenlView;         0       class ScreenlView;         0       config         >>       sestest         >>       bound         >>       config         >>       serenlPresenter : public touchgfx::Presenter, public ModelListener         >>       config         >>       serenlPresenter(ScreenlView&v);         >>       serenlPresenter(ScreenlView&v);         >>       screenlPresenter(ScreenlView&v);         >>       screenlPresenter(ScreenlView&v);         >>       model         >>       screenlPresenter(ScreenlView&v);         >>       screenlPresenter(ScreenlViewAv);         >>       ScreenlPresenter(ScreenlViewAve);         >>       ScreenlPresenter();         >>       ScreenlPresenter() {};         >>       ScreenlPresenter();         >>       ScreenlPresenter();         >>       ScreenlPresenter();         >>       splication.config         >>       splication.config         >>       splication.config         >>       splication.config         >>       splication.config			.hpp>							
<sup>3</sup> DuchGFX <sup>3</sup> ⊂ lass ScreenlView; <sup>3</sup> ⊂ lass ScreenlPresenter : public touchgfx::Presenter, public ModelListener <sup>3</sup> ⊂ lass ScreenlPresenter : public touchgfx::Presenter, public ModelListener <sup>3</sup> ⊂ lass ScreenlPresenter : public touchgfx::Presenter, public ModelListener <sup>3</sup> ⊂ gui <sup>3</sup> ⊂ greenlPresenter/ScreenlView& v); <sup>3</sup> ⊂ greenlPresenter/ScreenlView& v); <sup>3</sup> ⊂ greenlPresenter/Presenter(ScreenlView& view); <sup>3</sup> ⊂ greenlPresenter/ScreenlPresenter() {}; <sup>3</sup> ⊂ greenlPresenter(); <sup>3</sup> ⊂ greenlPresenter(); <sup>3</sup> ⊂ greenlPresenter(); <sup>3</sup> ⊂ greenlPresenter();										
<pre>class ScreenIView; id class ScreenIView; id c</pre>			x;							
<pre>&gt;&gt; bidd &gt;&gt; bidd &gt;&gt; bidd &gt;&gt; config &gt;&gt; generated &gt;&gt; ge</pre>	_									
<pre>&gt;&gt; buid &gt;&gt; buid &gt;&gt; config &gt;&gt; generated &gt;&gt; generated &gt;&gt; generated &gt;&gt; goui &gt;&gt; goui</pre>										
<pre>&gt;&gt; config &gt;&gt; gui &gt;&gt; gui &gt;&gt; gui &gt;&gt; gui &gt;&gt; common &gt;&gt; model &gt;&gt; common &gt;&gt; model &gt;&gt; common &gt;&gt; forcenl/resenter.hpp &gt;&gt; forcenl/resent</pre>			: public touchgfx:	:Presenter, public	ModelListene	۲				
<pre>&gt;&gt; generated is given and it is screeniPresenter(ScreeniView&amp; v); &gt;&gt; generated is screeniPresenter(ScreeniView&amp; v); &gt;&gt; gui is screeniPresenter(ScreeniView&amp; v); &gt;&gt; gui is screeniPresenter(ScreeniView&amp; v); &gt;&gt; gui is screeniPresenter(ScreeniView&amp; v); &gt;&gt; generation is called automatically when this screen is "switched in" &gt;&gt; generation is screeniPresenter(); &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; generation is called automatically when this screen is "switched out" &gt;&gt; g</pre>										
y = gui       15       /**         Y = gui       169       /**         Y = gui       17       * The activate function is called automatically when this screen is "switched in"         Y = gui       18       * (fex. made active). Initialization logic can be placed here.         Y = gui       19       *//         Y = gui       *       *         Y = guisation.config       *       *         Y = aplication.config       *       *         Y = aplication.config       *       * <td></td> <td></td> <td>reen1View&amp; v):</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>			reen1View& v):							
<pre>&gt;&gt;&gt;&gt; include &gt;</pre>		15								
<pre>* @ gui</pre>		100 /								
<pre>&gt; common &gt; commo</pre>						"switched in"				
<pre>&gt;</pre>			e). Initialization	logic can be place	d here.					
<pre>&gt; b streen_streen &gt; b StreenIPstenter.hpp &gt; b StreenIPstenter.hpp &gt; b StreenIPstenter.hpp &gt; b StreenIPstenter.hpp &gt; b streen_streen &gt; b streen &gt; b stre</pre>			te();							
<pre>&gt; Dis ScreenTPresenter.hpp &gt; ScreenTPr</pre>		21								
> ScreenView.hpp     * (ds. made inactive). Technic for fide.index for the fide.in				N 2016-1 442 - 67	12127					
<pre>&gt;&gt; interms is streen_streen &gt;&gt;&gt; interms is streen_streen &gt;&gt;&gt; interms is streen_streen &gt;&gt;&gt;&gt; interms is streen_streen &gt;</pre>						is "switched out"				
<pre>&gt; Screen3_screen &gt; Screen4_screen &gt; Screen4_screen &gt; Screen4_screen &gt; Screen4_screen &gt; Screen4_screen &gt; Screen4_screen &gt; Virtual void deactivate();  virtual void deactivate(); virtual void deactivate(); virtual void keyUpdate(uint8_t val); virtual void trigger(); virtual void void watchdogstatus(uint8_t *val); X Al280_Project_Integration.touchgfx     application.config     application.config     application.config     screen1Presenter();     Screen1Pres</pre>			ive). Jeardown Tunc	cionality can be p	taced nere.					
<pre>&gt; Screen1_screen &gt; Screen1_screen &gt; Screen1_screen1Presenter() {};      virtual void keyUpdate(uint8_t val);     virtual void krigger();     virtual void krigger();     virtual void void krigger();     screen1Presenter();     Screen1Presenter();     Screen1View&amp; view;     Screen1View&amp; view;     screen1View&amp; view;     screen1View&amp; view;     wendlif // SCREENIPRESENTER_HPP     VOID note klaudiointe     Void kardinger()     View klaudinger()     view klaudinger()</pre>			<pre>vate();</pre>							
<pre>&gt; src virtual voirteul voirteul voirteul voirteul voirtual vo</pre>										
<pre>&gt; &gt; imulator &gt; &gt; &gt; imulator &gt; &gt;</pre>		28 virtual ~Screen1Pre	esenter() {};							
<pre>&gt;&gt; brget virtual void trigger(); X Al280_Project_Integration_backup_backup_tox X Al280_Project_Integration_backup_backup_tox A Ap20_Project_Integration_backup_backup_tox application.config application.config application.config backup_tox A Ap20_Project_Integration_backup_tox application.config backup_tox A Ap20_Project_Integration_backup_tox backup_tox B Application.config B Application.config B</pre>			ata(uint0 + ual).							
X       Al280, Project_Integration, backup, ba										
X     Al280 Project_Integration.touchgfx     33       B     application.config     35       Screen1Presenter();     36       Screen1Presenter();     36       V     ApplicationTemplate.touchgfx.part     37       Screen1View& view;     38       Debug     38       P     Release       V     VSR DEVICE       40     #endif // SCREENIPRESENTER_HPP				val);						
application.config       34       private:         application.config       35       ScreentPresenter();         36       ScreentPresenter();         37       ScreentPresenter();         Bebug       37       ScreentPresenter();         Bebug       38       };         Bebug       38       };         Bebug       39       ;         UDB DEVICE       40       #endif // SCREENIPRESENTER_HPP         UDB DEVICE       41		33								
application.config.bak ability applicationTemplate.touchgfx.part below										
X ApplicationTemplate.touchgfx.part     37     Screen1View& view;       >> Debug     38     );       >> Release     39       >>> USB_DEVICE     40       #endlf // SCREENIPRESENTER_HPP										
Bebug     38 };       Belase     39       WB DDVICE     40       Hendlf // SCREENIPRESENTER_HPP       USB DDVICE     41										
Kelease     40 #endif // SCREENIPRESENTER_HPP     41     41										
UDB OFFICE     de fenat // Skreetarkesentek_HPP     de										
Al200 Device to the section in a			TER_HPP							1
	Al280_Project_Integration.ioc									
	STM32H743IITX_FLASH_DEBUG.Id	Problems 🖉 Tasks 📮 Consol	e 🗙 🔲 Properties				" 🗖 🔜 B 🛋 S	A S	× =	3

Add the following functions which will be used for the CAN Module in

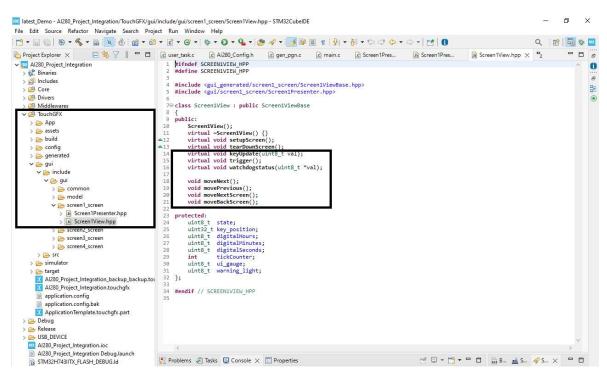
"Screen1View.hpp" file under the

"TouchGFX/gui/include/gui/Screen1\_screen" folder structure.

```
"virtual void keyUpdate(uint8_t val);"
"void trigger();"
"void moveNext();"
"void movePrevious();"
Add the following functions for screen navigation purposes.
```

"void moveNextScreen();" "void moveBackScreen();"

Dept: Software	User Manual Document	Rev No.	Rev 3.0
- Prostant		Date:	Oct 17, 2022

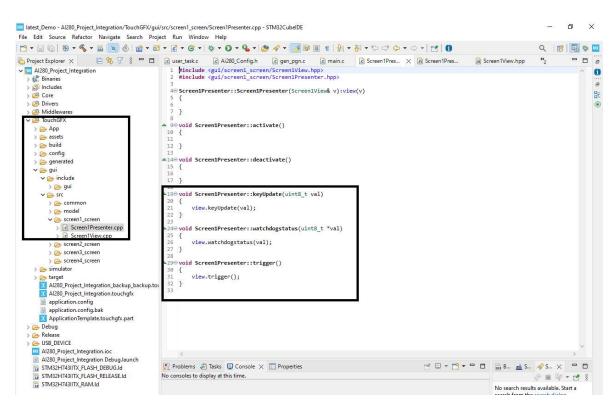


Add the following keypad function to access the functionality in "Screen1Presenter.hpp" file under the

"TouchGFX/gui/include/gui/Screen1\_screen" folder structure.

```
void Screen1Presenter::keyUpdate(uint8_t val);
{
    view.keyUpdate(val);
}
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022



To access the DB variables, the following header need to be added.

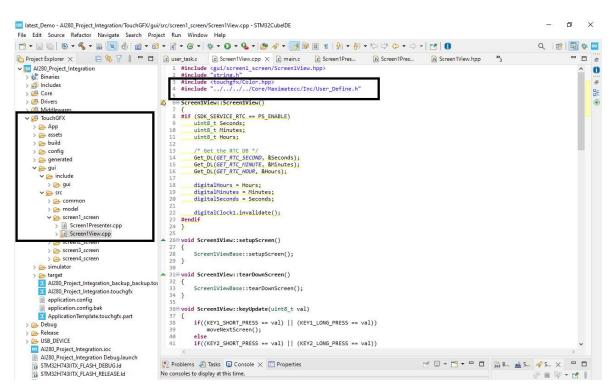
#include "../../../Core/Maximatecc/Inc/User\_Define.h"

To highlight the selected text, include the following line.

#include <touchgfx/Color.hpp>

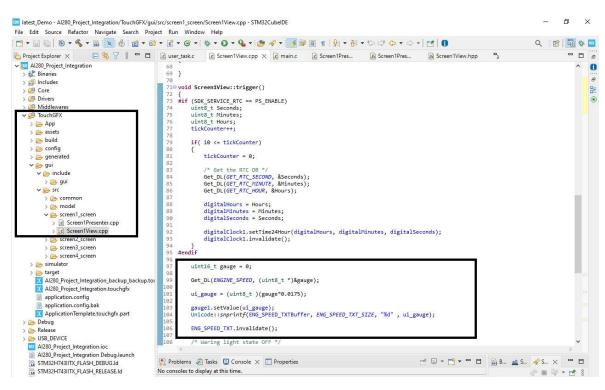
Refer the below screen to see the code snippet and file path.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022



Based on the Gauge data, it will adjust Needle movement. Get\_DL function is used to read the DB variable of CAN data. Set\_DL function is used to modify the DB variable of CAN.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
- Province		Date:	Oct 17, 2022



4.3.3 Configurations

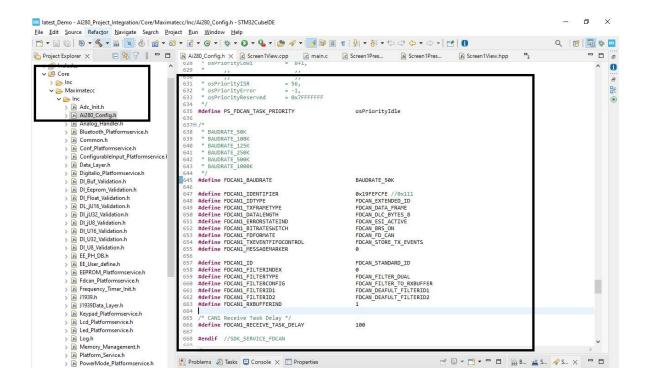
This section describes the configurations required for CAN module. This default configuration can be done in the AI280\_config.h file under the section core/Maximatecc/Inc.

The below are the configurable parameters available for CAN module

To configure these, we are using

FDCAN1_BAUDRATE	BAUDRATE_50K
FDCAN1_IDENTIFIER	0x19FEFCFE //0x111
FDCAN1_IDTYPE	FDCAN_EXTENDED_ID
FDCAN1_TXFRAMETYPE	FDCAN_DATA_FRAME
FDCAN1_DATALENGTH	FDCAN_DLC_BYTES_8
FDCAN1_ERRORSTATEIND	FDCAN_ESI_ACTIVE
FDCAN1_BITRATESWITCH	FDCAN_BRS_ON
FDCAN1_FDFORMATE	FDCAN_FD_CAN
FDCAN1_TXEVENTFIFOCONTROL	FDCAN_STORE_TX_EVENTS
FDCAN1_MESSAGEMARKER	0

FDCAN1_ID	FDCAN_STANDARD_ID
FDCAN1_FILTERINDEX	0
FDCAN1_FILTERTYPE	FDCAN_FILTER_DUAL
FDCAN1_FILTERCONFIG	FDCAN_FILTER_TO_RXBUFFER
FDCAN1_FILTERID1	FDCAN_DEAFULT_FILTERID1
FDCAN1_FILTERID2	FDCAN_DEAFULT_FILTERID2
Kindly refer the below screen for code snippet.	



Now we need to compile and test it on the board. For compilation, follow the instructions provided under the section 3.2

## 4.3.4 Output

Once we flashed the updated binary on MAX AI280 board we will observe that the application we designed is launched on the GUI.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
	oser munum Document	Date:	Oct 17, 2022

Refer the below screens captured to show different Gauge values being updated in the UI.



Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022



Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022



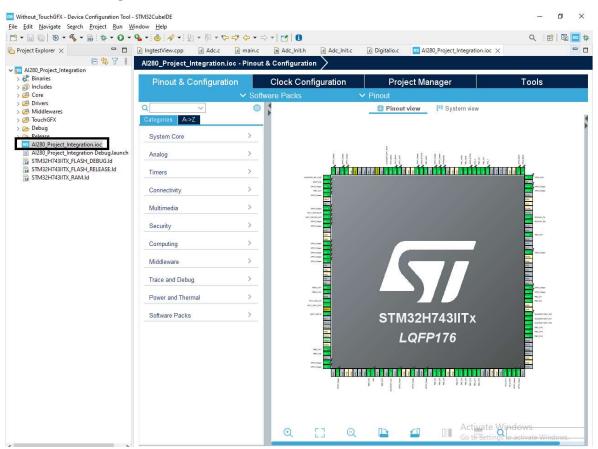
Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 4.4 LED Demo

The demo application is written for user who would like to write application without using the TouchGFX. They can use the same blank project released in the SDK, but disable the touch GFX and then write code which will link with the SDK and use the functionalities but will have a blank UI.

To Disable TouchGFX we need to click on AI280\_Project\_Integration.ioc.

Go to STM32CubeIDE Screen. Now double click on AI280\_Project\_Integration.ioc. The below screen will come up.



You will get the below screen once you click on AI280\_Project\_Integration.ioc. Now click Pinout & Configuration  $\rightarrow$  Categories  $\rightarrow$  Software Packs  $\rightarrow$  STMicroelectronics.X-CUBE-TOUCHGFX. Now you can untick Graphics Application box. This will disable the TouchGFX configurations. Refer the highlighted areas in the below screen.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
1		Date:	Oct 17, 2022

9 • 🔄 🐚   🥸 • 🗞 • 📷   🔅 • 🔘 •	C IngtestView.cpp C Adc.			Adc_Init.c	Digitalio.c	Al280_Project_Integrati	Q 📑 🖬	☆ <u>□</u>
E 😫 🎖 🕴	Al280_Project_Integration		Part -					
Al280_Project_Integration > \$\\$ Binaries	Pinout & Configu	ration	Clock Con	figuration	Proir	ect Manager	Tools	
> 🗊 Includes > 😕 Core	Findut & Conligu		ftware Packs		✓ Pinout	ectivianagei	Tools	
> 🛃 Drivers > 🛃 Middlewares	Q	0	STMicroelectronics			💭 Pinout v	riew 📲 System view	
> 😕 TouchGFX	Categories A->Z		_	Mode				
> 🗁 Debug > 📴 Release	System Core	>	🗹 Graphics Ap	plication				1
Al280_Project_Integration.ioc Al280_Project_Integration Debug.launch	Analog	>						
STM32H743IITX_FLASH_DEBUG.Id	Timers	>						
STM32H743IITX_RAM.Id	- Timers							
	Connectivity							
	Multimedia	>				111 1	u. H. R. adam. H	- 1
	Security	>						
	Computing	>						
	Middleware	>					$\nabla$	
	Trace and Debug	>		Configuration	_	_		
	Power and Thermal	>	Reset Configura			- s	TM32H743IITx LQFP176	
	Software Packs	~	TouchGFX Get	nerator 🛛 📀 User	Constants	=		
	STMicroelectronics.X-	CUBE-TO	Configure the below Configure	0 0	G (FMC)	<u>And the second se</u>		
			Height Framebu Buffer Lo	320 pixels 320 pixels ffer Double Buffe cati By Allocation se 0x60000000	n			
				nk ID FMC Bank 4		⊙ □Acti	vate Windows Settings to activate Windows.	<b>T</b> ET

Now click Pinout & Configuration  $\rightarrow$  Software Packs  $\rightarrow$  Select Components  $\rightarrow$  STMicroelectronics.X-CUBE-TOUCHGFX  $\rightarrow$  Graphics Application  $\rightarrow$  Application. Then select "Not selected" from the list. Then click on OK.

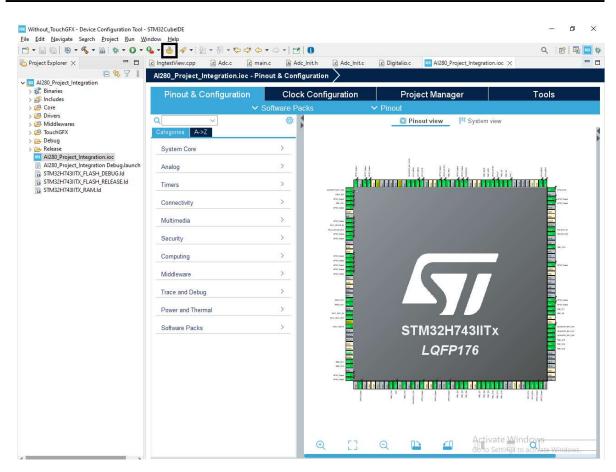
Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Pinout & Configurati	on	Clock Configuration	Project Manager	Tools
		✓ Software Packs	✓ Pinout	
s Q ~	STN	licroelectronics.X-CUBE-TOUCHGFX.4.1	18.1 Mo 🔯 Pinout view 🕮 System	view
Software Packs Component Selector	_			Σ
acks			Details and warnings	
			Bundle details	~
Pack / Bundle / Component	Status		Pack STMicroelectronics X-CUBE-TOUCHGFX	4.18.1
STMicroelectronics X-CUBE-EEPRMA1		3.1.1 🖻 👻 İnstali	Name Application Version 4.18.1	
STMicroelectronics X-CUBE-GNSS1		5.2.0 🕑 Install	Class Graphics	
STMicroelectronics X-CUBE-MEMS1		9.1.0 🖻 👻 Install	Add to favorites	
STMicroelectronics X-CUBE-NFC4		2.0.4 👜 👻 instail		
STMicroelectronics X-CUBE-NFC7		1.0.0 🖻 Install	Required languages	~
STMicroelectronics X-CUBE-SFXS2LP1		3.1.0 🖻 Install		
STMicroelectronics X-CUBE-SUBG2		4.2.0 🖻 👻 İnstali	C++ v99	
STMicroelectronics.X-CUBE-TOF1	-	3.0.0 🖻 👻 Install		~
STMicroelectronics X-CUBE-TOUCHGFX	<u>⊗</u> Ę	4.18.1 ~	Migration	~
✓ Graphics Application	$\odot$	4.18.1	Migrate to: Select a version	✓ Try
Application	$\odot$	4.18.1 TouchGFX ~		
emotas.I-CUBE-CANOPEN		1.1.2 🖻 😜 Install	Description	U
portGmbH.I-Cube-SoM-uGOAL		1.1.0 🗄 🗳 Install		
wolfSSL I-CUBE-wolfSSL		5.3.0 🖻 👻 İnstall	TouchGFX Generator	
FreeRTOS	G⊗			~
HAL Drivers	[+⊘		Documents	~
LIBJPEG	÷		License	
MBEDTLS	G.			
PDM2PCM	E+			

Now TouchGFX is disabled.

The next step would be for the user to generate the code so that he can add his features. Click Device Configuration Tool Code Generation button.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022



To add the Test task function, go to main.c file under the section core/src. Refer the below screen.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

• 🛛 🕼 🕹 • 🔪 🖡 🕨 🗉 🔳			100		×   🗹   🟮		Q : 🖻	1 40
Debug 🍋 Project Explorer 🗙 👘 🕻		S startup_stm	🖻 user_task.c 🗙 🔂 CANView.cpp	C *EEPROM1View	WREEPROM1Vi	»» <sub>17</sub>	-	• 0
🖻 🔩 🎖	8 64 */	Hean Tackt Stant/	void *pvParameters)					^
Al280_Project_Integration	66 {	USEL_IASKT_STALL	void pyparameters)					
🗱 Binaries		/* Enable this tes	t task by uncommenting "#define	TEST USER TASK1 CO	DE " */			
and Includes		efTEST_USER_TAS	C1_CODE					
Core		<pre>uint8_t ps_state;</pre>						
> 🥭 Inc	70	<pre>int8_t led_state;</pre>						
> 🧀 Maximatecc		while(1)						
V 🗁 Src		{						
> c freertos.c	74		WIT DONE, &ps state);					
> 🖻 main.c	75							
> 💽 shared_params.c	76	if(TRUE == ps_	state)					
> istm32h7xx_hal_msp.c	77 78	break;						
> stm32h7xx_hal_timebase_tim.c	79	vTaskDelay(500						
> c stm32h7xx_it.c	80	}	,					
> c syscalls.c	81	,						
> c sysmem.c		<pre>led_state = LED_ON</pre>						
> c system_stm32h7xx.c	83							
> c user_task.c	84 3	SET_DB(LED_RED_STA	<pre>FE, (uint8_t *)&amp;led_state);</pre>					
> 🅞 Startup	86 #end	if						
冯 Drivers	87							
Middlewares		while(1)						
冯 TouchGFX	89	{	ar					
> > App	90 91	vTaskDelay(100	ð);					
> 🗁 assets	91	}						
> 🕞 config	93 }							
> 🔁 generated	94 #end	if						
> 🕞 gui	95	and the second second						
> 🗁 simulator		USER_TASK2 == PS_E						
> 👝 target		User_Task2_Start(	void *pvParameters)					
Al280_Project_Integration.touchgfx	98 {							
application.config		/* Enable this tes	t task by uncommenting "#define	TEST USER TASK2 CO	DE "*/			
application.config.bak	101 #ifd	ef TEST USER TAS	(2 CODE					
ApplicationTemplate.touchqfx.part		uint8_t ps_state;						
> Debug		int8_t state;						Ų
Release	104							~
Al280_Project_Integration.ioc	e.							
Al280_Project_Integration.loc	Console	🗙 📳 Problems 🕥 E	xecutables 🛛 🙀 Debugger Console 📋 N	1emory 🛷 Search	🔲 🗶 🎉 🔜 👧	😼 📮 🖉 🚽 🖳 🕶	· 🖻 🕶 =	- 1
		Integration Debug (STM						

Now create the test task function. Refer the screen below for code snippet.

) • 🗐 🐚   🛞 • 🍕 • 🛍 🔯 • 😂 • 👩 •			77.44					Q 🔡 🖬 🔤
Project Explorer 🗙 📑 🖏 🏹 🖇 📟 🗖	C main.		LCD.c	c stm32h7xx_it.c	user_task.c	»2	- 0	🗄 Outline 🗙 🛞 Build Tar 😁 🗖
AI280_Project_Integration	247 248	<pre>/* USER CODE BEGIN F /* start timers, add</pre>					^	📄 🎼 😿 🖋 🖌 🕺
> 🐇 Binaries	248	/* USER CODE END RTC						warnings_removing(): void ^
> 🗊 Includes	249	7 OSEK CODE LIND KIT	S_TINCKS /					PS_TIM1_Init() : void
🗸 🥵 Core	251	/* USER CODE BEGIN F	TOS OUEUES	1				S VectorBase_Config(void) : void
> 🧁 Inc	252	/* add gueues, *						main(void) : int
✓ → Maximatecc	253	/* USER CODE END RTC	S_QUEUES */					SystemClock Config(void) : vo
> 🕞 Inc	254							<ul> <li>PeriphCommonClock_Config(\</li> </ul>
> 😝 Src	255	/* Create the thread						<ul> <li>S MX ADC1 Init(void) : void</li> </ul>
V 🕞 Src	256 257	<pre>/* creation of defau defaultTaskHandle =</pre>		Charlen Anderstein	MULL 04-6-147	and an and the second of		<ul> <li><sup>8</sup> MX_ADC3_Init(void) : void</li> </ul>
> c freertos.c	258	defaultlasknahole =	os inreadivew(	StartDefaultiask,	NULL, &deraulti	ask_attributes);		S MX_CRC Init(void) : void
> i main.c	259	/* creation of AITou	chGEXTask */	/ %	- 2			<ul> <li><sup>8</sup> MX_DMA2D Init(void) : void</li> </ul>
> c shared_params.c	260	AITouchGFXTaskHandle	= osThread	New AITestTaskFunc	ULL, &AITouch	GFXTask attributes	;);	
stm32h7xx_hal_msp.c	261					-	~	S MX_FDCAN1_Init(void) : void
stm32h7xx_hal_timebase_tim.c	262	/* USER CODE BEGIN F		*/				S MX_I2C3_Init(void) : void
	263	/* add threads,						S MX_IWDG1_Init(void) : void
> 🖻 stm32h7xx_it.c	264	/* add threads,	*/					S MX_QUADSPI_Init(void) : void
> 🛃 syscalls.c	265 266	/* USER CODE END RTO	C TUDEADC #	,				S MX_RTC_Init(void) : void
> 💽 sysmem.c	267	/ DOER CODE END KIL	J_TINCAUS /					S MX_SPI1_Init(void) : void
> c system_stm32h7xx.c	268	/* USER CODE BEGIN F	TOS EVENTS	1				S MX_SPI2_Init(void) : void
> 💼 user_task.c	269	/* add events, *	/ -					MX_TIM1_Init(void) : void
> 🧁 Startup	270	/* USER CODE END RTC	S EVENTS */					MX TIM2 Init(void) : void
> 😂 Drivers	271		1.000					S MX TIM3 Init(void) : void
> 🤔 Middlewares	272	/* Start scheduler *	1					S MX TIM4 Init(void) ; void
> 👝 Debug	273	osKernelStart();						S MX TIM7 Init(void) : void
> 🕞 Release	274 275	/* We should never g		and the second	a ha aba a bada	1=		<sup>S</sup> MX_TIM8_Init(void) : void
Al280_Project_Integration.ioc	275	/* We should never a /* Infinite loop */	et nere as c	CONTROL IS NOW TAKE	en by the schedu	ier -/		<ul> <li>S MX TIM12 Init(void) : void</li> </ul>
Al280_Project_Integration Debug.launch	277	/* USER CODE BEGIN W	HTLE */					
STM32H743IITX_FLASH_DEBUG.Id	278	while (1)	ALLER 7					S MX_TIM13_Init(void) : void
STM32H743IITX_FLASH_RELEASE.Id	279	{						MX_TIM14_Init(void) : void
STM32H743IITX_FLASH_RELEASE.ID	280	/* USER CODE END W	HILE */					MX_TIM16_Init(void) : void
STM32H743ITIX_KAM.Id	281							S MX_TIM17_Init(void) : void
	282	/* USER CODE BEGIN	13*/					S MX_DMA_Init(void) : void
	283 284	} /* USER CODE END 3 *	1					S MX_FMC_Init(void) : void
	284		1					S MX_GPIO_Init(void) : void
	286							StartDefaultTask(void*) : void
	2878	re e					~	<ul> <li>AlTestTaskFunc(void*) : void</li> </ul>

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

The user can add his code inside this test task.



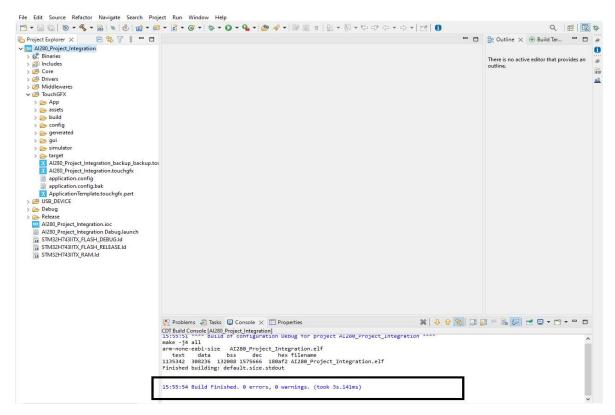
For example, we have added the below lines for LED functionality. User can call Get\_DL and Set\_DL functions inside AITestTaskFunc. Refer the below screen for code snippet.

Now click Project ===> Build All. Refer the below image.

	Open Project		🔊 🛷 📲 🕼 💷 🗤 🖓 🗢 🖗	• ♥ ♥ ♥ ♥ ♥   E	2 0		C	د 🖻	E.
roject Explorer 🗙 🛛 🔚 😘 🍞 🖇 🖷	<ul> <li>Close Project</li> </ul>				-	D 🔠 Outline	e 🗙 🛞 Build T	ar 🗢	E
Al280_Project_Integration	Build All	Ctrl+B							
Sinaries	Build Configurations	>					o active editor tha	t provides	an
Includes Core	Build Project					outline.			
Orivers	Build Working Set	>							
A Middlewares	Clean								
2 TouchGFX	Build Automatically								
> 🧀 App									
> 🗁 assets	Build Targets	>							
> 😂 build	Generate Report								
> 🧀 config	db Generate Code								
> 🧽 generated									
> 😕 gui > 😂 simulator	Properties								
> 🕞 target									
X Al280_Project_Integration_backup_ba	:kup.toi								
X Al280_Project_Integration.touchgfx									
application.config									
application.config.bak									
ApplicationTemplate.touchgfx.part									
CB_USB_DEVICE									
🗁 Debug 🍋 Release									
Al280_Project_Integration.ioc									
Al280_Project_Integration Debug.launch									
STM32H743IITX_FLASH_DEBUG.Id									
STM32H743IITX_FLASH_RELEASE.Id									
STM32H743IITX_RAM.Id									
	(*) Problems 27 To	ssiss 🖸 Console X	Properties 📑 🖸 🛪 📑		× all Static Sta	ick Analyzer		# 8 ==	
	জু Problems প্রায No consoles to displa		Properties 😁 🖙 🌱		× <u>all</u> Static Sta			e 8 📼 2ug - Sep	
			Properties 😁 🖸 👻 📑	Al280_Project_li 2022, 11:04:27	ntegration.elf - , AM				
			Properties 😁 🗆 👻 📩	Al280_Project_li 2022, 11:04:27 / Memory Regions	ntegration.elf - , AM Memory Details	/AI280_Project_I	ntegration/Del	oug - Sep	o 1
			Properties 🗂 🗆 🕶 📑	Al280_Project II 2022, 11:04:27 J Memory Regions Region	ntegration.elf - , AM Memory Details Start address	/AI280_Project_I	integration/Del	oug - Sep Free	5 1
			Properties 😁 🖸 👻 📑	Al280_Project_li 2022, 11:04:27 / Memory Regions	ntegration.elf - , AM Memory Details	/AI280_Project_I	integration/Del	oug - Sep	<b>1</b>
			Properties 🗂 🗆 🕶 📑	Al280_Project_II 2022, 11:04:27 J Memory Regions Region 5000 FLASH 1000 DTCMRAM	Memory Details Start address 0x08000000 0x20000000	/A1280_Project_1 End address 0x08200000 0x20020000	Size 2 MB 128 KB	Free 638.2 128 k	21 K
			Properties 😁 🖬 🕶 📑	AI280 Project JI 2022, 11:04:27 J Memory Regions Region Imp FLASH Imp TCMRAM	AM Memory Details Start address 0x08000000 0x2000000 0x24000000	/AI280_Project_I End address 0x08200000 0x20020000 0x24080000	Size 2 MB 128 KB 512 KB	Free 638.2 128 k 82.03	21 K 3
			Properties 🗂 🗆 🕶 📑	Al280_Project_II 2022, 11:04:27 J Memory Regions Region 5000 FLASH 1000 DTCMRAM	Memory Details Start address 0x08000000 0x20000000	/A1280_Project_1 End address 0x08200000 0x20020000	Size 2 MB 128 KB	Free 638.2 128 k	21 K 3
			Properties 😁 🖬 🛪 📑	AI280 Project JI 2022, 11:04:27 J Memory Regions Region Imp FLASH Imp TCMRAM	AM Memory Details Start address 0x08000000 0x2000000 0x24000000	/AI280_Project_I End address 0x08200000 0x20020000 0x24080000	Size 2 MB 128 KB 512 KB	Free 638.2 128 k 82.03	21 K 3 K

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Once the build is successful, you will get the console windows with "0 errors, 0 warnings." As shown below.



Now we will flash the binary using the ST JTAG, click debug  $\frac{1}{2}$  icon to flash the code. Refer the highlighted ( $\square$ ) part in the below screen.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date: Oct 17,	Oct 17, 2022

Project Explorer X       P 3       P 3       P 10         V Classo       V S       V S       P 10         V Classo       V S       V S       P 10         V S       Nations       V S       Nations         V S       Nations       V S
Image: Al200_Project_Integration.ioc         Image: Al200_Project_Integration_Debug.launch         Image: Strukt Strukture         Image: Strukture

During the flashing, the code you will get the below screen. Please click "Switch" button to continue.

AI280 - AI280_Project_Integration/Coro/Startup/s	lartup.stm32h743ity.s+S1M32CubeIDU - D	$\times$
Eile Edit Source Refactor Navigate Search Broj	ect Bun Window Help	
	· ● • • • • • • • • • • • • • • • • • •	-
Project Explorer 🕫 🔋 🕫 🖓 🖡 🖻 🕇	<sup>1</sup> E main.c B Ai280_Config.h B user_task.c B Watchdog Platformservice.c S startup_stm32h743iitx.s ≈	= 0
Al280 Project Integration	61 ldr sp, =_estack /* set stack pointer */	^
🔹 🗱 Binaries	62 63/* Call the clock system initialization function.*/	
Includes	64 bl Svsteminit	100
* @ Core	65	
🤋 😂 Inc	66/* Copy the data segment initializers from flash to SRAM */	
<ul> <li>E Maximatecc</li> </ul>	67 10 mm - France Switch X	
> les Sec	69 ld	
Startup     Onivers	70 md 2 This kind of launch is configured to open the Debug perspective when it suspends.	
B Middlewares		
Office Windoweres     Office Windoweres	This bebug perspective supports application debugging by providing views for displaying the	
> Bebug	74 1d debug stack, variables and breakpoints.	
> in Release	75 st 76 ad Switch to this perspective?	
Al280 Project Integration.ioc	70 au 77	
AI280 Project Integration Debug launch	78Loop	
STM32H743IITX_FLASH_DEBUG.Id	27 HU 5	
a STM32H743IITX_FLASH_RELEASE.id	80 cm 81 bc	
STM32H743IITX_RAM.Id	82 /* Z	
ButtonThread (in ButtonThread2)	83 ld+ r2, =_sbss	
Demo	84 ldr r4, =_ebss	~
Keypad testing	- C	2
maxAl280_gauge	😢 Problems 🖉 Tasks 🛽 Console 🛎 🗆 Properties 🗟 Build Analyzer 🛎 Static Stack Analyzer 🛷 Search 🕫 Progress 🛛 🖷 🗮 🐘 👔 🖉 🕎 🖤 🗖	9 · · · · · · · ·
	Al280 Project Integration Debug (STM32 Cortex-M C/C++ Application)	
	InitWhile : Enabled	^
	Waiting for debugger connection	
	Debugger connected	
	Waiting for debugger connection Debugger connected	10
	Debugger connected Maiting for debugger connection Activate Windows	
	Go to Settings to activate Windows	~
<	So to Settings to activate Windows	
	Writable Smart Insert 61:1:2239 Launching Al280_Project Jion Debug: (93)	.%) 🚥 👒
Type here to search	🚵 🕺 🐂 🚱 🖉 💘 🕮 🚾 🔣 🐧 🖓 🖓 🖓 👘 🕅	5

Dept: Software	User Manual Document	Rev No.Rev 3.0Date:Oct 17, 2	Rev 3.0
			Oct 17, 2022

Once the flashing is completed you will get the below screen.

AD80 - AD80 Project Integration/Core/Src/main.c - STM32CubeIDE			×
File Edit Source Refactor Navigate Search Project Run Window Help			
		Q. (m) [	EZ 🖬 🕬
Debug R OProject Explorer     E R 14 1 7 7 8 main. R 8 A280 Config.h 8 user task.c 8 Watchdog Platformservice.c 8 startup stm32h/43#bs.s			
Carden Debug [STM32 Contex-M C/C+ 178 #ifndef DEBUG     27 M280 Project Integration Debug [STM32 Contex-M C/C+ 178 #ifndef DEBUG     27 M280 Project Integrational [cores:0]     171 VectorSase_Config();     vectorSase_Config();     172 #endif     173			^
<pre># main() at main(:1/20x80096:98</pre>			
<pre>187 /* Configure the peripherals common clocks */ 188 PeriphCommonClock_Config(); 189 198 /* USER CODE BEGIN SysInit */ 191 192 /* USER CODE END SysInit */ 193 194 /* Initialize all configured peripherals */ 195 PK_GPTO_Init(); 196 WK CBT Configured</pre>			~
Console # Sprogress 7: Problems @ Executables @ Debugger Console @ Memory # Search			
Al280 Project, Integration Debug (STM32 Cortex M C/C++ Application)			
Download verified successfully			^
	ctivate Windows o to Settings to activate W	Vindows.	
Writable Smart Insert 178:1:4981			
📲 🔎 Type here to search 🦾 🖾 🙀 🐂 🚳 🗭 🥨 🕮 🔣 🔣 🔩 🤉	Se CX ING	16:33 /09/2022	-

Click Resume button (Highlighted) in the above image) to run the application. You will get the below Screen on the AI280 board.

In the AITestTaskFunc, we enabled LED. You can see the below screen shot where the AMBER LED is enabled.

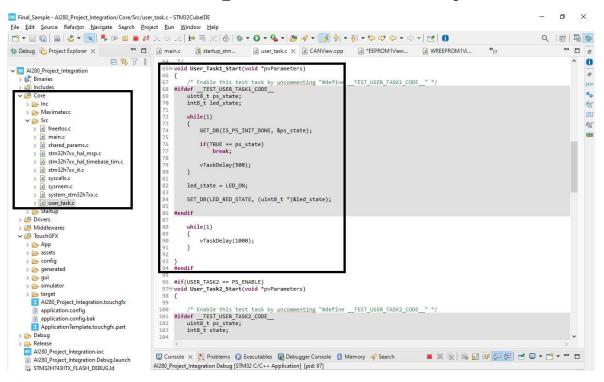


Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

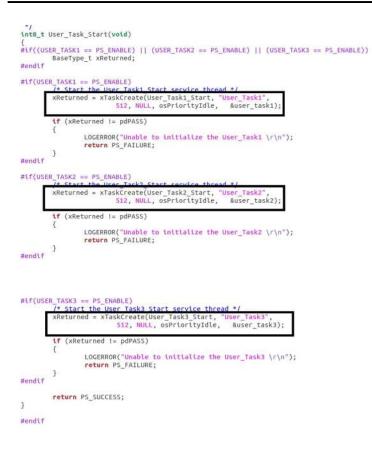
# 4.5 User Task Edit Details

This section elaborates how user can initialize their user task and call the APIs, Set\_DL and Get\_DL without using the TouchGFX. They can also perform any non-UI related tasks here. User can create the tasks as shown below. Below images shows that the user is creating three tasks. Refer to the below two images for the sample code snippet to create the tasks.

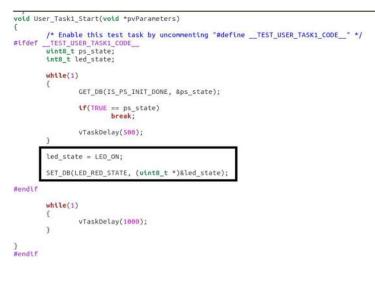
These tasks are created in their user task.c file as shown in the below image,



Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date: Oct 17, 2	Oct 17, 2022



User can call SET\_DB and GET\_DB APIs inside the created user tasks to access the required functionalities. Please see the below image for the code snippets.

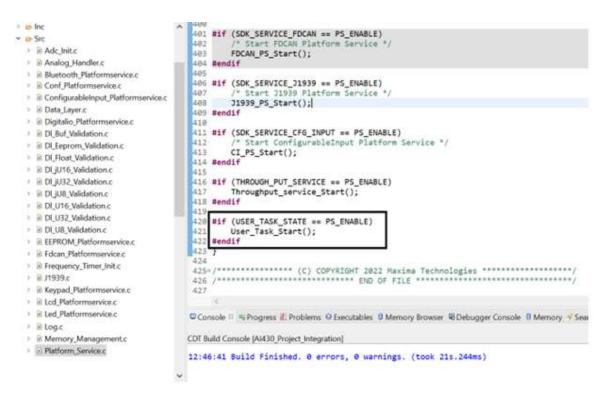


Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

```
#if(USER_TASK2 == PS_ENABLE)
void User_Task2_Start(void *pvParameters)
{
          /* Enable this test task by uncommenting "#define __TEST_USER_TASK2_CODE__" */
#ifdef __TEST_USER_TASK2_CODE_
uint8_t ps_state;
           int8_t state;
           while(1)
           {
                      GET_DB(IS_PS_INIT_DONE, &ps_state);
                      if(TRUE == ps_state)
                                break;
                      vTaskDelay(500);
           }
           state = KEY_BACKLIGHT_ON;
           Set_DL(KEYPAD_BACKLIGHT , (uint8_t *)&state);
#endif
           while(1)
           {
                     vTaskDelay(1000);
           }
#endif
HALLAND.
#lf(USER_TASK3 == PS_ENABLE)
void User_Task3_Start(void *pvParameters)
{
{
    /* Enable this test task by uncommenting "#define __TEST_USER_TASK3_CODE__" */
#ifdef __TEST_USER_TASK3_CODE__
uint8_t ps_state;
    int8_t led_state;
        while(1)
                GET_DB(IS_PS_INIT_DONE, &ps_state);
                tf(TRUE == ps_state)
    break;
                vTaskDelay(500);
        led_state = LED_ON;
        SET_DB(LED_AMB_STATE, (uint8_t *)&led_state);
#endif
        while(1)
        {
                vTaskDelay(1000);
        }
}
#endif
```

If user enables the USER\_TASK\_STATE in AI280\_Config.h file, User\_Task\_Start function gets called from Platform\_Service.c. Refer the below screen for code snippet.

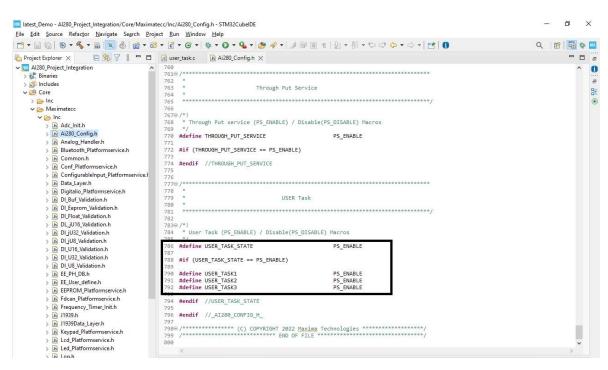
Dept: Software	User Manual Document	Rev No.	Rev 3.0
<b>F</b>		Date:	Oct 17, 2022



Before using the tasks, the user will need to initialize the tasks from the main.c. User can call there USERTASK1\_PS\_Start under platform service init as shown in the below image.

User need to create and enable the User Tasks in the configuration file. Refer the below screen for code snippet.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
F		Date:	Oct 17, 2022



The user can then compile and flash the same following the below procedure.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

• 🗐 💿 • 🔨 • 📾 🔍 🙆 👩		t Run Window H		▶ @ - ◎ - ◎ ◎ - ◎ - ◎ - ◎ - ◎ - ◎ - ◎ - ◎					0		172
		Open Project Close Project		· [四 《 • [四 回 ]] · [2] • (4) • (2 •		3   U -	-	0		-	E
roject Explorer × 📄 🔄 🍞 🖇 📟 Al280_Project_Integration	-	luild All	Ctrl+B			-	D 🗄 Outline	e 🗙 🖲 Build	d Tar	-	E
Binaries	a second	luild All	Ctri+b								
ncludes		-	<i>*</i>				outline.	o active editor t	that pro	vides a	an
😕 Core		luild Project					ounner				
😕 Drivers		luild Working Set	>								
Middlewares		lean									
2 TouchGFX	В	luild Automatically									
> 🗁 App > 🗁 assets	B	luild Targets	>								
> > build											
> 👝 config		Senerate Report									
> 👝 generated	@ G	Generate Code									
> 🧽 gui	P	roperties									
> 🗁 simulator				1							
> 👝 target											
Al280_Project_Integration_backup_back	up.toi										
Al280_Project_Integration.touchgfx application.config											
application.config.bak											
X ApplicationTemplate.touchgfx.part											
USB_DEVICE											
🔁 Debug											
😕 Release											
Al280_Project_Integration.ioc											
Al280_Project_Integration Debug.launch											
Al280_Project_Integration Debug.launch											
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld											
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld											
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld											
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld											
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld											
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld											
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld											
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld											
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld	ş	🕐 Problems 🖉 Task	s 📮 Consol	le X 🔲 Properties 😁 💷 🕶 🗖	Build Analyze	X 🚊 Static Sta	:k Analyzer		000		E
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld	1.12	👔 Problems 🧔 Task	and the second second second	ie X 🔲 Properties 😁 🕒 🕶 📑 🗖		a and a second data		Integration /			
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld	1.12	the second second second second second second second second second second second second second second second s	and the second second second	le X	Al280_Project_I	ntegration.elf - /		integration/D			
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld	1.12	the second second second second second second second second second second second second second second second s	and the second second second	le X 🔲 Properties 📑 🖬 ¥ 📑 🗖	Al280_Project_I 2022, 11:04:27	ntegration.elf - / AM		integration/D			
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld	1.12	the second second second second second second second second second second second second second second second s	and the second second second	ie X 🔲 Properties 🖻 🖵 😁 🗖 🗖	Al280_Project_I 2022, 11:04:27 Memory Regions	ntegration.elf - / AM Memory Details	Al280_Project_l		)ebug	Sep	1
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld	1.12	the second second second second second second second second second second second second second second second s	and the second second second	le X 🔲 Properties 📑 🖬 ¥ 📑 🗖	AI280_Project_I 2022, 11:04:27 Memory Regions Region	ntegration.elf - / AM Memory Details Start address	Al280_Project_li	Size	)ebug -	Sep Free	1
	1.12	the second second second second second second second second second second second second second second second s	and the second second second	le X 🗋 Properties 📑 🖬 💌 📑 🗖	Al280_Project_I 2022, 11:04:27 Memory Regions Region FLASH	ntegration.elf - / AM Memory Details Start address 0x08000000	Al280_Project_li	Size 2 MB	)ebug -	Sep Free 638.20	1
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld	1.12	the second second second second second second second second second second second second second second second s	and the second second second	ie X 🔲 Properties 🖻 🖢 * 🖻 * 🗖	Al280_Project_I 2022, 11:04:27 Memory Regions Region FLASH	Memory Details Start address 0x08000000 0x20000000	A1280_Project_I	Size 2 MB 128 KB	)ebug	- Sep Free 638.20 128 K	1
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld	1.12	the second second second second second second second second second second second second second second second s	and the second second second	le X 🗖 Properties 🖻 🖬 🕶 🗖 🕶 🗖	Al280_Project   2022, 11:04:27 / Memory Region Region ELASH TOTCMRAM RAM_D1	Memory Details Start address 0x0800000 0x2000000 0x24000000	End address 0x08200000 0x20020000 0x24080000	Size 2 MB 128 KB 512 KB	)ebug	Free 638.21 128 K 82.03	1
Al280_Project_Integration Debug.launch STM32H743IITX_FLASH_DEBUG.ld STM32H743IITX_FLASH_RELEASE.ld	1.12	the second second second second second second second second second second second second second second second s	and the second second second	ie X 🔲 Properties 📑 🖬 💌 📑 🗖	Al280_Project_I 2022, 11:04:27 Memory Regions Region FLASH	Memory Details Start address 0x08000000 0x20000000	A1280_Project_I	Size 2 MB 128 KB	)ebug	- Sep Free 638.20 128 K	1

Once the build is successful, you will get the console windows with "0 errors, 0 warnings." As shown below.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

• 🔙 🕼   🛞 • 🐔 • 🛗   🔌   🙆 • 🚳	• 🖸 • 🞯 •   株 • 🔘 • 💁 • 🥭 🛷 •   歸 圓 👖   狙 • 初 • や ヴ ク • ウ •	1 1 1		Q .	3   4
Project Explorer 🗙 📄 🗟 🍟 🖗 🗖		- 0	E Outline × @	Build Tar	
Al280_Project_Integration					
👬 Binaries			There is no active e	ditor that provi	dec an
ncludes			outline.	ditor that provi	aes an
🔁 Core					
Drivers					
Middlewares					
2 😂 TouchGFX					
> 🗁 App					
> 🗁 assets					
> 🔁 build					
> 🗁 config					
> 👝 generated					
> 👝 gui > 👝 simulator					
> 🧀 simulator > 🧀 target					
X Al280_Project_Integration_backup_backup.tou					
X Al280_Project_Integration.touchgfx					
application.config					
application.config.bak					
ApplicationTemplate.touchgfx.part					
B USB_DEVICE					
👝 Debug					
🛛 🗁 Release					
Al280_Project_Integration.ioc					
Al280_Project_Integration Debug.launch					
STM32H743IITX_FLASH_DEBUG.Id					
STM32H743IITX_FLASH_RELEASE.Id					
STM32H743IITX_RAM.Id					
	🕐 Problems 🧟 Tasks 📮 Console 🗙 🛄 Properties	×   & & & 🕄 🗔		🕒 • 😁 •	- c
	CDT Build Console (Al280_Project_Integration) 15:55:51 ***** Build of configuration Debug for project Al280_Project_Integrati				
	make -j4 all	on thir			
	arm-none-eabi-size AI280_Project_Integration.elf				
	text data bss dec hex-filename				
	1135342 308236 132088 1575666 180af2 AI280_Project_Integration.elf Finished building: default.size.stdout				
Г	15:55:54 Build Finished. 0 errors, 0 warnings. (took 3s.141ms)				
	assisted outro remained, o citors, o waintings, (cook ssiteting)				

Now we will flash the binary using the ST JTAG, click debug \* icon to flash the code. Refer the highlighted ( $\square$ ) part in the below screen.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Project Explorer ×     □     □     □       ✓ main Al280_Project_Integration       > main Minimum	Dutline X 🖲 Build Tar
<ul> <li>Dirkudes</li> <li>Dirkudes</li> <li>Dirkuts</li> <li>Dirkuts</li> <li>Middlewares</li> <li>App</li> <li>Dassets</li> <li>build</li> <li>config</li> <li>generated</li> <li>gui</li> <li>config</li> <li>generated</li> <li>gui</li> <li>config</li> <li>persented</li> <li>A1280_Project_Integration.backup.tox</li> <li>STM32H743IITX_FLASH_DEBUG.ld</li> <li>STM32H743IITX_RAM.ld</li> </ul>	There is no active editor that provides outline.

During the flashing, the code you will get the below screen. Please click "Switch" button to continue.

AI280 - AI280_Project_Integration/Core/Startup/	lartup_stm32h743ity.s = S1M32CubsIDU - D	$\times$
Eile Edit Source Refactor Navigate Search Pro-	ect Bun Window Help	
	· ● • • • • • • • • • • • • • • • • • •	-
Project Explorer 🕫 🗧 🎕 🖓 🖡 🖻 🛙	<sup>3</sup> E main.c B Ai280_Config.h B user_task.c B Watchdog,Platformservice.c S startup_stm32h743iitx.s ≈	= 0
▼ C Al280 Project Integration	61 ldr sp, =_estack /* set stack pointer */	^
🔹 🗱 Binaries	62 63/* Call the clock system initialization function.*/	
if Includes	64 bl Svsteminit	100
* @ Core	65	
🤋 😂 Inc	66/* Copy the data segment initializers from flash to SRAM */	
<ul> <li>E Maximatecc</li> </ul>	67 10 min - France Switch X	
> te Sec	69 ld	
Startup     Opivers	70 nd 2 This kind of launch is configured to open the Debug perspective when it suspends.	
B Middlewares		
OuchGFX	72 Copy 23 Copy debug stack, variables and breakpoints.	
> e Debug	74 1d	
> 🔊 Release	75 st 76 ad Switch to this perspective?	
Al280 Project Integration.ioc	77	
Al280_Project_Integration Debug.launch	78 Long Bemember my decision	
STM32H743IITX_FLASH_DEBUG.Id	72 #9	
STM32H743IITX_FLASH_RELEASE.ld	so cn Switch No	
III STM32H743IITX_RAM.Id	82/* Z	
ButtonThread (in ButtonThread2)	83 ldm m2, =_sbss 84 ldm m4, = ebss	
> 😂 Demo	34 10F F4, =_epss	~
Keypad testing		
> maxAl280_gauge	a Probens c lasss console - Chropenes sibolici Analyze - static static Analyze - search in Progress	100000
	AI280 Project Integration Debug (STM32 Cortex-M C/C++ Application) InitWhile : Enabled	
	Initwhile : Enabled	^
	Waiting for debugger connection	
	Debugger connected	
	Waiting for debugger connection Debugger connected	10
	Waiting for debugger connection Activate Windows	
	Go to Settings to activate Windows	~
٢	• «	>
	Writable Smart Insert 61:1:2239 Launching Al280_Project Jion Debug: (93	%) 🚥 👒
Type here to search	🚲 🖬 🐂 🜀 💉 💘 🚝 🚾 🔣 🛝 🖓 🍗 🦧 🗤 11632	5

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Once the flashing is completed you will get the below screen.

AI280 - AI280 Project Integration/Core/Src/main.c - STM3	2CubeIDE	- 5	×
File Edit Source Refactor Navigate Search Project Run	Window Help		
T + U U K & + K I W M X & . (F = 1)	8 8 • 8 • 8 • 8 4 • 8 8 • 8 • 8 • 8 • 8	Q.[雨]	E11 🛄 (10)
Debug R Project Explorer     E is in 1 * 0	iš main,c ≈ iš Ai280 Config.h iš user task.c iš Watchdog Platformservice.c iš startup stm32h743itx.s		
<ul> <li>AI280 Project Integration Debug [STM32 Cortex-M C/C</li> <li>AI280_Project_Integration.elf [cores: 0]</li> </ul>	171 VectorBase_Config();		^
<ul> <li>         Thread #1 [main] 1 [core: 0] [Suspended : Breakpo</li></ul>	174 175 /* NCU Configuration*/		
I ST-LINK (ST-LINK GDB server)	176 177 /* Reset of all peripherals, Initializes the Flash interFace and the <u>Systick</u> . */ 178 HAL_Init(); 178 HAL_Init();		
	179 180 /* USER CODE BEGIN Init */ 181 /* USER CODE DEGIN Init */		
	182 /* USER CODE END Init */ 183 /* Configure the system clock */		
	<pre>185 SystemClock_Config(); 186</pre>		
	187 /* Configure the peripherals common clocks */ 188 PeriphCommonClock_Config();		
	189		
	198 /* USER CODE BEGIN SysInit */ 191		
	192 /* USER CODE END SysInit */		
	194 /* Initialize all configured peripherals */		
	195 MX_GPI0_Init(); 195 MX_CRC_Init():		~
	14m DX LAU DXTLL.		3
	🛡 Console 🗉 🖷 Progress 🖏 Problems 🔍 Executables 🖷 Debugger Console 🛛 Memory 🛷 Search 🛛 🗮 🗮 👘 📦 🕬		• = 0
	Al280_Project_Integration Debug (STM32 Cortex-M C/C++ Application)		
	Download verified successfully		^
	Activate Window Go to Settings to activ		~
(	Writable Smart Insert 178 : 1 : 4981		2
P Type here to search		G 16:33 14/09/2022	<b>R</b> 4

Click Resume is button (Highlighted in the above image) to run the application. You will get the below Screen on the AI280 board.

Once the device powers up with this build user can see the output on the board. By default, the MAXAI280 LED state is OFF but, in the user, task3 that we created and flashed we have changed the LED state as ON.

Hence if we power up the board with this build, the user can see that LED is ON as shown below which confirms that the user task 3 has executed successfully.

Hence the user can add any non-UI based functionalities in these tasks and execute them in the background.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
*		Date:	Oct 17, 2022

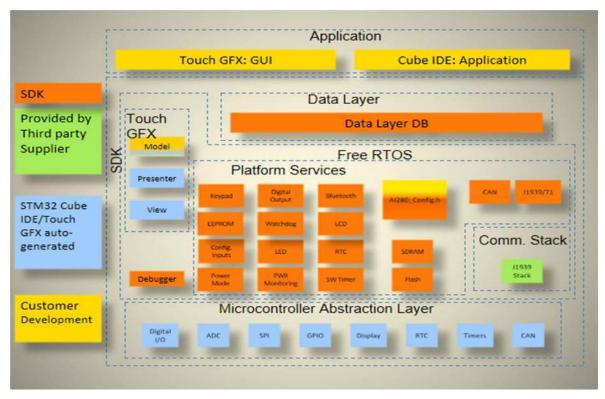


Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 5 SDK Overview

# 5.1 SDK Architecture

The AI280 SDK is built with the below four layers which are well organized to give the application the flexibility to be written independently with minimum knowledge regarding in the internal functionality of the underlying hardware.



A quick overview of the layers is described below,

#### Application :

Blank Touch GFX will be provided where USER will be able to create the GUI layout using Touch GFX or USER developed widgets, Hardware configuration of the Touch GFX project would be predefined in the SDK. The user can create the graphical elements and link then with the SDK modules to achieve the desired results. The user modified code resides here.

#### Data Layer DB:

Data Layer DB is the interface between USER application and the SDK Platform services. It acts as an intermediate layer and used for communication between user application and platform services.

Date: Oct 17, 2022	Dept: Software	User Manual Document	Rev No.	Rev 3.0
			Date:	Oct 17, 2022

Data Layer DB consist of a collection of RAM variables containing the data of the platform services, this data shall be updated with latest data from each platform service iteration/event. The Data Layer DB will also work as a channel to input data from the application to the Platform services.

#### Platform Services :

Platform Services will work as an interface between Data Layer DB and platform driver. It will create and manage tasks for hardware peripherals based on user configuration/application requirements. These created tasks will run in RTOS. Data requested from user application will be obtained by Platform services from platform driver. After receiving data, Platform services will push that information to Data Layer DB. Then, user application can fetch requested data from the Data Layer DB.

#### **Platform Drivers**

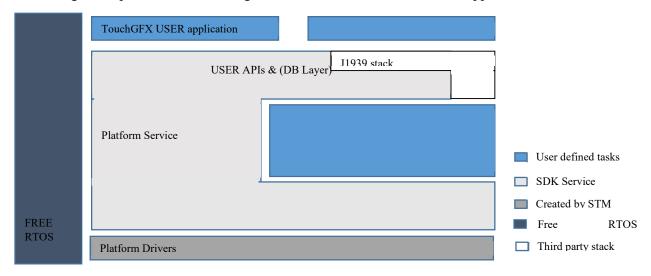
:

Platform Drivers will be used for accessing and controlling the hardware. Platform drivers will configure the hardware based on user configuration/application requirements. It will receive the relevant data needed by user application. Data received by platform driver will be sent to platform services.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date: Oct 17, 202	Oct 17, 2022

# 5.2 SDK Interfaces

The SDK adaptation software provides an interface between the USER tasks and platform driver layer on AI280 hardware platform. This design provides the easy to include / exclude design for the SDK modules/drivers using the configuration file (.h) in the final firmware application. And the USER can easily integrate the TouchGFX UI into the SDK and use it on the AI280 hardware platform. Using this design, the USER can easily focus on the design of the end application. The below diagram depicts the overall design architecture of the final firmware application.



The user interacts with the SDK for the below functionalities,

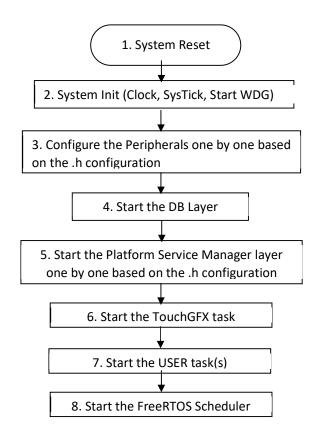
- 1) Enable and disable the modules via the configuration file.
- 2) Provide default configuration for the properties of the modules as per their requirements.
- 3) Access Data Layer Data Base to get/set individual properties of the modules.

The user is free to enable any of the services or modules if needed to improve memory constraints. So, the first step is to enable the desired services in the configuration file and then configure the properties for each manager. The managers are directly connected to the OS and works automatically on the background, so the user does not need to worry about the usage or the error management. In the below sections a detailed description is provided on how each module of the SDK can be accessed by the user for the full filling their requirements.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
	Oser manual Document	Date:	Oct 17, 2022

#### 5.3 SDK Boot flow

The diagram below depicts the SDK boot flow for the MAXAI280 platform.



Date: Oct 17, 2022	Dept: Software User Manual Document		Rev No.	Rev 3.0
				Oct 17, 2022

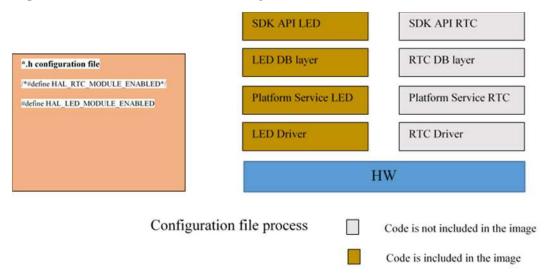
# 6 Application and SDK Interaction

The user can interact with the SDK to configure the individual modules of the SDK. This can happen either during power up configuration or during the run time configuration.

# 6.1 SDK Module default configuration

The user can configure the SDK by enabling and disabling individual modules. If a user enables or disables a particular module the entire stack is disabled for the particular module as shown in the below diagram.

For Example, If the USER disables the RTC driver, the RTC related (SDK USER APIs, RTC DB Layer, RTC Platform service and RTC Driver) codes will not be included in the final firmware image. That is, each device driver related SDK APIs, DB layer service, platform service and driver will be blocked with preprocessor MACROs (e.g., HAL\_RTC\_MODULE\_ENABLED). This implementation will reduce the final image size.



**For example:** - If user wants to enable the keypad module in the current build, user has to configure the below mentioned variable as PS\_ENABLE.

#### #define SDK\_SERVICE\_KEYPAD

#### PS\_ENABLE

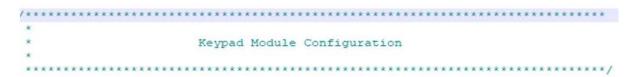
The user can find the configuration file in the source code in the below-mentioned path. Users can configure variables for any modules based on the requirements in the AI280\_Config.h.

#### AI280\_Project\Core\Maximatecc\Inc\AI280\_Config.h

Each module in the configuration file is differentiated with Headers/comments and users can easily find the SDK modules they are looking for.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

For Example:- Keypad Module configuration



As shown in the above pic, the keypad module configurations are listed in the header file after the above comments. You can find similar comments for each module section in the configuration file.

The user can configure certain parameters per module which will impact the default configuration of the individual modules. This can be done by modifying the configuration file which is used by the SDK to configure the individual modules during the power up sequence. Once configured the modules continue the same configuration until it is changed by the user.

**For Example:** - The below parameter configures the keypad backlight. It can be configured as ON / OFF and when the device powers up the SDK Reads this configuration file and updates the keypad backlight accordingly. In the MAXAI280 the default configuration for this parameter is true and hence the keypad backlight is always ON after device powers up unless the application turns it of during runtime.

```
/*!
 * MACRO Supported
 *
 * Keypad <u>backlight</u> configuration state
 * 1: KEY_BACKLIGHT_ON /
 * 0: KEY_BACKLIGHT_OFF
 */
```

```
#define KEYPAD_BACKLIGHT_CFG_STATE
```

KEY\_BACKLIGHT\_ON

When the user modifies any configuration in the config.h file, the user will have to re - compile the source code and flash the updated binaries to the device and verify the changes.

# 6.2 Run Time Configuration:

The user can modify certain parameters per module during the run time to interact with the module and to execute their desired functionality. This can be achieved by using the Datalayer database API's to read/write into the DB entries for each module.

Data Layer DB will collect the data from the platform service / platform layers and update the data into the proper variable.

Data Layer DB will be accessed by using GET/SET APIs from the application. If the application needs any platform related data, it uses the GET/SET API of the DB layer with the proper platform field name/id.

Dept: Software	User Manual Document	Rev No. R	Rev 3.0
		Date:	Oct 17, 2022

The below platform service / platform layers are linked with the DB layer.

- ➢ Keypad
- Digital Output
- Configurable Inputs
- Power Mode
- > LED
- Power Monitoring
- Bluetooth
- ≻ RTC
- > SW Timer
- ➢ EEPROM
- > Watchdog
- > LCD
- > Flash
- > CAN
- ➢ J1939

The Data Layer DB will interact with the Platform service through platform service SDK GET/SET APIs.

#### 6.3 DBLayer USER APIs

The USER shall access the DB layer field through the below set and get functions.

#### 6.4 Function Name: GET\_DL

S no	API Syntax	Parameter	Return value
1.	<b>GET_DL()</b> The user can use this API to retrieve the value of the data from the database. The field id is defined in the database.h file which identifies the data field we are interested in.	uint16_t dl_index , uint8_t *buf Value : We have to pass the Data ID for the data field we are looking to retrieve the data and then pass a buffer where the data to be written will be stored when the function call returns to the application.	0: Success 1: Failure

Below is the snippet of the description of the function,

/\*\* @brief Set\_DL

```
*
   This function will Set the data to the platform service
 *
 *
   @param dl index[IN] : DB index value
 *
           buf[IN]
                         : input buffer
 *
   @return 0 : FAILURE
 *
 *
            1 : SUCCESS
 *
           -3 : NULL_POINTER
 *
 */
int8_t Set_DL(uint16_t dl_index, uint8_t *buf)
```

#### 6.5 Function Name: SET\_DL

S no	API Syntax	Parameter	Return value
1.	SET_DL()	uint16_t dl_index , uint8_t *buf	0: Success
	The user can use this API to store the value of the data from the database. The field id is defined in the database.h file which identifies the data field we are interested in.	Value: We have to pass the index ID for the data field we are looking to retrieve the data and then pass a buffer where the data to be written will be stored when the function call returns to the application.	1: Failure

Below is the snippet of the description of the function,

```
/** @brief Get DL
*
*
   This function will get the data from the platform service
*
*
   @param dl index[IN] : DB index value
*
           buf[IN]
                    : input buffer
*
   @return 0 : FAILURE
*
 *
            1 : SUCCESS
 *
           -3 : NULL POINTER
*
*/
```

```
int8_t Get_DL(uint16_t dl_index, uint8_t *buf)
For Example: -
```

If we want to get the current status of the keypad backlight and then toggle it, we can do so by using the below code snippet,

```
/* Get the current backlight status */
```

```
If (Get_DL(KEYPAD_BACKLIGHT , &state) == SUCCESS)
{
    /* If current state is ON, set it OFF */
    If ( state == KEY_BACKLIGHT_ON )
    {
        State = KEY_BACKLIGHT_OFF;
        Set_DL(KEYPAD_BACKLIGHT , &state);
    }
    /* If current state is OFF, set it ON */
    Else
    {
        State = KEY_BACKLIGHT_ON;
        Set_DL(KEYPAD_BACKLIGHT , &state);
    }
}
```

Dept: Software	ept: Software User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 7 SDK Modules

Described below are the functionalities supported by each SDK module which can be used by the application developer to full fill their requirements.

# 7.1 Keypad module

The User would be able to use the below functionalities of the keypad module via the DB variables and configuration file.

#### 7.1.1 Keypad module Enable/Disable

The SDK provides the user the ability to enable/disable the Keypad functionality by modifying the default configuration file. Please see section 7.1.7 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_ KEYPAD	PS_ENABLE PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE: -</li> <li>Enables the Keypad module in the SDK</li> <li>PS_DISABLE: -</li> <li>Disables the Keypad module in the SDK</li> </ul>

# 7.1.2 Keypad BackLight ON/OFF

The AI280 SDK supports default configuration of the Keypad backlight and this can be done by modifying the below parameter in the configuration file. Please see section 7.1.7 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	KEYPAD_BACK LIGHT_CFG_STA TE	1: KEY_BACKLIG HT_ON / 0: KEY_BACKLIG HT_OFF	KEY_BACKLIG HT_ON	User can configure the default state of the keypad Backlight to ON/OFF using this variable.

During runtime the user can read and modify the keypad backlight by reading and writing to the below DB variables.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Field ID	Data Type	Permission	Size	Description	Comments
KEYPAD_BA CKLIGHT	DBu8	READ/WRI TE	1	ON/OFF	This field is used to turn ON/OFF the Keypad Backlight.

The sample code for set/get the Backlight using Key#1 is as below:

```
uint8_t state;
if (KEY1_SHORT_PRESS == val)
{
    state = KEY_BACKLIGHT_ON;
    Set_DL(KEYPAD_BACKLIGHT, &state);
}
/* Get the backlight state */
Get_DL(KEYPAD_BACKLIGHT, &state);
```

#### 7.1.3 Keypad Time Out Configuration

The AI280 SDK user can configure the timeout value of keypress to differentiate between short press and long press. Short Press can be configured in the range (>10ms && <255ms). If the key is pressed longer than the short press timeout it would be considered as long press. This default configuration can be done in the AI280\_config.h. Please see section 7.1.7 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SHORT_PRESS_T IMEOUT	Short Press (> 10 && < 255) Long press (> short press time)	10ms (Recommended Value)	The user can configure the timeout value of keypress to differentiate between short press and long press.

# 7.1.4 Keypad task Priority

The AI280 SDK supports the below task priorities and the user can modify the task priority for the keypad module in the configuration file. Please see section 7.1.7 for sample configuration.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Sr. No	Variables	Options	Default State	Description
1	PS_KEYPAD_TA SK_PRIORITY	osPriorityNone, osPriorityIdle, osPriorityLow, osPriorityLow1, osPriorityISR, osPriorityError, osPriorityReserved	osPriorityIdle (Recommended Value)	User can select any one of the priorities based on the application requirement

# 7.1.5 Keypad Keys Enable/Disable

The AI280 SDK supports four keys and the user can enable/disable each of the keys using the default configuration file.

Sr. No	Variables	Options	Default State	Description
1	CONF_KEYPAD_ 01_STATE	PS_ENABLE PS DISABLE	PS_ENABLE	<b>PS_ENABLE:</b> Enable the HW Key01 in the Keypad SDK platform service <b>PS_DISABLE:</b> Disable the HW Key01 in the Keypad SDK platform service
2	CONF_KEYPAD_ 02_STATE	PS_ENABLE PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE: Enable the HW Key02 in the Keypad SDK platform service</li> <li>PS_DISABLE: Disable the HW Key02 in the Keypad SDK platform service</li> </ul>
3	CONF_KEYPAD_ 03_STATE	PS_ENABLE PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE: Enable the HW Key03 in the Keypad SDK platform service</li> <li>PS_DISABLE: Disable the HW Key03 in the Keypad SDK platform service</li> </ul>
4	CONF_KEYPAD_ 04_STATE	PS_ENABLE PS_DISABLE	PS_ENABLE	<ul><li>PS_ENABLE: Enable the HW Key04 in the Keypad SDK platform service</li><li>PS_DISABLE: Disable the HW Key04 in the Keypad SDK platform</li></ul>

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

service	

# 7.1.6 Keypad Keys read status

The AI280 SDK user can then read the Key status variables to know if the keys are active or inactive. This DB entry has to be read first for receiving the keypress event. If the KEY\_STATUS\_01 is KP\_ACTIVE then the USER has to read the KEY\_PRESS\_01 DB variable to check the state of the key press.

Field ID	Data Type	Permission	Size	Description	Comments
KEY_STATUS_01	DBu8	READ	1	KP_ACTIVE/KP_IN ACTIVE	This field is used to Read the status of the Key#1.
KEY_STATUS_02	DBu8	READ	1	KP_ACTIVE/KP_IN ACTIVE	This field is used to Read the status of the Key#2
KEY_STATUS_03	DBu8	READ	1	KP_ACTIVE/KP_IN ACTIVE	This field is used to Read the status of the Key3
KEY_STATUS_04	DBu8	READ	1	KP_ACTIVE/KP_IN ACTIVE	This field is used to Read the status of the Key#4
KEY_PRESS_01	DBu8	READ/WRI TE	1	INACTIVE/SHORT_ PRESS/LONG_PRE SS/CONTINUOUS_ PRESSED	This field detects the type of KeyPress for Key#1.
KEY_PRESS_02	DBu8	READ/WRI TE	1	INACTIVE/SHORT PRESS/LONG_PRE SS/CONTINUOUS_ PRESSED	This field detects the type of KeyPress for Key#2.
KEY_PRESS_03	DBu8	READ/WRI TE	1	INACTIVE/SHORT_ PRESS/LONG_PRE SS/CONTINUOUS_ PRESSED	This field detects the type of KeyPress for Key#3.
KEY_PRESS_04	DBu8	READ/WRI TE	1	INACTIVE/SHORT_ PRESS/LONG_PRE SS/CONTINUOUS_ PRESSED	This field detects the type of KeyPress for Key#4.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

The user can read the key state (short press/ long press/ continuous / inactive) by continuously monitoring the below DB variables. Once the USER gets any one of the Keypress events (SHORT\_PRESS/LONG\_PRESS), the USER has to ACK the key press (KEY\_PRESS\_01) with the value 1. For CONTINUOUS state is not necessary to acknowledge the state.

Below is a code snippet for the keypress read and acknowledge usecase.

```
uint8 t key status = 0;
#if(CONF_KEYPAD_01_STATE == PS_ENABLE)
   /* Read the Key Status from the DB */
   res = Get_DL(KEY_STATUS_01, &key_status);
   /* Proceed only if the Key # 1 is active */
   if (KEY INACTIVE != key status)
   {
         res = Get_DL(KEY_PRESS_01, &key_status);
         if (SHORT_PRESSED == key_status)
         {
               key status = 1;
               /* ACK the Keypress */
               res = Set_DL(KEY_PRESS_01, &key_status);
         }
         else
         if (LONG_PRESSED == key_status)
         {
               key_status = 1;
               /* ACK the Keypress */
               res = Set_DL(KEY_PRESS_01, &key_status);
               modelListener->keyUpdate(KEY1_LONG_PRESS);
         }
         else
         if (CONTINUOUS_PRESSED == key_status)
         {
               modelListener->keyUpdate(KEY1_LONG_PRESS);
         }
   }
#endif
            //KEYPAD_01
```

#### 7.1.7 Keypad sample Configuration

```
* Keypad Platform service Enable (PS_ENABLE) / Disable (PS_DISABLE) Macros
*/
#define SDK_SERVICE_KEYPAD PS_ENABLE
#if (SDK_SERVICE_KEYPAD == PS_ENABLE)
/*!
* Keypad Task Priority
* osPriorityNone = 0,
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

<pre>* osPriorityIdle</pre>	= 1,	
* osPriorityLow	= 8,	
<pre>* osPriorityLow1</pre>	= 8+1,	
۔ و و		ر ر
د ر		ر ر
* osPriorityISR	= 56,	
* osPriorityError	= -1,	
<pre>* osPriorityReserved</pre>	= 0x7FFFFFFF	
*/		
#define PS_KPD_TASK_PRIOR	ΓTY	osPriorityIdle
/*!		
* MACRO Supported		
*		
* Keypad <u>backlight</u> config	guration state	
* 1: KEY_BACKLIGHT_ON /		
* 0: KEY_BACKLIGHT_OFF		
*/		
#define KEYPAD_BACKLIGHT_C	CFG_STATE	KEY_BACKLIGHT_ON
/*!		
* SHORT Press timeout (Mi	illisecond)	
*/		
#define SHORT_PRESS_TIMEOU	JT 10	
/*!		
* MACOR Supported		
*		
* Keypad configuration st	tate	
* 1: PS_ENABLE /		
* 0: PS_DISABLE		
*/		
#define CONF_KEYPAD_01_STA	ATE	PS_ENABLE
#define CONF_KEYPAD_02_STA	ATE	PS_ENABLE
#define CONF_KEYPAD_03_STA	ATE	PS_ENABLE
#define CONF_KEYPAD_04_STA	ATE	PS_ENABLE

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 7.2 Digital Output Module

The User would be able to use the below functionalities of the digital output module via the DB variables and configuration file.

#### 7.2.1 Digital Output module Enable/Disable

The SDK provides the user the ability to enable/disable the Digital Output functionality by modifying the default configuration file. Please see section 7.2.6 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_D IGITAL_OUTPUT	PS_ENABLE PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE: -</li> <li>Enables the Digital Output module in the SDK</li> <li>PS_DISABLE: -</li> <li>Disables the Digital Output module in the SDK</li> </ul>

# 7.2.2 Digital Output Configuration

The AI280 SDK supports default configuration of the digital output status and this can be done by modifying the below parameter in the configuration file. Please see section 7.2.6 for sample Configuration.

Sr. No	Variables	Options	Default State	Description
1	DIGITAL_OUTPU T_CFG_01	CONF_OPEN_D RIVE_DRIVER/ CONF_LOW_SI DE_DRIVER/ CONF_HIGH_SI DE_DRIVER	CONF_OPEN_D RIVE_DRIVER	User can select the Digital output 01 configuration as open drive driver when the digital output status is OFF User can select the Digital output configuration as low side driver. User can select the Digital output configuration as high side driver.
2	DIGITAL_OUTPU T_CFG_02	CONF_OPEN_D RIVE_DRIVER/ CONF_LOW_SI DE_DRIVER/ CONF_HIGH_SI DE_DRIVER	CONF_OPEN_D RIVE_DRIVER	User can select the Digital output 02 configuration as open drive driver when the digital output status is OFF User can select the Digital output configuration as low side driver. User can select the Digital output configuration as high side driver.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

3	DIGITAL_OUTPU T_CFG_03	CONF_OPEN_D RIVE_DRIVER/ CONF_LOW_SI DE_DRIVER/ CONF_HIGH_SI DE_DRIVER	CONF_OPEN_D RIVE_DRIVER	User can select the Digital output 03 configuration as open drive driver when the digital output status is OFF User can select the Digital output configuration as low side driver. User can select the Digital output configuration as high side driver.
4	DIGITAL_OUTPU T_CFG_04	CONF_OPEN_D RIVE_DRIVER/ CONF_LOW_SI DE_DRIVER/ CONF_HIGH_SI DE_DRIVER	CONF_OPEN_D RIVE_DRIVER	User can select the Digital output 04 configuration as open drive driver when the digital output status is OFF User can select the Digital output configuration as low side driver. User can select the Digital output configuration as high side driver.

The user can do the same configuration during the runtime via the DB variables and configuration file as shown below,

Field ID	Data Type	Permission	Size	Description	Comments
CF_DIGITAL_ OUTPUT_01_C FG	DBu8	READ/WRI TE	1	OPEN_DRIVE / HIGH_SIDE_DRIVE R / LOW_SIDE_DRIVE R	This field is used to configure the digital output 01 as high side, low side or open drive. The field is also used to enable/ disable the Digital Output. The status of the field can also be read back once enabled.
CF_DIGITAL_ OUTPUT_02_C FG	DBu8	READ/WRI TE	1	OPEN_DRIVE / HIGH_SIDE_DRIVE R / LOW_SIDE_DRIVE R	This field is used to configure the digital output 02 as high side, low side or open drive. The field is also used to enable/ disable the Digital Output. The status of the field can also be read back once enabled.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
	User Manual Document	Date:	Oct 17, 2022

CF_DIGITAL_ OUTPUT_03_C FG	DBu8	READ/WRI TE	1	OPEN_DRIVE / HIGH_SIDE_DRIVE R / LOW_SIDE_DRIVE R	This field is used to configure the digital output 03 as high side, low side or open drive. The field is also used to enable/ disable the Digital Output. The status of the field can also be read back once enabled.
CF_DIGITAL_ OUTPUT_04_C FG	DBu8	READ/WRI TE	1	OPEN_DRIVE / HIGH_SIDE_DRIVE R / LOW_SIDE_DRIVE R	This field is used to configure the digital output 04 as high side, low side or open drive. The field is also used to enable/ disable the Digital Output. The status of the field can also be read back once enabled.

Below is the sample code for accessing the Digital output configuration DB variables.

```
uint8_t state;
/* Read the Digital Output configuration */
Get_DL(DIGITAL_OUTPUT_01_CFG, &state);
if(state == CONF_LOW_SIDE_DRIVER)
{
      state = CONF_HIGH_SIDE_DRIVER;
      /* Set the High side Digital Output */
      Set_DL(DIGITAL_OUTPUT_01_CFG, &state);
}
Get_DL(DIGITAL_OUTPUT_01_CFG, &state);
if(state == CONF_HIGH_SIDE_DRIVER)
{
      state = CONF_LOW_SIDE_DRIVER;
      /* Set the Low side Digital Output */
      Set_DL(DIGITAL_OUTPUT_01_CFG, &state);
}
```

#### 7.2.3 Digital Output ON/OFF

The AI280 SDK user can turn ON / OFF the digital output during runtime. To do so he can use the below DB variables.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Field ID	Data Type	Permission	Size	Description	Comments
DIGITAL_OUT PUT_01_STAT E	DBu8	READ/WRI TE	1	CONF_DIGITAL_O UTPUT_ON/CONF_ DIGITAL_OUTPUT _OFF	This field is used to turn ON or OFF the digital output 01. The status of the field can also be read back once to get the current status of the pins.
DIGITAL_OUT PUT_02_STAT E	DBu8	READ/WRI TE	1	CONF_DIGITAL_O UTPUT_ON/CONF_ DIGITAL_OUTPUT _OFF	This field is used to turn ON or OFF the digital output 02. The status of the field can also be read back once to get the current status of the pins.
DIGITAL_OUT PUT_03_STAT E	DBu8	READ/WRI TE	1	CONF_DIGITAL_O UTPUT_ON/CONF_ DIGITAL_OUTPUT _OFF	This field is used to turn ON or OFF the digital output 03. The status of the field can also be read back once to get the current status of the pins.
DIGITAL_OUT PUT_04_STAT E	DBu8	READ/WRI TE	1	CONF_DIGITAL_O UTPUT_ON/CONF_ DIGITAL_OUTPUT _OFF	This field is used to turn ON or OFF the digital output 04. The status of the field can also be read back once to get the current status of the pins.

Below is the sample code for turning ON/OFF the Digital output DB variable.

```
state = CONF_DIGITAL_OUTPUT_ON;
/* Set the open drive Digital Output */
Set_DL(DIGITAL_OUTPUT_01_STATE, &state);
state = CONF_DIGITAL_OUTPUT_OFF;
/* Set the open drive Digital Output */
Set_DL(DIGITAL_OUTPUT_01_STATE, &state);
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Soltware		Date:	Oct 17, 2022

#### 7.2.4 Digital Output Time Out Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware and update it in the DB so that when the user reads the DB, he will receive the latest updated data if any or perform any other routine tasks. This default configuration can be done in the AI280\_config.h. Please see section 7.2.6 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_DIO_TASK_TI MEOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommended Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the values in the Database.

#### 7.2.5 Digital Output task Priority

The AI280 SDK supports the below task priorities and the user can modify the task priority for the digital output module in the configuration file. Please see section 7.2.6 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_DIO_TASK_P RIORITY	osPriorityNone, osPriorityIdle, osPriorityLow, osPriorityLow1, osPriorityISR, osPriorityError, osPriorityReserve d	osPriorityIdle (Recommended Value)	User can select any one of the priorities based on the application requirement

# 7.2.6 Digital Output Sample Configuration

```
/*!
 * DIO Platform service Enable (PS_ENABLE) / Disable (PS_DISABLE) Macros
 */
#define SDK_SERVICE_DIGITAL_OUTPUT PS_ENABLE
#if (SDK_SERVICE_DIGITAL_OUTPUT == PS_ENABLE)
/*!
 * DIO Task Periodicity 100ms
 */
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Soltware		Date:	Oct 17, 2022

```
#define PS_DIO_TASK_TIMEOUT
                                                    100
/*!
* DIO Task Priority
* osPriorityNone
                       = 0,
* osPriorityIdle
                       = 1,
 * osPriorityLow
                       = 8,
 * osPriorityLow1
                     = 8+1,
           ر ر
                                         ر ر
*
           ر ر
                                         رر
* osPriorityISR
                     = 56,
* osPriorityReserved = 0x7FFFFFF
 */
#define PS_DIO_TASK_PRIORITY
                                                         osPriorityIdle
/*!
 * Select the DIGITAL_OUTPUT_CFG 00 : CONF_LOW_SIDE_DRIVER
*
                               01 : CONF_HIGH_SIDE_DRIVER
 *
                               02 : CONF_OPEN_DRIVE_DRIVER
 */
#define DIGITAL OUTPUT CFG 01
                                                 CONF OPEN DRIVE DRIVER
#define DIGITAL_OUTPUT_CFG_02
                                                 CONF_LOW_SIDE_DRIVER
#define DIGITAL_OUTPUT_CFG_03
                                                 CONF_HIGH_SIDE_DRIVER
#define DIGITAL_OUTPUT_CFG_04
                                                 CONF_HIGH_SIDE_DRIVER
```

**#endif** //SDK\_SERVICE\_DIGITAL\_OUTPUT

Dept: Software	User Manual Document	Rev No.	Rev 3.0
<b>F</b>		Date:	Oct 17, 2022

# 7.3 Configurable Inputs Module

The User would be able to use the below functionalities of the keypad module via the DB variables and configuration file.

#### 7.3.1 Configurable Inputs module Enable/Disable

The SDK provides the user the ability to enable/disable the configurable functionality by modifying the default configuration file. Please see section 7.3.6 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_C FG_INPUT	PS_ENABLE/ PS_DISABLE	PS_ENABLE	PS_ENABLE: -Enables the Configurable inputmodule in the SDKPS_DISABLE: -Disables the configurable inputmodule in the SDK

#### 7.3.2 Configurable Inputs task Priority

The AI280 SDK supports the below task priorities and the user can modify the task priority for the configurable inputs module in the configuration file. Please see section 7.3.6 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_CFG_INPUT_T ASK_PRIORITY	osPriorityNone, osPriorityIdle, osPriorityLow, osPriorityLow1, osPriorityISR, osPriorityError, osPriorityReserved	osPriorityIdle (Recommended Value)	User can select any one of the priorities based on the application requirement

# 7.3.3 Configurable Inputs Task Time Out Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware for the configured inputs and update it in the DB so that when the user reads the DB, he will receive the latest updated data. This default configuration can be done in the AI280\_config.h. Please see section 7.3.6 for sample configuration.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Depti Soltmare		Date:	Oct 17, 2022

Sr. No	Variables	Options	Default State	Description
1	PS_CFG_INPUT_ TASK_TIMEOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommended Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the configured inputs in the Database.

#### 7.3.4 Configurable Inputs – Configure the number of samples

The AI280 SDK user can configure the number of samples to be considered for the average calculation of the readings from the hardware before it is updated to the database. This would improve the accuracy of the data updated in the DB. This default configuration can be done in the AI280\_config.h. Please see section 7.3.6 for sample configuration.

Sr.N 0	Variables	Options	Default Value	Description
1	CONF_INPUT_01 _NUMB_SAMPL _ES	MIN VALUE: 1 MAX VALUE: 255	1	User can set the number of sample values to be considered for the average calculation for configurable input 1
2	CONF_INPUT_02 _NUMB_SAMPL ES	MIN VALUE: 1 MAX VALUE: 255	1	User can set the number of sample values to be considered for the average calculation for configurable input 2
3	CONF_INPUT_03 _NUMB_SAMPL ES	MIN VALUE: 1 MAX VALUE: 255	1	User can set the number of sample values to be considered for the average calculation for configurable input 3
4	CONF_INPUT_04 _NUMB_SAMPL ES	MIN VALUE: 1 MAX VALUE: 255	1	User can set the number of sample values to be considered for the average calculation for configurable input 4
5	CONF_INPUT_05 _NUMB_SAMPL ES	MIN VALUE: 1 MAX VALUE: 255	1	User can set the number of sample values to be considered for the average calculation for configurable input 5

The user can dynamically set/get the number of samples to be considered for the average calculation for configurable input during the run time. To do so, the user can use the below DB variables.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Software		Date:	Oct 17, 2022

Field ID	Data Type	Permission	Size	Options	Description
CFG_INPUT_01_NUM B_SAMPLES	DBu8	READ/WRI TE	1	1 -255 range	This field is used to set number of samples to get an average value. The field is also used to read the number of samples set.
CFG_INPUT_02_NUM B_SAMPLES	DBu8	READ/WRI TE	1	1 -255 range	This field is used to set number of samples to get an average value. The field is also used to read the number of samples set.
CFG_INPUT_03_NUM B_SAMPLES	DBu8	READ/WRI TE	1	1 -255 range	This field is use to set number of samples to get an average value. The field is also used to read the number of samples set.
CFG_INPUT_04_NUM B_SAMPLES	DBu8	READ/WRI TE	1	1 -255 range	This field is use to set number of samples to get an average value. The field is also used to read the number of samples set.
CFG_INPUT_05_NUM B_SAMPLES	DBu8	READ/WRI TE	1	1 -255 range	This field is use to set number of samples to get an average value. The field is also used to read the number of samples set.

The below code snippet shows how the sample configuration can be altered from the application code,

if(update\_sample)

{

int num sample = 10;

/\* Set the sample count to 10 for configurable input 1 \*/

res = DL\_Set(CFG\_INPUT\_01\_NUMB\_SAMPLES,&num\_sample);

}

# 7.3.5 Configurable Inputs configuration

The AI280 SDK user can configure the 5 available configurable inputs as per his desired requirement as supported by the platform. To do so, he can configure the below parameters in the configuration file. This default configuration can be done in the AI280\_config.h. Please see section 7.3.6 for sample configuration.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Sr.No	Variables	Default Values	Default State	Description
1	CONF_INPUT_T YPE_01	CI_INPUT_FRE QUENCY, CI_INPUT_RESI STANCE, CI_DIGITAL_ST G, CI_DIGITAL_ST B, CI_INPUT_VOL TAGE_HIGH, CI_INPUT_VOL TAGE_LOW_6V,	CI_DIGITAL_S TB	User can configure the configurable input type 1 for input voltage, input frequency, input resistance, Digital STB and digital STG
2	CONF_INPUT_T YPE_02	CI_INPUT_FRE QUENCY, CI_INPUT_RESI STANCE, CI_DIGITAL_ST G, CI_DIGITAL_ST B, CI_INPUT_VOL TAGE_HIGH, CI_INPUT_VOL TAGE_LOW_6V,	CI_DIGITAL_S TG	User can configure the configurable input type 2 for input voltage, input frequency, input resistance, Digital STB and digital STG
3	CONF_INPUT_T YPE_03	CI_INPUT_FRE QUENCY, CI_INPUT_RESI STANCE, CI_DIGITAL_ST G, CI_DIGITAL_ST B, CI_INPUT_VOL TAGE_HIGH, CI_INPUT_VOL TAGE_LOW_6V,	CI_INPUT_RES ISTANCE	User can configure the configurable input type 3 for input voltage, input frequency, input resistance, Digital STB and digital STG
4	CONF_INPUT_T YPE_04	CI_INPUT_FRE QUENCY, CI_INPUT_RESI STANCE,	CI_INPUT_RES ISTANCE	User can configure the configurable input type 4 for input voltage, input frequency, input resistance, Digital STB and digital STG

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Soltware		Date:	Oct 17, 2022

		CI_DIGITAL_ST		
		G,		
		CI_DIGITAL_ST		
		В,		
		CI_INPUT_VOL		
		TAGE_HIGH,		
		CI_INPUT_VOL		
		TAGE_LOW_6V,		
		CI_INPUT_FRE		
		QUENCY,		
		CI_INPUT_RESI		
		STANCE,		
		CI_DIGITAL_ST		User can configure the configurable
5	CONF_INPUT_T	G,	CI_INPUT_RES	input type 5 for input voltage, input
5	YPE_05	CI_DIGITAL_ST	ISTANCE	frequency, input resistance, Digital STB
		В,		and digital STG
		CI_INPUT_VOL		
		TAGE_HIGH,		
		CI_INPUT_VOL		
		TAGE_LOW_6V,		

The user also has the ability to run time configure the 5 available configurable inputs as per his desired requirement as supported by the platform. To do so the user can use the below DB variables.

Field ID	Data Type	Permission	Size	Description	Comments
CFG_INPUT_01_TYP E	DBu8	READ/WRI TE	1	CI_INPUT_FREQUENCY, CI_INPUT_RESISTANCE, CI_DIGITAL_STG, CI_DIGITAL_STB, CI_INPUT_VOLTAGE_HIG H, CI_INPUT_VOLTAGE_LO W_6V	This field is used to Configure the input to any one of the types suggested. The field is also used to turn off the input. The status of the CFG_Input#01 type can be read using this field.
CFG_INPUT_02_TYP E	DBu8	READ/WRI TE	1	CI_INPUT_FREQUENCY, CI_INPUT_RESISTANCE, CI_DIGITAL_STG,	This field is used to Configure the input to any one of the types

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

				CI_DIGITAL_STB, CI_INPUT_VOLTAGE_HIG H, CI_INPUT_VOLTAGE_LO W_6V	suggested. The field is also used to turn off the input. The status of the CFG_Input#02 type can be read using this field.
CFG_INPUT_03_TYP E	DBu8	READ/WRI TE	1	CI_INPUT_FREQUENCY, CI_INPUT_RESISTANCE, CI_DIGITAL_STG, CI_DIGITAL_STB, CI_INPUT_VOLTAGE_HIG H, CI_INPUT_VOLTAGE_LO W_6V, CI_INPUT_OFF,	This field is used to Configure the input to any one of the types suggested. The field is also used to turn off the input. The status of the CFG_Input#03 type can be read using this field.
CFG_INPUT_04_TYP E	DBu8	READ/WRI TE	1	CI_INPUT_FREQUENCY, CI_INPUT_RESISTANCE, CI_DIGITAL_STG, CI_DIGITAL_STB, CI_INPUT_VOLTAGE_HIG H, CI_INPUT_VOLTAGE_LO W_6V	This field is used to Configure the input to any one of the types suggested. The field is also used to turn off the input. The status of the CFG_Input#04 type can be read using this field.
CFG_INPUT_05_TYP E	DBu8	READ/WRI TE	1	CI_INPUT_FREQUENCY, CI_INPUT_RESISTANCE, CI_DIGITAL_STG, CI_DIGITAL_STB, CI_INPUT_VOLTAGE_HIG H, CI_INPUT_VOLTAGE_LO W_6V	This field is used to Configure the input to any one of the types suggested. The field is also used to turn off the input. The status of the CFG_Input#05 type can be read using this field.

The below code snippet shows how the Configurable inputs type can be configured during the run time.

uint8\_t input1\_type;

/\* Read the current configuration for Configurable input 1 \*/

```
Get_DL(CFG_INPUT_01_TYPE , &input1_type);
/* If it is currently configured as frequency, change it to resistance */
if(CI_INPUT_FREQUENCY == input1_type)
{
     input1_type = CI_INPUT_RESISTANCE;
     Set_DL(CFG_INPUT_01_TYPE , &input1_type);
}
```

}

Once the user configures the various configurable inputs the platform service will read the data from the hardware every time the task time out occurs and update the below DB variables. The user can then access the same by using the DL\_get/DL\_set API's.

For example, if he has configured the Configurable input 1 as CI\_INPUT\_VOLTAGE\_HIGH then the user will have to read the CFG\_INPUT\_01\_VOLTAGE\_32V DB entry to read the voltage value in milli volts.

Field ID	Data Type	Permission	Size	Options	Description
CFG_INPUT_01_F REQUENCY	DBu32	READ	4	10Hz-20000Hz range	This field is used to read the frequency of CFG_Input#01. The frequency is read in hertz.
CFG_INPUT_01_V OLTAGE_32V	DBu16	READ	2	0-32000 range	This field is used to read the high voltage of CFG_Input#01. The voltage is read in milli-volts
CFG_INPUT_01_V OLTAGE_LOW_6 V	DBu16	READ	2	0-6V	This field is used to read the low voltage of CFG_Input#01. The voltage is read in volts
CFG_INPUT_01_R ESISTANCE	DBu16	READ	2	10hm – 5000hm range	This field is used to read the

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

					Resistance of CFG_Input#01. Resistance is read in ohms
CFG_INPUT_01_D IGITAL_STG	DBu8	READ	1	TRUE / FALSE	This field is used to read the Digital Input level of CFG_Input#01. If TRUE = digital active and FALSE = digital inactive
CFG_INPUT_01_D IGITAL_STB	DBu8	READ	1	TRUE / FALSE	This field is used to read the Digital Input level of CFG_Input#01. If TRUE = digital active and FALSE = digital inactive
CFG_INPUT_02_F REQUENCY	DBu32	READ	4	10Hz-20000Hz range	This field is used to read the frequency of CFG_Input#02. The frequency is read in milli- hertz.
CFG_INPUT_02_V OLTAGE_32V	DBu16	READ	2	0-32000 range	This field is used to read the voltage of CFG_Input#02. The voltage is read in milli-volts
CFG_INPUT_02_V OLTAGE_LOW_6 V	DBu16	READ	2	0-6V	This field is used to read the low voltage of CFG_Input#02. The voltage is read in volts

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Soltware		Date:	Oct 17, 2022

CFG_INPUT_02_R ESISTANCE	DBu16	READ	2	10hm – 5000hm range	This field is used to read the Resistance of CFG_Input#02. Resistance is read in ohms
CFG_INPUT_02_D IGITAL_STG	DBu8	READ	1	TRUE / FALSE	This field is used to read the Digital Input level of CFG_Input#02. If TRUE = digital active and FALSE = digital inactive
CFG_INPUT_02_D IGITAL_STB	DBu8	READ	1	TRUE / FALSE	This field is used to read the Digital Input level of CFG_Input#02. If TRUE = digital active and FALSE = digital inactive
CFG_INPUT_03_F REQUENCY	DBu32	READ	4	0-20000 range	This field is used to read the frequency of CFG_Input#03. The frequency is read in milli- hertz.
CFG_INPUT_03_V OLTAGE_32V	DBu16	READ	2	0-32000 range	This field is used to read the voltage of CFG_Input#03. The voltage is read in milli-volts
CFG_INPUT_03_V OLTAGE_LOW_6 V	DBu16	READ	2	0-6V	This field is used to read the low voltage of CFG_Input#03

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Soltware		Date:	Oct 17, 2022

					The voltage is read in volts
CFG_INPUT_03_R ESISTANCE	DBu16	READ	2	10hm – 5000hm range	This field is used to read the Resistance of CFG_Input#03. Resistance is read in ohms
CFG_INPUT_03_D IGITAL_STG	DBu8	READ	1	TRUE / FALSE	This field is used to read the Digital Input level of CFG_Input#03. If TRUE = digital active and FALSE = digital inactive
CFG_INPUT_03_D IGITAL_STB	DBu8	READ	1	TRUE / FALSE	This field is used to read the Digital Input level of CFG_Input#03. If TRUE = digital active and FALSE = digital inactive
CFG_INPUT_04_F REQUENCY	DBu32	READ	4	0-20000 range	This filed is used to read the frequency of CFG_Input#04. The frequency is read in milli- hertz.
CFG_INPUT_04_V OLTAGE_32V	DBu16	READ	2	0-32000 range	This field is used to read the voltage of CFG_Input#04. The voltage is read in milli-volts

Dept: Software	are User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

CFG_INPUT_04_V OLTAGE_LOW_6 V	DBu16	READ	2	0-6V	This field is used to read the low voltage of CFG_Input#04. The voltage is read in volts
CFG_INPUT_04_R ESISTANCE	DBu16	READ	2	10hm – 5000hm range	This filed is used to read the Resistance of CFG_Input#04. Resistance is read in ohms
CFG_INPUT_04_D IGITAL_STG	DBu8	READ	1	TRUE / FALSE	This field is used to read the Digital Input level of CFG_Input#04. If TRUE = digital active and FALSE = digital inactive
CFG_INPUT_04_D IGITAL_STB	DBu8	READ	1	TRUE / FALSE	This field is used to read the Digital Input level of CFG_Input#04. If TRUE = digital active and FALSE = digital inactive
CFG_INPUT_05_F REQUENCY	DBu32	READ	4	0-20000 range	This filed is used to read the frequency of CFG_Input#05. The frequency is read in milli- hertz.
CFG_INPUT_05_V OLTAGE_32V	DBu16	READ	2	0-32000 range	This field is used to read the voltage of CFG_Input#05.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

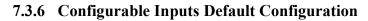
					The voltage is read in milli-volts
CFG_INPUT_05_V OLTAGE_LOW_6 V	DBu16	READ	2	0-6V	This field is used to read the low voltage of CFG_Input#05. The voltage is read in volts
CFG_INPUT_05_R ESISTANCE	DBu16	READ	2	10hm – 5000hm range	This filed is used to read the Resistance of CFG_Input#05. Resistance is read in ohms
CFG_INPUT_05_D IGITAL_STG	DBu8	READ	1	TRUE / FALSE	This field is used to read the Digital Input level of CFG_Input#05. If TRUE = digital active and FALSE = digital inactive
CFG_INPUT_05_D IGITAL_STB	DBu8	READ	1	TRUE / FALSE	This field is used to read the Digital Input level of CFG_Input#05. If TRUE = digital active and FALSE = digital inactive

The below code sample shows the configuration values read from CI1 during runtime,

```
uint8_t input1_type;
uint8_t val;
if(CI_INPUT_FREQUENCY == input1_type)
{
     val = 0;
     Get_DL(CFG_INPUT_01_FREQUENCY , (uint8_t *)&val);
}
else
if(CI_INPUT_VOLTAGE_HIGH == input1_type)
```

Dept: Software	: Software User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

```
{
      val = 0;
      Get_DL(CFG_INPUT_01_VOLTAGE_32V , (uint8_t *)&val);
}
else
if(CI_INPUT_VOLTAGE_LOW_6V == input1_type)
{
      val = 0;
      Get_DL(CFG_INPUT_01_VOLTAGE_LOW_6V , (uint8_t *)&val);
}
else
if(CI_INPUT_RESISTANCE == input1_type)
{
     val = 0;
      Get_DL(CFG_INPUT_01_RESISTANCE , (uint8_t *)&val);
}
else
if(CI_DIGITAL_STG == input1_type)
{
     val = 0;
      Get_DL(CFG_INPUT_01_DIGITAL_STG , (uint8_t *)&val);
}
else
if(CI_DIGITAL_STB == input1_type)
{
      val = 0;
      Get_DL(CFG_INPUT_01_DIGITAL_STB , (uint8_t *)&val);
}
```



```
/*!
 * Config input Platform service Enable (PS ENABLE) / Disable (PS DISABLE)
Macros
 */
#define SDK SERVICE CFG INPUT
                                                           PS ENABLE
#if (SDK_SERVICE_CFG_INPUT == PS_ENABLE)
/*!
 * Config_input Task <a href="Periodicity">Periodicity</a>
                                             100ms
 */
#define PS_CFG_INPUT_TASK_TIMEOUT
                                                     100
/*!
* CFG Input Task Priority
                             = 0,
* osPriorityNone
 * osPriorityIdle
                             = 1,
```

Dept: Software User Manual	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

<pre>* osPriorityLow * osPriorityLow1 * ,, * ,, * osPriorityISR * osPriorityError * osPriorityReserved */</pre>		و و و و	
<pre>#define PS_CFG_INPUT_TASK #define CONF_INPUT_TYPE_G #define CONF_INPUT_TYPE_G #define CONF_INPUT_TYPE_G #define CONF_INPUT_TYPE_G #define CONF_INPUT_01_NUK #define CONF_INPUT_02_NUK #define CONF_INPUT_03_NUK #define CONF_INPUT_04_NUK #define CONF_INPUT_05_NUK #define CONF_</pre>	01 02 03 04 05 08_SAMPLES 08_SAMPLES 08_SAMPLES 08_SAMPLES 5_CFG_INPUT		osPriorityIdle CI_DIGITAL_STB CI_DIGITAL_STG CI_INPUT_RESISTANCE CI_INPUT_RESISTANCE CI_INPUT_RESISTANCE 1 1 1 1

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 7.4 LED Module

The User would be able to use the below functionalities of the LED module via the DB variables and configuration file.

#### 7.4.1 LED module Enable/Disable

The SDK provides the ability to the user to enable/disable the LED functionality by modifying the default configuration file. Please see section 7.4.11 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_L ED	PS_ENABLE PS_DISABLE	PS_ENABLE	<b>PS_ENABLE: -</b> Enables the LED module in the SDK <b>PS_DISABLE: -</b> Disables the LED module in the SDK

# 7.4.2 LED Time Out Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware and update it in the DB so that when the user reads the DB, he will receive the latest updated data if any or perform any other routine tasks. This default configuration can be done in the AI280\_config.h. Please see section 7.4.11 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_LED_TASK_T IMEOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommended Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the configured inputs in the Database.

# 7.4.3 LED task Priority

The AI280 SDK supports the below task priorities and the user can modify the task priority for the LED module in the configuration file. Please see section 6.4.11 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_LED_TASK_P RIORITY	osPriorityNone, osPriorityIdle, osPriorityLow, osPriorityLow1, osPriorityISR,	osPriorityIdle (Recommended Value)	User can select any one of the priorities based on the application requirement

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Depti Soltmare		Date:	Oct 17, 2022

osPriorityError, osPriorityReserve	
d	

## 7.4.4 Maximum LED'S Configuration

The AI280 SDK supports a maximum of 2 LED's and the user has the ability to configure the MAX LEDs supported by the device in the configuration file. Please see section 7.4.11 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	MAX_LED_NUM	1 or 2	2	User can operate maximum 2 LED

## 7.4.5 Configuring RED LED Enable/Disable

The SDK provides the ability to the user to enable/disable the RED LED functionality by modifying the default configuration file. Please see section 7.4.11 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	CONF_RED_LED _PS_STATE	PS_ENABLE/ PS_DISABLE	PS_ENABLE	User can enable/Disable the RED LED

### 7.4.6 Configuring RED LED State

The AI280 SDK supports the user to configure the default state of the RED LED and this can be done by modifying the below parameter in the configuration file. Please see section 7.4.11 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	CONF_RED_LED _STATE	LED_CONF_ON/ LED_CONF_OF F	LED_CONF_ON	User can turn ON/OFF the RED LED

During runtime, the user can read and modify the RED LED state by reading and writing to the below DB variables.

Field ID	Data Type	Permission	Size	Description	Comments
LED_RED_STATE	DBu8	READ/WR ITE	1	ON / OFF	This field is used to enable/disable the LED_RED.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
	Ultware User Manual Document	Date:	Oct 17, 2022

		ON = Light up the LED, OFF =
		Turn off the LED.

Below code snippet shows how the RED LED can be read and written into the DB.

```
uint8_t state;
#if (SDK SERVICE LED == PS ENABLE)
#if(CONF RED LED PS STATE == PS ENABLE)
      /* Get the RED LED Status from the DB */
     Get_DL(LED_RED_STATE, &state);
      if (LED ON == state)
     {
            /* LED is on */
            state = LED_OFF;
            /* Set the RED LED Status from the DB */
            Set_DL(LED_RED_STATE, &state);
      }
     else
      {
            /* LED is OFF; */
      }
#endif
       #endif
```

### 7.4.7 Configuring RED LED blinking

The AI280 SDK supports the user to configure the RED LED blinking time period in milli second and this can be done by modifying the below parameter in the configuration file. Please see section 7.4.11 for sample configuration.

The value configured here is multiplied by 250ms to get the blinking period. So, if we have set a value of 2 here, between every blink there will be a (2\*250ms = 500ms) time lag.

Sr. No	Variables	Options	Default State	Description
1	CONF_RED_LED _BLINKING_MS	0-1000	l (Recommended Value)	User can change the Blink time period for the RED LED

If this value is set as 0 then the blinking is disabled.

During runtime, the user can read and modify the RED LED state and RED LED blinking time period by reading and writing to the below DB variables.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
	Sept. Software User Manual Document	Date:	Oct 17, 2022

Field ID	Data Type	Permission	Size	Description	Comments
LED_RED_BLINKIN G	DBu16	READ/WR ITE	2	0-65535 range	This field is used to set and read back the Blinking period in milli seconds.

Below code snippet shows how the RED LED can be read from the DB.

```
uint8_t state;
uint16_t blink;
#if (SDK_SERVICE_LED == PS_ENABLE)
#if(CONF_RED_LED_PS_STATE == PS_ENABLE)
blink = 0;
/* Get the RED LED blink period from the DB */
Get_DL(LED_RED_BLINKING, (uint8_t *)&blink);
state = LED_ON;
/* Set the RED LED Status to the DB */
Set_DL(LED_RED_STATE, &state);
#endif
```

#endif

### 7.4.8 Configuring AMB LED Enable/Disable

The SDK provides the user the ability to enable/disable the AMBER LED functionality by modifying the default configuration file. Please see section 7.4.11 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	CONF_AMB_LED _PS_STATE	PS_ENABLE/ PS_DISABLE	PS_ENABLE	User can enable/Disable the AMB LED

### 7.4.9 Configuring AMB LED State

The AI280 SDK supports the user to configure the default state of the AMBER LED and this can be done by modifying the below parameter in the configuration file. Please see section 7.4.11 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	CONF_AMB_LED _STATE	LED_CONF_ON/ LED_CONF_OF F	LED_CONF_ON	User can turn ON/OFF the AMB LED

During runtime, the user can read and modify the AMB LED state by reading and writing to the below DB variables.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Depti Soltmare		Date:	Oct 17, 2022

Field ID	Data Type	Permission	Size	Description	Comments
LED_AMB_STATE	DBu8	READ/WR ITE	1	ON / OFF	This field is used to enable/disable the LED_AMB. ON = Light up the LED, OFF = Turn off the LED.

Below code snippet shows how the AMB LED can be read and written into the DB.

```
uint8_t state;
#if (SDK_SERVICE_LED == PS_ENABLE)
#if(CONF_AMB_LED_PS_STATE == PS_ENABLE)
      /* Get the RED LED Status from the DB */
     Get_DL(LED_AMB_STATE, &state);
     if (LED_ON == state)
      {
            /* LED is on */
            state = LED_OFF;
            /* Get the RED LED Status from the DB */
            Set_DL(LED_AMB_STATE, &state);
      }
     else
      {
            /* LED is OFF; */
      }
#endif
       #endif
```

### 7.4.10 Configuring AMB LED blinking

The AI280 SDK supports the user to configure the AMBER LED blinking time period in milli second and this can be done by modifying the below parameter in the configuration file. Please see section 7.4.11 for sample configuration.

The value configured here is multiplied by 250ms to get the blinking period. So, if we have set a value of 2 here, between every blink there will be a (2\*250ms = 500ms) time lag. If this value is set as 0 then the blinking is disabled.

Sr. No	Variables	Options	Default State	Description
1	CONF_AMB_LED_ BLINKING_MS	0-1000	l (Recommended Value)	User can change the Blink time period for the AMB LED

During runtime, the user can read and modify the AMBER LED state and AMBER LED blinking by reading and writing to the below DB variables.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Software		Date:	Oct 17, 2022

Field ID	Data Type	Permission	Size	Description	Comments
LED_AMB_BLINKI NG	DBu16	READ/WR ITE	2	0-65535 range	This field is used to set and read back the Blinking period in milli seconds.

Below code snippet shows how the AMB LED can be read and written into the DB.

```
uint8_t state;
uint16_t blink;
#if (SDK_SERVICE_LED == PS_ENABLE)
#if(CONF_AMB_LED_PS_STATE == PS_ENABLE)
blink = 0;
/* Get the AMB LED blink period from the DB */
Get_DL(LED_AMB_BLINKING, (uint8_t *)&blink);
blink = 2;
/* Set the AMB LED blink period from the DB */
Set_DL(LED_AMB_BLINKING, (uint8_t *)&blink);
state = LED_ON;
/* Set the RED LED Status to the DB */
Set_DL(LED_AMB_STATE, &state);
```

#### #endif

#endif

#### 7.4.11 LED Sample Configuration

```
/*!
* LED Platform service Enable (PS_ENABLE) / Disable (PS_DISABLE) Macros
*/
#define SDK SERVICE LED
                                                       PS ENABLE
#if (SDK_SERVICE_LED == PS_ENABLE)
/*!
 * LED Task Periodicity
                              100ms
*/
#define PS_LED_TASK_TIMEOUT
                                                       100
/*!
* LED Task Priority
* osPriorityNone
                           = 0,
 * osPriorityIdle
                         = 1,
                          = 8,
 * osPriorityLow
 * osPriorityLow1
                           = 8+1,
            رر
                                            ر ر
           رر
                                            ر ر
 * osPriorityISR
                           = 56,
 * osPriorityError
                           = -1,
```

```
* osPriorityReserved = 0x7FFFFFF
*/
#define PS LED TASK PRIORITY
                                                             osPriorityIdle
/*!
* Maximum Number of LED required for this application
 *
 * MACOR Supported
 * MAX_LED_NUM : This hardware support maximum of 2 LED's
 * CONF_xx_LED_PS_STATE
                                                PS ENABLE
 *
                                                PS_DISABLE
 *
                                                LED CONF OFF
 * CONF_xx_LED_STATE
 *
                                                LED_CONF_ON
 * CONF_xx_LED_BLINKING_MS
                                                <0-65535>
 *
*/
                                                      2
#define MAX_LED_NUM
#define CONF_RED_LED_PS_STATE
                                                 PS ENABLE
#define CONF_RED_LED_STATE
                                                LED_CONF_ON
#define CONF RED LED BLINKING MS
                                                      1
#define CONF_AMB_LED_PS_STATE
                                                 PS ENABLE
#define CONF AMB LED STATE
                                                LED_CONF_ON
#define CONF_AMB_LED_BLINKING_MS
                                                      1
#endif
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 7.5 Power Monitor Module

The User would be able to use the below functionalities of the power monitor module via the DB variables and configuration file.

### 7.5.1 Power Monitor module Enable/Disable

The SDK provides the user the ability to enable/disable the power monitor functionality by modifying the default configuration file. Please see section 7.5.5 for sample configuration.

If the Configurable inputs is disabled then the power monitor module will also be disabled in the configuration file.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_P OWER_MONITO R	PS_ENABLE PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE: -</li> <li>Enables the power monitor module in the SDK</li> <li>PS_DISABLE: -</li> <li>Disables the power monitor module in the SDK</li> </ul>

## 7.5.2 Power Monitor Time Out Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware and update it in the DB so that when the user reads the DB, he will receive the latest updated data if any or perform any other routine tasks. This default configuration can be done in the AI280\_config.h. Please see section 7.5.5 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_POWER_MON ITOR_TASK_TIM EOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommended Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the configured inputs in the Database.

# 7.5.3 Power Monitor task Priority

The AI280 SDK supports the below task priorities and the user can modify the task priority for the Power Monitor module in the configuration file. Please see section 7.5.5 for sample configuration.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Depti Soltmare		Date:	Oct 17, 2022

Sr. No	Variables	Options	Default State	Description
1	PS_POWER_MON ITOR_TASK_PRI ORITY	osPriorityNone, osPriorityIdle, osPriorityLow, osPriorityLow1, osPriorityISR, osPriorityError, osPriorityReserved	osPriorityIdle (Recommended Value)	User can select any one of the priorities based on the application requirement

### 7.5.4 Power Monitor Functionality Support

The AI280 SDK supports the below values which are monitored by the power monitor module. They are,

- 1) Battery LEVEL
- 2) Ignition Status
- 3) THERMOSTAT

During runtime, the user can read the Battery level, Ignition status and Thermostat level by reading the below DB variables.

Field ID	Data Type	Permission	Size	Description	Comments
BATTERY_LEVEL	DBu16	READ	2	Voltage in milli volts	This field is used to read the BATTERY_LEVEL in milli-volts.
IGNITION_STATUS	DBu8	READ	1	ON/OFF	This field is used to read the status of IGNITION_STATUS.
THERMOSTAT_LEV EL	float	READ	4	temperature in Celsius	This field is used to read theTHERMOSTAT_LEVELinCelsius.

The below code snippet shows how to read the Battery Level, Ignition Status and the Thermostat level.

```
uint8_t val;
void PWRMNTRView::trigger()
{
#if (SDK_SERVICE_POWER_MONITOR == PS_ENABLE)
        uint16_t val = 0;
        float val_thermostat = 0;
        switch(key_position)
```

}

```
{
            case 1:
                  /* Get ignition Status */
                  Get_DL(IGNITION_STATUS, (uint8_t*)&val);
                  break;
            case 2:
                  /* Get Temperature level */
                  Get_DL(THERMOSTAT_LEVEL, (uint8_t*)&val_thermostat);
                  break;
            case 3:
                  /* Get Battery level */
                  Get_DL(BATTERY_LEVEL, (uint8_t*)&val);
                  break;
      }
#endif
```

## 7.5.5 Power Monitor sample configuration

```
/* Power Monitor Platform Service Enable (PS ENABLE) / Disable (PS DISABLE)
Macros */
#define SDK SERVICE POWER MONITOR
                                                      PS ENABLE
#if (SDK_SERVICE_POWER_MONITOR == PS_ENABLE)
#if ((SDK_SERVICE_POWER_MONITOR == PS_ENABLE) && (SDK_SERVICE_CFG_INPUT ==
PS DISABLE))
#undef SDK_SERVICE_POWER_MONITOR
                                                     PS DISABLE
#define SDK SERVICE POWER MONITOR
#endif
/*!
* Power Monitor Task Priority
* osPriorityNone = 0,
                         = 1,
 * osPriorityIdle
 * osPriorityLow
                         = 8,
                          = 8+1,
 * osPriorityLow1
 *
           ر ر
                                            ر ر
 *
           ر ر
                                           ر ر
 * osPriorityISR
                          = 56,
 * osPriorityReserved = 0x70
*/
                          = 0x7FFFFFFF
 */
#define PS POWER MONITOR TASK PRIORITY
                                                osPriorityIdle
/*!
 * Power Monitor Task <u>Periodicity</u> 100ms
*/
#define PS_POWER_MONITOR_TASK_TIMEOUT
                                                 100
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

### **#endif** //SDK\_SERVICE\_POWER\_MONITOR

# 7.6 Bluetooth Low Energy (BLE) Module

The AI280 SDK User would be able to use the below functionalities of the BLE module via the DB variables and configuration file.

## 7.6.1 BLE module Enable/Disable

The SDK provides the user the ability to enable/disable the BLE functionality by modifying the default configuration file. Please see section 7.6.6 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_B LE	PS_ENABLE/ PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE: -</li> <li>Enables the BLE module in the SDK</li> <li>PS_DISABLE: -</li> <li>Disables the BLE module in the SDK</li> </ul>

# 7.6.2 BLE Time Out Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware and update it in the DB so that when the user reads the DB, he will receive the latest updated data if any or perform any other routine tasks. This default configuration can be done in the AI280\_config.h. Please see section 7.6.6 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_BLE_TASK_T IMEOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommended Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the configured inputs in the Database.

# 7.6.3 BLE Monitor task Priority

The AI280 SDK supports the below task priorities, and the user can modify the task priority for the BLE module in the configuration file. Please see section 7.6.6 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_BLE_TASK_P RIORITY	osPriorityNone, osPriorityIdle, osPriorityBelowN	ormal7	User can select any one of the priorities based on the application requirement

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

ormal7,	Value)	
osPriorityLow,		
osPriorityLow1,		
osPriorityISR,		
osPriorityError,		
osPriorityReserve		
d		

### 7.6.4 BLE module device Name configuration

The AI280 SDK user can configure the device name of BLE. This default configuration can be done in the AI280\_config.h

Sr. No	Variables	Options	Default State	Description
1	BLE_DEVICE_N AME	Any name as per the user requirement	"Max12345678"	This field is used to set and read the BLE device name. The maximum length is 20 characters.

The user would be able to read the BLE module name during run time via the DB variables shown below.

Field ID	Data Type	Permission	Size	Description	Comments
BLE_DEVICE_NAM E	DBu8	READ/WRITE	20	"devicename"	This field is used to set and read the BLE device name. The maximum length is 20 characters.

The below code snippet shows how to read the BLE name,

```
#if (SDK_SERVICE_BLE == PS_ENABLE)
    uint8_t name;
    /* Get the BLE device name */
    Get_DL(BLE_DEVICE_NAME, &name);
```

# 7.6.5 BLE module RX/TX

The AI280 SDK allows the users to use the BLE channel to send or receive data. To do so please use the below variables.

To read incoming data over the BLE channel, the user will need to monitor the RX BUFFER STATUS variable and see if there is any pending data available and if yes read the data and then update the RX STATUS.

To send data over the BLE channel, the user will fill the TX buffer and then update the status as true. The platform will then send the data over BLE and then clear the status when the data is sent.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Field ID	Data Type	Permission	Size	Options	Description
BLE_TX_STATU S	DBu8	READ	1	TRUE/FALSE	This field is used read and write the BLE_TX_STATUS. One needs to write TRUE to send data. The same is cleared when data is sent
BLE_TX_BUFFE R_STATUS	DBu8	READ	1	FULL/NO_F ULL	This field is used to read the Status of the TX buffer.
BLE_TX_DATA	DBu8	WRITE	64	Data to be send to BLE	This field contains the BLE TX buffer data.
BLE_RX_STATU S	DBu8	READ/WRITE	1	TRUE/FALSE	This field is used read and write the BLE_RX_STATUS. One needs to read TRUE to receive data. The same is cleared when data is sent
BLE_RX_BUFF ER_STATUS	DBu8	READ	1	MSG_PENDI NG/EMPTY	This field is used to read the Status of the RX buffer.
BLE_RX_DATA	DBu8	READ	64	DATA received	This field contains the BLE RX buffer data.
BLE_RX_DATA _SIZE	DBu8	READ	1	(Only applicable for USER_DATA _MODE)	This field is used the read the size of BLE RX data.

The sample code below suggests the process to read the RX Data received.

```
Unicode::UnicodeChar buffer1[64];
uint8_t rxbuffer1[64];
Unicode::UnicodeChar trxbuffer3[64];
char str4[50];
void BLEView::trigger()
{
#if (SDK_SERVICE_BLE == PS_ENABLE)
uint8_t status ;
/* Get the RX status */
Get_DL(BLE_RX_BUFFER_STATUS, &status);
if(BLE_RX_MSG_PENDING == status)
{
/* Clear the memory */
```

De	nt:	Software
DU	<b>br</b> .	Solumate

```
memset(&rxbuffer1[0], 0x00, sizeof(rxbuffer1));
            /* Read the Rx data from the DB */
            Get DL(BLE RX DATA, (uint8 t *)&rxbuffer1[0]);
            memset(&trxbuffer3[0], 0x00, sizeof(trxbuffer3));
            Unicode::strncpy(&trxbuffer3[0], (const char*)&rxbuffer1[0],
            strlen((const char*)rxbuffer1));
            memset(RCVTEXTBBuffer, 0x00, sizeof(RCVTEXTBBuffer));
            Unicode::snprintf(RCVTEXTBBuffer, RCVTEXTB_SIZE, "%c %c %c %c
%c",
            trxbuffer3[0], trxbuffer3[1] ,trxbuffer3[2],trxbuffer3[3],
      trxbuffer3[4]);
            RCVTEXTB.invalidate();
            status = TRUE;
            /* Clear the RX buffer */
            Set_DL(BLE_RX_STATUS, &status);
      }
#endif
      this->getRootContainer().invalidate();
}
```

The sample code below suggests the process to send the Data over Bluetooth,

```
/*set the BLE tx data */
Set_DL(BLE_TX_DATA , (uint8_t *)&buffer1[0]);
status = TRUE;
/* Clear the RX buffer */
Set_DL(BLE_TX_STATUS, &status);
```

### 7.6.6 BLE sample configuration

```
/*!
* BLE Platform service Enable (PS ENABLE) / Disable (PS DISABLE) Macros
*/
#define SDK SERVICE BLE
                                                    PS ENABLE
#if (SDK SERVICE BLE == PS ENABLE)
/*!
* BLE Task Priority
* osPriorityNone
                        = 0,
                         = 1,
* osPriorityIdle
 * osPriorityLow
                         = 8,
 * osPriorityLow1
                          = 8+1,
           ر ر
                                          ر ر
           رر
                                          ر ر
 * osPriorityISR
                        = 56,
 * osPriorityError
                        = -1,
                      = 0x7FFFFFFF
 * osPriorityReserved
 */
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

#define PS_BLE_TASK_PRIORITY	
osPriorityBelowNormal7	
/*!	
* BLE Task <u>Periodicity</u> 100ms	
*/	
<pre>#define PS_BLE_TASK_TIMEOUT</pre>	100
/*!	
* BLE Device Name	
*/	
<pre>#define BLE_DEVICE_NAME</pre>	"Max12345678"
<pre>#endif //SDK_SERVICE_BLE</pre>	

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 7.7 Timer Module

The AI280 SDK User would be able to use the below functionalities of the Timer module via the DB variables and configuration file.

### 7.7.1 Timer Module Enable/Disable

The SDK provides the user the ability to enable/disable the Timer functionality by modifying the default configuration file. Please see section 7.7.7 for sample code snippet.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_S WTIMER	PS_ENABLE/PS _DISABLE	PS_ENABLE	<b>PS_ENABLE: -</b> Enables the timer module in the SDK <b>PS_DISABLE: -</b> Disables the timer module in the SDK

# 7.7.2 Timer Module Time Out Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware and update it in the DB so that when the user reads the DB, he will receive the latest updated data if any or perform any other routine tasks. This default configuration can be done in the AI280\_config.h. Please see section 7.7.7 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_SWT_TASK_T IMEOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommended Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the configured inputs in the Database.

# 7.7.3 Timer Module Task Priority

The AI280 SDK supports the below task priorities and the user can modify the task priority for the timer module in the configuration file. Please see section 7.7.7 for sample code snippet.

Sr. No	Variables	Options	Default State	Description
1	PS_SWT_TASK_P RIORITY	osPriorityNone, osPriorityIdle, osPriorityLow, osPriorityLow1, osPriorityISR, osPriorityError,	osPriorityIdle (Recommended Value)	User can select any one of the priorities based on the application requirement

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

osPriorityReserve	
d	

# 7.7.4 Timer Start or Stop

The AI280 SDK supports six software timers. The user can start or stop the timers during run time and also get the current status of the timer. To do so he can read or write the timer state using the below DB variables.

Field ID	Data Type	Permission	Size Bytes	Description	Comments
TIMER_STATUS_01	DBu8	READ/WRITE	1	START/ST OP	This field is used set and read the Timer state (START/STOP).
TIMER_STATUS_02	DBu8	READ/WRITE	1	START/ST OP	This field is used set and read the Timer state (START/STOP).
TIMER_STATUS_03	DBu8	READ/WRITE	1	START/ST OP	This field is used set and read the Timer state (START/STOP).
TIMER_STATUS_04	DBu8	READ/WRITE	1	START/ST OP	This field is used set and read the Timer state (START/STOP).
TIMER_STATUS_05	DBu8	READ/WRITE	1	START/ST OP	This field is used set and read the Timer state (START/STOP).
TIMER_STATUS_06	DBu8	READ/WRITE	1	START/ST OP	This field is used set and read the Timer state (START/STOP).

Below code snippet shows how the timer state can be set

```
uint8_t state;
Get_DLTIMER_STATUS_01, &state);
if(state == 1)
{
state = 2;
Set_DL(TIMER_STATUS_01, &state);
}
else if(state == 2)
{
```

state = 1;

set\_DL(TIMER\_STATUS\_01, &state);

}

Once the timer expires the SDK updates the timer callback parameter in the DB with the status as *CALLBACK\_OCCURED* and the user can monitor the same to know if the timer has expired. He can use the below DB variables to do the same.

Field ID	Data Type	Permission	Size Bytes	Description	Comments
TIMER_CALLBAC K_01	DBu8	READ/WRITE	1	CALLBAC K_CLEAR/ CALLBAC K_OCCUR ED	This field is used set and clear the Timer state.
TIMER_CALLBAC K_02	DBu8	READ/WRITE	1	CALLBAC K_CLEAR/ CALLBAC K_OCCUR ED	This field is used set and clear the Timer state.
TIMER_CALLBAC K_03	DBu8	READ/WRITE	1	CALLBAC K_CLEAR/ CALLBAC K_OCCUR ED	This field is used set and clear the Timer state.
TIMER_CALLBAC K_04	DBu8	READ/WRITE	1	CALLBAC K_CLEAR/ CALLBAC K_OCCUR ED	This field is used set and clear the Timer state.
TIMER_CALLBAC K_05	DBu8	READ/WRITE	1	CALLBAC K_CLEAR/ CALLBAC K_OCCUR ED	This field is used set and clear the Timer state.
TIMER_CALLBAC K_06	DBu8	READ/WRITE	1	CALLBAC K_CLEAR/ CALLBAC K_OCCUR ED	This field is used set and clear the Timer state.

The below code snippet shows you how you can read the S/W timer status,

#if (SDK\_SERVICE\_SWTIMER == PS\_ENABLE)

Rev No.	<b>Rev 3.0</b>
Date:	Oct 17, 2022

```
uint8_t timer1_state = 0;
uint8_t rtc1_val = 0;
uint8_t timeout_val = 0;
Get_DL(TIMER_CALLBACK_01, &timer1_state);
if(CALLBACK_OCCURED == timer1_state)
{
    /* Timer expired */
  }
#endif
```

## 7.7.5 Timer Mode Configuration

The S/W timers can be configured as single shot and periodic. During runtime, the user can read or write timer mode variable in the DB to update/get the configuration of the S/W timers.

Field ID	Data Type	Permission	Size Bytes	Description	Comments
TIMER_MODE_01	DBu8	READ/WRITE	1	ONESHOT/ PERIODIC	This field is used to set and read Timer Mode. (ONESHOT/PERIODIC)
TIMER_MODE_02	DBu8	READ/WRITE	1	ONESHOT/ PERIODIC	This field is used to set and read Timer Mode. (ONESHOT/PERIODIC)
TIMER_MODE_03	DBu8	READ/WRITE	1	ONESHOT/ PERIODIC	This field is used to set and read Timer Mode. (ONESHOT/PERIODIC)
TIMER_MODE_04	DBu8	READ/WRITE	1	ONESHOT/ PERIODIC	This field is used to set and read Timer Mode. (ONESHOT/PERIODIC)
TIMER_MODE_05	DBu8	READ/WRITE	1	ONESHOT/ PERIODIC	This field is used to set and read Timer Mode. (ONESHOT/PERIODIC)
TIMER_MODE_06	DBu8	READ/WRITE	1	ONESHOT/ PERIODIC	This field is used to set and read Timer Mode. (ONESHOT/PERIODIC)

Below code snippet shows how the timer state can be set and read,

```
uint8_t shot;
GET_DB(TIMER_MODE_01, (uint8_t *)&shot);
if(shot == 0)
{
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

```
shot = 1;
}
else if(shot == 1)
{
    shot = 0;
}
/* Set the Timer Mode_1 */
SET_DB(TIMER_MODE_01, (uint8_t *)&shot);
```

### 7.7.6 Timer Timeout Configuration

During runtime, the user can set or get the timeout period for the S/W timers using the below DB variables.

Please note that the timer timeout can be increased in steps of 50ms. And the max timeout value should be lesser than 65535.

Field ID	Data Type	Permission	Size Bytes	Description	Comments
TIMER_TIMEOUT _01	DBu16	READ/WRITE	2	time in milli seconds	This field is used to get/set the timeout in milli second
TIMER_TIMEOUT _02	DBu16	READ/WRITE	2	time in milli seconds	This field is used to get/set the timeout in milli second
TIMER_TIMEOUT _03	DBu16	READ/WRITE	2	time in milli seconds	This field is used to get/set the timeout in milli second
TIMER_TIMEOUT _04	DBu16	READ/WRITE	2	time in milli seconds	This field is used to get/set the timeout in milli second
TIMER_TIMEOUT _05	DBu16	READ/WRITE	2	time in milli seconds	This field is used to get/set the timeout in milli second
TIMER_TIMEOUT _06	DBu16	READ/WRITE	2	time in milli seconds	This field is used to get/set the timeout in millisecond

Below code snippet shows how the timer timeout can be set,

```
uint32_t sw_timeout;
uint32_t timeout;
if ((timeout > 0) && (timeout <= 1300))
{
timeout --;
sw_timeout = (timeout * 50);
}
else
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

```
{
}
/* Set the Timer Timeout_1 */
SET_DB(TIMER_TIMEOUT_01, (uint8_t *)&sw_timeout);
```

Please note that the timer timeout can be increased in steps of 50ms. And the max timeout value should be lesser than 65535 hence the max counter in the above loop is restricted to 1300.

```
7.7.7 Timer sample configuration
/*!
* SWTIMER Platform service Enable (PS_ENABLE) / Disable (PS_DISABLE) Macros
*/
#define SDK_SERVICE_SWTIMER
                                                    PS_ENABLE
#if (SDK_SERVICE_SWTIMER == PS_ENABLE)
/*!
* SWTIMER Task Priority
                     = 0,
 * osPriorityNone
* osPriorityIdle
                        = 1,
                        = 8,
 * osPriorityLow
 * osPriorityLow1
                        = 8+1,
 *
           رر
                                          رر
 *
           ر ر
                                          ر ر
                        = 56,
* osPriorityISR
 * osPriorityError
                        = -1,
 * osPriorityReserved = 0x7FFFFFF
 */
                                                          osPriorityIdle
#define PS_SWT_TASK_PRIORITY
/*!
* SWTimer Task Periodicity
                            100ms
*/
#define PS_SWT_TASK_TIMEOUT
                                                          100
#endif
       //SDK_SERVICE_SWTIMER
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
· · · · · · · · · · · · · · · · · · ·		Date:	Oct 17, 2022

# 7.8 RTC Module

The AI280 SDK User would be able to use the below functionalities of the RTC module via the DB variables and configuration file.

## 7.8.1 RTC Module Enable/Disable

The SDK provides the ability to the user to enable/disable the RTC functionality by modifying the default file. Please see section 7.8.8 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_R TC	PS_ENABLE/PS _DISABLE	PS_ENABLE	<b>PS_ENABLE: -</b> Enables the RTC module in the SDK <b>PS_DISABLE: -</b> Disables the RTC module in the SDK

# 7.8.2 RTC Timeout Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware and update it in the DB so that when the user reads the DB, he will receive the latest updated data if any or perform any other routine tasks. This default configuration can be done in the AI280\_config.h. Please see section 7.8.8 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_RTC_TASK_T IMEOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommen ded Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the configured inputs in the Database.

# 7.8.3 RTC Task Priority

The AI280 SDK supports the below task priority and the user can modify the task priority for the RTC module in the configuration file. Please see section 7.8.8 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_RTC_TASK_P RIORITY	osPriorityNone, osPriorityIdle, osPriorityLow, osPriorityLow1, osPriorityISR,	osPriorityIdle (Recommended Value)	User can select any one of the priorities based on the application requirement

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

osPriorityError,	
osPriorityReserve	
d	

The user would be able to use the below functionalities of the RTC module via the DB variables and configuration file.

## 7.8.4 RTC Date and Time Configuration

The user can get or set the real time clock using the following DB variables.

To set the RTC, the user has to set individually each of the RTC parameters and then call the SET\_RTC DB variable to set the RTC TIME.

Field ID	Data Type	Permission	Size	Descriptio n	Comments
RTC_SECOND	DBu8	READ/WRITE	1	0-59	Valid values to set the real time second are from 0 to 59
RTC_MINUTE	DBu8	READ/WRITE	1	0-59	Valid values to set the real time minute are from 0 to 59
RTC_HOUR	DBu8	READ/WRITE	1	0-24	Valid values to set the real time hour are from 0 to 24
RTC_DATE	DBu8	READ/WRITE	1	1-31	Valid values to set the real time day are from 1 to 31
RTC_WEEK_DA Y	DBu8	READ/WRITE	1	1-7	Valid values to set the real time week day are from 1 = Monday to 7= Sunday
RTC_MONTH	DBu8	READ/WRITE	1	1-12	Valid values to set the real time month are from 1 to 12
RTC_YEAR	DBu8	READ/WRITE	1	00-99	Valid values to set the real time year are from 0 to 99
SET_RTC	DBu8	READ/WRITE	1	SET_RTC	(User need to set the above RTC parameters and then enable the SET RTC to set the time)

The below snapshot is a sample for updating the RTC Time

uint8\_t hours; uint8\_t Minutes; uint8\_t Seconds; uint8\_t WeekDay;

Dept: Software	User Manual Document	Rev No.	Rev 3.0
- · <b>F</b> ·· · · · · · · · · · ·		Date:	Oct 17, 2022

```
uint8_t date;
        uint8_t Month;
        uint8_t Year;
        uint8_t res;
Set DL(GET RTC SECOND, &Seconds);
Set_DL(GET_RTC_MINUTE, &Minutes);
Set DL(GET RTC HOUR, &Hours);
Set_DL(GET_RTC_DATE, &Date);
Set_DL(GET_RTC_WEEK_DAY, &WeekDay);
Set_DL(GET_RTC_MONTH, &Month);
Set_DL(GET_RTC_YEAR, &Year);
res = 1;
      Set_DL(SET_RTC, &res);
The sample code below is an example of reading the RTC values.
      void RTCNXTView::trigger()
      {
      #if (SDK_SERVICE_RTC == PS_ENABLE)
            uint8_t Seconds;
            uint8_t Minutes;
            uint8 t Hours;
         tickCounter++;
         if( 10 <= tickCounter)</pre>
         {
               tickCounter = 0;
                  /* Get the RTC DB */
                  Get_DL(GET_RTC_SECOND, &Seconds);
                  Get_DL(GET_RTC_MINUTE, &Minutes);
                  Get_DL(GET_RTC_HOUR, &Hours);
                //screenViewBase::setupScreen();
                digitalHours = Hours;
                digitalMinutes = Minutes;
                digitalSeconds = Seconds;
              digitalClock1.setTime24Hour(digitalHours, digitalMinutes,
      digitalSeconds);
                digitalClock1.invalidate();
         }
      #endif
      }
```

#### 7.8.5 RTC Time Format

The SDK supports the 12- and 24-hour time format. The user can read/update the RTC Time format during run time using the below DB variables. Below are their definitions,

**#define** FORMAT\_12\_HOUR

Dept: Software	User Manual Document	Rev No.	Rev 3.0
T. T. T. T. T. T. T. T. T. T. T. T. T. T		Date:	Oct 17, 2022

#### #define FORMAT\_24\_HOUR

0

Field ID	Data Type	Permission	Size	Descriptio n	Comments	
RTC_TIME_FO RMAT	DBu8	READ/WRITE	1	AM/PM	RTC Time Format (AM/PM)	
uint8_t form	uint8_t format;					
/* Get the F	/* Get the RTC time format */					
Get_DL( <i>RT</i>	C_TIME_	FORMAT, &format	t);			
		format = FOR	MAT_24	4_HOUR;		
/* Set the RTC time format */						
<pre>Set_DL(RTC_TIME_FORMAT, &amp;format);</pre>						

# 7.8.6 RTC Alarm Date and Time

The SDK platform supports 2 alarms and they can be configured by the user during run time. To set an alarm the user will need to configure the below parameters of the alarm and then enable the SET\_ALARM.

Field ID	Data Type	Permission	Size	Descriptio n	Comments
RTC_ALARM_ A_SECOND	DBu8	READ/WRITE	1	0-59	Valid values to set the alarm are from 0 to 59. From 60 to 255 the values are don't care to set the alarm.
RTC_ALARM_ A_MINUTE	DBu8	READ/WRITE	1	0-59	Valid values to set the alarm are from 0 to 59. From 60 to 255 the values are don't care to set the alarm.
RTC_ALARM_ A_HOUR	DBu8	READ/WRITE	1	0-24	Valid values to set the alarm are from 0 to 24. From 25 to 255 the values are don't care to set the alarm.
RTC_ALARM_ A_DAY	DBu8	READ/WRITE	1	1-31	Valid values to set the alarm are from 1 to 31. From 32 to 255 the values are don't care to set the alarm.
RTC_ALARM_ A_WEEK_DA Y	DBu8	READ/WRITE	1	1-7	Valid values to set the alarm are from 1 to 7. From 8 to 255 the values are don't care to set the alarm.
RTC_ALARM_ A_MONTH	DBu8	READ/WRITE	1	1-12	Valid values to set the alarm are from 1 to 12. From 13 to 255 the values are don't care to set the alarm.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

RTC_ALARM_ A_YEAR	DBu8	READ/WRITE	1	00-99	Valid values to set the alarm are from 0 to 99. From 100 to 255 the values are don't care to set the alarm.
SET_ ALARM_A	DBu8	READ/WRITE	1	(ON/OFF)	(User need to set the above ALARM parameters and then enable the SET_ALARM1 to set the alarm time)

Field ID	Data Type	Permission	Size	Options	Description
RTC_ALARM_ B_SECOND	DBu8	READ/WRITE	1	0-59	Valid values to set the alarm are from 0 to 59. From 60 to 255 the values are don't care to set the alarm.
RTC_ALARM_ B_MINUTE	DBu8	READ/WRITE	1	0-59	Valid values to set the alarm are from 0 to 59. From 60 to 255 the values are don't care to set the alarm.
RTC_ALARM_ B_HOUR	DBu8	READ/WRITE	1	0-24	Valid values to set the alarm are from 0 to 24. From 25 to 255 the values are don't care to set the alarm.
RTC_ALARM_ B_DAY	DBu8	READ/WRITE	1	1-31	Valid values to set the alarm are from 1 to 31. From 32 to 255 the values are don't care to set the alarm.
RTC_ALARM_ B_WEEK_DAY	DBu8	READ/WRITE	1	1-7	Valid values to set the alarm are from 1 to 7. From 8 to 255 the values are don't care to set the alarm.
RTC_ALARM_ B_MONTH	DBu8	READ/WRITE	1	1-12	Valid values to set the alarm are from 1 to 12. From 13 to 255 the values are don't care to set the alarm.
RTC_ALARM_ B_YEAR	DBu8	READ/WRITE	1	00-99	Valid values to set the alarm are from 0 to 99. From 100 to 255 the values are don't care to set the alarm.
SET_ ALARM_B	DBu8	READ/WRITE	1	(ON/OFF)	(User need to set the above ALARM parameters and then enable the SET_ALARM2 variable)

The below code snippet show how we can set the alarm,

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Software		Date:	Oct 17, 2022

uint8\_t ahours; uint8\_t aminutes; uint8\_t aseconds; uint16\_t awkdays; uint8\_t ares; Set\_DL(*RTC\_ALARM\_A\_HOUR*, &ahours); Set\_DL(*RTC\_ALARM\_A\_HOUR*, &ahours); Set\_DL(*RTC\_ALARM\_A\_MINUTE*, &aminutes); Set\_DL(*RTC\_ALARM\_A\_SECOND*, &aseconds); Set\_DL(*RTC\_ALARM\_A\_WEEK\_DAY*, (uint8\_t \*)&awkdays); Set\_DL(*SET\_ALARM\_A*, &ares);

Once the alarm is set the user can read the ALARM\_STATUS DB variable to know the status of the alarm as seen in the below table. Once the alarm occurs the status variable will be updated to OCCURRED. After the user reads the status, he will need to reset the same in the DB.

Field ID	Data Type	Permission	Size	Description	Comments
ALARM_A_ST ATUS	DBu8	READ/WRITE	1	(1: OCCURRED/ 0: NOTOCCUR RED)	Alarm1status (OCCURRED/ NOTOCCURRED)
ALARM_B_ST ATUS	DBu8	READ/WRITE	1	(1: OCCURRED/ 0: NOTOCCUR RED)	Alarm2status (OCCURRED/ NOTOCCURRED)

The below code snippet shows the alarm status,

```
uint8_t alarm_status;
/* Read the Alarm A Status from the DB */
res = Get_DL(ALARM_A_STATUS, &alarm_status);
if (ALARM_OCCURED == alarm_status)
{
    alarm_status = 0;
    /* Set the ALARM A status*/
    res = Set_DL(ALARM_A_STATUS, &alarm_status);
  }
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

### 7.8.7 RTC Alarm Time Format

The SDK supports the 12- and 24-hour time format. The user can read/update the RTC Alarm format during run time using the below DB variables.

Below are their definitions,

<pre>#define</pre>	FORMAT_12_HOUR	1
<pre>#define</pre>	FORMAT_24_HOUR	0

Field ID	Data Type	Permission	Size	Descriptio n	Comments
RTC_ALARM_ A_TIME_FOR MAT	DBu8	READ/WRITE	1	AM/PM	RTC ALARMA time format (AM/PM)

Field ID	Data Type	Permission	Size	Descriptio n	Comments
RTC_ALARM_ B_TIME_FOR MAT	DBu8	READ/WRITE	1	AM/PM	RTC ALARMB time format (AM/PM)

### 7.8.8 RTC Sample Configuration

```
/*RTC Platform service Enable (PS_ENABLE) / Disable (PS_DISABLE) Macros */
#define SDK_SERVICE_RTC
                                                      PS_ENABLE
#if (SDK SERVICE RTC == PS ENABLE)
/*!
 * RTC Task Periodicity 100ms
 */
#define PS_RTC_TASK_TIMEOUT
                                                      100
/*!
 * RTC Task Priority
 * osPriorityNone
                          = 0,
 * osPriorityIdle
                         = 1,
                          = 8,
 * osPriorityLow
 * osPriorityLow1
                          = 8+1,
 *
           ر ر
                                           ر ر
 *
           ر ر
                                           ر ر
 * osPriorityISR
                          = 56,
 * osPriorityError
                         = -1,
 * osPriorityReserved
                          = 0x7FFFFFFF
 */
#define PS_RTC_TASK_PRIORITY
                                                           osPriorityIdle
#endif
         //SDK_SERVICE_RTC
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 7.9 EEPROM Module

The AI280 SDK User would be able to use the below functionalities of the EEPROM module via the DB variables and configuration file.

## 7.9.1 EPROM Module Enable/Disable

The SDK provides the user the ability to enable/disable the EEPROM functionality by modifying the default file. Please see section 7.9.5 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_E EPROM	PS_ENABLE/PS _DISABLE	PS_ENABLE	PS_ENABLE: - Enables the EEPROM module in the SDK PS_DISABLE: - Disables the EEPROM module in the SDK

# 7.9.2 EEPROM Time Out Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware and update it in the DB so that when the user reads the DB, he will receive the latest updated data if any or perform any other routine tasks. This default configuration can be done in the AI280\_config.h. Please see section 7.9.5 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_EE_TASK_TI MEOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommended Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the configured inputs in the Database.

# 7.9.3 EEPROM Module Task Priority

The AI280 SDK supports the below task priorities and the user can modify the task priority for the timer module in the configuration file. Please see section 6.9.5 for sample code snippet.

Sr. No	Variables	Options	Default State	Description
1	PS_EE_TASK_PR IORITY	osPriorityNone, osPriorityIdle, osPriorityLow, osPriorityLow1, osPriorityISR,	osPriorityIdle (Recommended Value)	User can select any one of the priorities based on the application requirement

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

osPriorityError,	
osPriorityReserved.	

#### 7.9.4 EEPROM Placeholder

The AI280 SDK supports the below size of the placeholder and the user can modify them in the configuration file. Please see section 7.9.5 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SIZE_OF_PLACE HOLDER	1 to 65535	100	User can select the size of the placeholder based on the application requirement

The SDK has currently defined 300 placeholders but the user can use 65535 placeholders. This can be used as a reference for all the additional elements that the user can use. The user would be able to read and write into the EEPROM places holder using the below DB variables. These place holders are defined in the EE\_PH\_DB.h file. The user can add additional variables here.

Field ID	Data Type	Permission	Size Bytes	Options	Description
EE_CAL01	EEPROM_t	READ/ WRITE	variable	Place holder for EEPROM variable defined in EEPROM map	Place holder 1
EE_CAL02	EEPROM_t	READ/ WRITE	variable	Place holder for EEPROM variable defined in EEPROM map	Place holder 2
EE_CAL03	EEPROM_t	READ/ WRITE	variable	Place holder for EEPROM variable defined in EEPROM map	Place holder 3
EE_CAL04	EEPROM_t	READ/ WRITE	variable	Place holder for EEPROM variable defined in EEPROM map	Place holder 4
EE_CAL05	EEPROM_t	READ/ WRITE	variable	Place holder for EEPROM variable defined in EEPROM map	Place holder 5
EE_CAL300	EEPROM_t	READ/ WRITE	variable	Place holder for EEPROM variable defined in EEPROM map	Considered placeholder for worst case scenario of each variable of 1 byte size

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

One the placeholder is defined is necessary to set the parameters of the EEPROM variable in the EE User define.h.

```
.*/#9
int8_t·EE_CAL01_default·=·10;#9
#9
EE_Element_info·····EE_user_elements[]·=¤9
{#9
····/*·ID,» Size,» CRC_enable,»» Redundancy,»» Default_data_enable,» Default_data.*/¤9
····{EE_CAL01·, sizeof(EE_CAL01_default), TRUE, 1, TRUE, %EE_CAL01_default},¤9
» {EE_CAL02·, sizeof(EE_CAL01_default), TRUE, 2, TRUE, %EE_CAL01_default},¤9
» {EE_CAL03·, sizeof(EE_CAL01_default), TRUE, 3, TRUE, %EE_CAL01_default},¤9
» {EE_CAL03·, sizeof(EE_CAL01_default), TRUE, 4, TRUE, %EE_CAL01_default},¤9
» {EE_CAL04·, sizeof(EE_CAL01_default), TRUE, 4, TRUE, %EE_CAL01_default},¤9
```

The parameters to be set is size, CRC enable, redundancy (multiple copies of the variable), enable default data and a pointer to the default data (if the reading of the variable fails it going to report the default data).

The functionality of the EEPROM platform service if all the parameters are enabled is the following: the data is going to be stored in the variable and the redundancy copies, if the principal variable fail to write the redundancy variable will be used until it fails and then a default value will be reported.

To make the EEPROM platform service update the values in the external EEPROM is necessary to set a break point in the core/Maximatecc/EEPROM\_Paltformservice.c in the following section.

```
158 void EEPROM Shadow Init()
159 {백
160 10
      uint32 t index; "
161 ....uint32_t val;
162 ····uint8_t··retrycount·=·10;¤5
163 ····uint8_t··*USER_Shadow_addr;¤¶
164 四明
165 ·····/* Read the EEPROM First page to check the Magic number */¤¶
166 ····/* Check Magic number is present in the EEPROM ····
                                                           */11=
/* If it is present, the EEPROM is already initialized */m
168 /* with the USER data */m
169 15
170 ····while(retrycount > 0)¤
171 ····{¤¶
172 .....
           TakeSPIBusLock();
173
           EEPROM_SPI_ReadBuffer((uint8_t *)&val, 0x00, MAGIC_NUMBER_SIZE);
174 .....
           GiveSPIBusLock();
           if (EEPROM_MAGIC_NUMBER == val)
175
176 .....
               break:
177 119
178 .....
           retrycount--;=
    ····}¤¶
179
180
181 if (EEPROM_MAGIC_NUMBER != val)
182
      ···{¤¶
            183
           Initialize_EEPlaceHolder(); #5
184
185 1
           Init_Shadow_memory();
       }¤¶
186
187 ....
        else
188
       {¤¶
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

After the breakpoint is reached in the window of local variables (upper right of the screen) the val value should be modified to make the EEPROM platform service format the external EEPROM, this is only necessary at debug stage and only when a new variable is defined.

							× :00   46	1 100 14
Ξ	(x)= V 53	©₀ B 6	🙀 E	🛋 M	🔜 Di	1010 R	ଟ୍ଟେ Li 📾 S	
							🖄 📲 🗖 🗖	8 🖻
	Name			Туре			Value	
	(x)= index		uint32_t			0x2404d1b8 (Hex)		
	(×)= val			uint32	2_t		0xac2eac20 (Hex)	
	(≫= retrycount >		uint8_t uint8_t *			10 "\n' 0x12c "öwxä1\2001, Ī\232Â[Ī\2		

To access the variables the user must use the START\_EEPROM value + offset. For example, to access the variable to access the variable  $EE\_CAL01$ , the user will use the OFFSET as START\_EEPROM + EE\_CAL01.

The sample code below gives an example to access the Placeholders for EEPROM.

```
uint8_t value;
if(KEY4_SHORT_PRESS == val)
      {
            Get_DL((START_EEPROM+ EE_CAL01),(uint8_t *)&value);
     }
    7.9.5 EEPROM Sample Configuration
/*!
 * EEPROM Platform service Enable (PS_ENABLE) / Disable (PS_DISABLE) Macros
*/
#define SDK SERVICE EEPROM
                                                       PS ENABLE
#if (SDK_SERVICE_EEPROM == PS_ENABLE)
/*!
 * EEPROM Task Priority
* osPriorityNone
                           = 0,
 * osPriorityIdle
                          = 1,
 * osPriorityLow
                           = 8,
 * osPriorityLow1
                           = 8+1,
            ر ر
                                            ر ر
            ر ر
                                            ر ر
 * osPriorityISR
                           = 56,
 * osPriorityError
                           = -1,
 * osPriorityReserved
                           = 0x7FFFFFF
 */
#define PS_EE_TASK_PRIORITY
                                                             osPriorityIdle
/*!
 * EEPROM Task Periodicity
                              100ms
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

<pre>*/ #define PS_EE_TASK_TIMEOUT /*! * EEPROM Place holder size</pre>	100
<pre>*/ #define SIZE_OF_PLACEHOLDER #endif //SDK_SERVICE_EEPROM</pre>	100

Dept: Software	Dept: Software User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 7.10 Watch Dog Module

The User would be able to use the below functionalities of the watch dog module via the DB variables and configuration file.

#### 7.10.1 Watch dog module Enable/Disable

The SDK provides the user the ability to enable/disable the watch dog module functionality by modifying the default configuration file. Please see section 7.10.7 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_ WATCHDOG	PS_ENABLE/ PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE: -</li> <li>Enables the watch dog module in the SDK</li> <li>PS_DISABLE: -</li> <li>Disables the watch dog module in the SDK</li> </ul>

## 7.10.2 Watch dog Time Out Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware and update it in the DB so that when the user reads the DB, he will receive the latest updated data if any else perform any other routine tasks. This default configuration can be done in the AI280\_config.h. Please see section 7.10.7 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_WD_TASK_TI MEOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommended Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the database.

# 7.10.3 Watch Dog Task Priority

The AI280 SDK supports the below task priorities, and the user can modify the task priority for the Watchdog module in the configuration file. Please see section 7.10.7 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_WD_TASK_P RIORITY	osPriorityNone, osPriorityIdle osPriorityLow, osPriorityLow1, osPriorityISR,	osPriority Idle (Recomm ended	User can select any one of the priorities based on the application requirement

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

osPriorityError,	Value)
osPriorityReserved	

## 7.10.4 Watchdog User Task Enable\Disable

The SDK provides the user the ability to enable/disable the watch dog functionality by modifying the default configuration file. Please see section 7.10.7 for sample configuration.

S.No	Variables	Options	Default State	Description
1	USER_TASK_WD0	PS_ENABLE/ PS_DISABLE	PS_DISABLE	PS_ENABLE: - Enables the user task#0 watchdog module in the SDK PS_DISABLE: - Disables the user task#0 watchdog module in the SDK
2	USER_TASK_WD1	PS_ENABLE/ PS_DISABLE	PS_DISABLE	PS_ENABLE: - Enables the user task#1 watchdog module in the SDK PS_DISABLE: - Disables the user task#1 watchdog module in the SDK
3	USER_TASK_WD2	PS_ENABLE/ PS_DISABLE	PS_DISABLE	PS_ENABLE: - Enables the user task#2 watchdog module in the SDK PS_DISABLE: - Disables the user task#2 watchdog module in the SDK
4	USER_TASK_WD3	PS_ENABLE/ PS_DISABLE	PS_DISABLE	<ul> <li>PS_ENABLE: - Enables the user task#3 watchdog module in the SDK</li> <li>PS_DISABLE: - Disables the user task#3 watchdog module in the SDK</li> </ul>

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

6       USER_TASK_WD5       PS_ENABLE/ PS_DISABLE       PS_DISABLE       PS_DISABLE       PS_DISABLE: - Enables the user watchdog module SDK         7       USER_TASK_WD6       PS_ENABLE/ PS_DISABLE       PS_DISABLE       PS_ENABLE: - Disables the user watchdog module SDK         7       USER_TASK_WD6       PS_ENABLE/ PS_DISABLE       PS_DISABLE       PS_ENABLE: - Disables the user watchdog module SDK	task#4 in the task#4 in the
7       USER_TASK_WD6       PS_ENABLE/ PS_DISABLE       PS_DISABLE       PS_DISABLE       PS_DISABLE       PS_DISABLE: - Enables the user watchdog module Disables the user watchdog module	task#5 in the task#5
SDK	in the task#6
8 USER_TASK_WD7 PS_ENABLE/ PS_DISABLE PS_DISABLE PS_DISABLE PS_DISABLE: - Enables the user watchdog module SDK PS_DISABLE: - Disables the user watchdog module SDK	in the task#7
9       USER_TASK_WD8       PS_ENABLE/ PS_DISABLE       PS_DISABLE       PS_DISABLE       Enables the user watchdog module SDK         9       USER_TASK_WD8       PS_ENABLE/ PS_DISABLE       PS_DISABLE       PS_DISABLE       Enables the user watchdog module SDK         10       USER TASK WD9       PS ENABLE/ PS ENABLE/       PS DISABLE       PS ENABLE: -	in the task#8

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

PS_DISABLE	Enables the user task#9
	watchdog module in the
	SDK
	<b>PS_DISABLE:</b> -
	Disables the user task#9
	watchdog module in the
	SDK

### 7.10.5 Watchdog Feed Timer Configuration

Watchdog is used for automatic correction of temporary hardware/software faults by resetting the MCU. The AI280 SDK allows the user to configure the watchdog timer. Once this timer expires the watchdog service would check if all the registered tasks are if any of the tasks has not ping the watchdog service then it would reset the MCU. This timer value can be configured using the below parameter. Please see section 7.10.7 for sample configuration.

The user can configure the watchdog timer with different prescaler values as supported by the platform and they correspond to equivalent time. For example, when configured as IWDG\_PRESCALER\_256 the watchdog module expects to be refreshed every 40-50 secs else it would reset the MCU.

Sr. No	Variables	Options	Default State	Description
1	WATCHDO G_FEED_TI ME	IWDG_PRESCALER_4 IWDG_PRESCALER_8 IWDG_PRESCALER_16 IWDG_PRESCALER_32 IWDG_PRESCALER_64 IWDG_PRESCALER_128 IWDG_PRESCALER_256	IWDG_PRE SCALER_25 6	Watchdog feed time triggers a reset sequence when it is not refreshed within the expected time window

### 7.10.6 Watchdog Ping Functionality

The SDK watchdog service will reset the MCU if it finds that any of the threads are not functional. Hence as a user task it would be the user's responsibility to keep pinging the watchdog service and updating the alive status. During runtime, the user can write to the below DB variable to report the alive status to the watchdog service.

Each user task has a corresponding watchdog ping variable that it needs to update. For example, user task 1 would use the WDO\_PING variable as it has enabled the USER\_TASK\_WD0 variable in the configuration file.

Dept: Software	User Manual Document	Rev No. Re	Rev 3.0
		Date:	Oct 17, 2022

Field ID	Data Type	Permission	Size	Description	Comments
WD0_PIN G	DBu8	READ/WRI TE	1	TASK_ID (1-10)	Setting the task ID to the WDO_PING variable informs the platform service that task 0 is alive.
WD1_PIN G	DBu8	READ/WRI TE	1	TASK_ID (1-10)	Setting the task ID to the WD1_PING variable informs the platform service that task 1 is alive.
WD2_PIN G	DBu8	READ/WRI TE	1	TASK_ID (1-10)	Setting the task ID to the WD2_PING variable informs the platform service that task 2 is alive.
WD3_PIN G	DBu8	READ/WRI TE	1	TASK_ID (1-10)	Setting the task ID to the WD3_PING variable informs the platform service that task 3 is alive.
WD4_PIN G	DBu8	READ/WRI TE	1	TASK_ID (1-10)	Setting the task ID to the WD4_PING variable informs the platform service that task 4 is alive.
WD5_PIN G	DBu8	READ/WRI TE	1	TASK_ID (1-10)	Setting the task ID to the WD5_PING variable informs the platform service that task 5 is alive.
WD6_PIN G	DBu8	READ/WRI TE	1	TASK_ID (1-10)	Setting the task ID to the WD6_PING variable informs the platform service that task 6 is alive.
WD7_PIN G	DBu8	READ/WRI TE	1	TASK_ID (1-10)	Setting the task ID to the WD7_PING variable informs the platform service that task 7 is alive.
WD8_PIN G	DBu8	READ/WRI TE	1	TASK_ID (1-10)	Setting the task ID to the WD8_PING variable informs the platform service that task 8 is alive.
WD9_PIN G	DBu8	READ/WRI TE	1	TASK_ID (1-10)	Setting the task ID to the WD9_PING variable informs the platform service that task 9 is alive.

The sample code gives an example to ping for user task 5,

uint8\_t state;
if(user\_task\_wd5 == 1)

Dept: Software User Manual I	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

```
{
#if(USER TASK WD5 == PS ENABLE)
      state = 6; // where 6 is the task ID
      Set_DL(WD5_PING , &state);
#endif
}
        7.10.7 Watchdog Sample Configurations
/*!
 * Watchdog Platform service Enable (PS_ENABLE) / Disable (PS_DISABLE) Macros
*/
#define SDK_SERVICE_WATCHDOG
                                                        PS_DISABLE
#if (SDK_SERVICE_WATCHDOG == PS_ENABLE)
/*!
* Watchdog Task Periodicity
                                    100ms
*/
#define PS_WD_TASK_TIMEOUT
                                                        100
/*!
* Watchdog Task Priority
* osPriorityNone
                           = 0,
 * osPriorityIdle
                           = 1,
* osPriorityLow
                              8,
                             8+1,
 * osPriorityLow1
*
           • •
                                             ,,
 *
            , ,
                                             ر ر
* osPriorityISR
                           = 56,
 * osPriorityError
                           = -1,
* osPriorityReserved
                           = 0x7FFFFFFF
*/
#define PS_WD_TASK_PRIORITY
                                                              osPriorityIdle
/*!
* Watchdog Reset timer
*/
#define PS_WD_RESET_TIMER
                                                        500
/*
* IWDG PRESCALER 4
* IWDG_PRESCALER_8
 * IWDG PRESCALER 16
 * IWDG_PRESCALER_32
* IWDG PRESCALER 64
* IWDG PRESCALER 128
 * IWDG PRESCALER 256
*/
#define WATCHDOG FEED TIME
                                                        IWDG PRESCALER 256
```

Dept: Software	User Manual Document	Rev No.         Rev 3.0           Date:         Oct 17, 2022	Rev 3.0
			Oct 17, 2022

/*!	
* <u>Watchdog</u> external task ping_id	
*	
* MAX Supported USER Watchdog is 10	
*/	
#define USER_TASK_WD0	PS_DISABLE
#define USER_TASK_WD1	PS_DISABLE
#define USER_TASK_WD2	PS_DISABLE
#define USER_TASK_WD3	PS_DISABLE
#define USER_TASK_WD4	PS_DISABLE
#define USER_TASK_WD5	PS_DISABLE
#define USER_TASK_WD6	PS_DISABLE
#define USER_TASK_WD7	PS_DISABLE
#define USER_TASK_WD8	PS_DISABLE
#define USER_TASK_WD9	PS_DISABLE
<pre>#endif //SDK_SERVICE_WATCHDOG</pre>	

Dept: Software	User Manual Document	Rev No. Rev 3.	Rev 3.0
		Date:	Oct 17, 2022

## 7.11 Power Mode Module

The User would be able to use the below functionalities of the Power Mode module via the DB variables and configuration file.

### 7.11.1 Power mode module Enable/Disable

The SDK provides the user the ability to enable/disable the power mode functionality by modifying the default configuration file. Please see section 7.11.7 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_P M	PS_ENABLE PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE: -</li> <li>Enables the power mode module in the SDK</li> <li>PS_DISABLE: -</li> <li>Disables the power mode module in the SDK</li> </ul>

## 7.11.2 Power Mode Time Out Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware and update it in the DB so that when the user reads the DB, he will receive the latest updated data if any and perform any other routine tasks. This default configuration can be done in the AI280\_config.h. Please see section 7.11.7 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_PM_TASK_TI MEOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommended Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the database.

## 7.11.3 Power Mode task Priority

The AI280 SDK supports the below task priorities and the user can modify the task priority for the Power Mode module in the configuration file. Please see section 7.11.7 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_PM_TASK_PR IORITY	osPriorityNone, osPriorityIdle, osPriorityLow, osPriorityLow1,	osPriorityIdle (Recommended Value)	User can select any one of the priorities based on the application requirement

Dept: Software	User Manual Document	Rev No. Rev 3.0	Rev 3.0
	Oser manual Document	Date:	Oct 17, 2022

o Dri o nitri ISD	
osphorityISK,	
osPriorityError,	
osPriorityISR, osPriorityError, osPriorityReserve	
d	

## 7.11.4 Power Mode Wake Up Source Configuration

The AI280 SDK allows the user to configure the wake-up source, so that the device can exit from the low power mode. To do so, he can configure the below parameters in the configuration file. This default configuration can be done in the AI280\_config.h. Please see section 7.11.7 for sample configuration.

The platform support wakes up from the below sources and they can be configured using the below parameters.

- 1) Keypad
- 2) RTC
- 3) Ignition
- 4) CAN

Sr. No	Variables	Options	Default State	Description
1	KEYPAD01_WAK EUP_SOURCE	PS_ENABLE PS_DISABLE	PS_ENABLE	<b>PS_ENABLE: -</b> Enables the keypad 1 as a wake-up source to exit low power mode. <b>PS_DISABLE: -</b> Disables the keypad 1 as a wake-up source to exit low power mode.
2	KEYPAD02_WAK EUP_SOURCE	PS_ENABLE PS_DISABLE	PS_ENABLE	<b>PS_ENABLE: -</b> Enables the keypad 2 as a wake-up source to exit low power mode. <b>PS_DISABLE: -</b> Disables the keypad 2 as a wake-up source to exit low power mode.
3	KEYPAD03_WAK EUP_SOURCE	PS_ENABLE PS_DISABLE	PS_ENABLE	<b>PS_ENABLE:</b> - Enables the keypad 3 as a wake-up source to exit low power mode. <b>PS_DISABLE:</b> - Disables the keypad 3 as a wake-up source to exit low power mode.
4	KEYPAD04_WAK EUP_SOURCE	PS_ENABLE PS_DISABLE	PS_ENABLE	<b>PS_ENABLE: -</b> Enables the keypad 4 as a wake-up source to exit low power mode. <b>PS_DISABLE: -</b> Disables the keypad 4 as a wake-up source to exit low power mode.
5	RTC_WAKEUP_S OURCE_STATE	PS_ENABLE	PS_DISABLE	<b>PS_ENABLE: -</b> Enables the RTC as a wake-up source to exit low power mode. <b>PS_DISABLE: -</b>

Dept: Software	User Manual Document	Rev No.	Rev 3.0
	Osci Manual Document	Date:	Oct 17, 2022

		PS_DISABLE		Disables the RTC as a wake-up source to exit low power mode.
6	IGN_WAKEUP_S OURCE	PS_ENABLE PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE: -</li> <li>Enables the ignition as a wake-up source to exit low power mode.</li> <li>PS_DISABLE: -</li> <li>Disables the ignition as a wake-up source to exit low power mode.</li> </ul>
7	CAN_WAKEUP_S OURCE	PS_ENABLE PS_DISABLE	PS_ENABLE	<b>PS_ENABLE: -</b> Enables the CAN as a wake-up source to exit low power mode. <b>PS_DISABLE: -</b> Disables the CAN as a wake-up source to exit low power mode.

#### 7.11.5 Power Mode RTC Timeout

The AI280 SDK allows the user to configure the RTC as a wake-up source, so that the device can exit from the low power mode. He also can set the timeout for the RTC to wake up the system. To do so, he can configure the below parameters in the configuration file. This default configuration can be done in the AI280\_config.h. Please see section 7.11.7 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	RTC_WAKEUP_S OURCE_TIMEOU T	MIN VALUE: 0 MAX VALUE: 65535	10000ms (Recommende d Value)	User can set the RTC wake up timeout using the configuration.

### 7.11.6 Power Mode Enable

The user would be able to enter the power mode during runtime using the below the DB variable (POWER MODE). The platform supports the below three power mode configurations.

#### 1) STOP MODE

Field ID	Data Type	Permission	Size Bytes	Description	Comments
POWER_MODE	DBu8	READ/WRI TE	1	STOP	This field is used to set and read the Power mode configuration

The below code snippet shows how the user can enable the different power mode configuration.

```
pm_state = PM_STOP_MODE;
Set_DL(POWER_MODE, &pm_state);
```

} #endif

#### 7.11.7 Power Mode default Configurations

```
/*!
* Power Management Platform service Enable (PS ENABLE) / Disable
(PS_DISABLE) Macros
*/
#define SDK SERVICE PM
                                                 PS_ENABLE
#if (SDK SERVICE PM == PS ENABLE)
/*!
* Power Management Task Periodicity
                                          100ms
*/
                                                 100
#define PS_PM_TASK_TIMEOUT
/*!
* GPIO Wake up Source
*/
#define KEYPAD01_WAKEUP_SOURCE
                                                 PS_ENABLE
#define KEYPAD02 WAKEUP SOURCE
                                                 PS ENABLE
#define KEYPAD03_WAKEUP_SOURCE
                                                 PS_ENABLE
#define KEYPAD04_WAKEUP_SOURCE
                                                 PS_ENABLE
/*!
* RTC Wake up Source
*/
#define RTC_WAKEUP_SOURCE_STATE
                                                 PS_DISABLE
#define RTC_WAKEUP_SOURCE_TIMEOUT
                                                 10000
/*!
* IGN Wake up Source
*/
                                                             PS ENABLE
#define IGN WAKEUP SOURCE
#endif //SDK_SERVICE_PM
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
	twart Osti Manual Document	Date:	Oct 17, 2022

## 7.12 LCD Module

The User would be able to use the below functionalities of the LCD module via the DB variables and configuration file.

## 7.12.1 LCD mode module Enable/Disable

The SDK provides the user the ability to enable/disable the LCD functionality by modifying the default configuration file. Please see section 7.12.6 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_L CD	PS_ENABLE PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE: -</li> <li>Enables the lcd module in the SDK</li> <li>PS_DISABLE: -</li> <li>Disables the lcd module in the SDK</li> </ul>

## 7.12.2 LCD Module Timeout Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware and update it in the DB so that when the user reads the DB, he will receive the latest updated data if any and perform any other routine tasks. This default configuration can be done in the AI280\_config.h. Please see section 7.12.6 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_LCD_TASK_T IMEOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommended Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the database.

## 7.12.3 LCD task Priority

The AI280 SDK supports the below task priorities and the user can modify the task priority for the LCD module in the configuration file. Please see section 7.12.6 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_LCD_TASK_P RIORITY	osPriorityNone, osPriorityIdle, osPriorityLow, osPriorityLow1, osPriorityISR, osPriorityError, osPriorityReserved	osPriorityIdle (Recommended Value)	User can select any one of the priorities based on the application requirement

Dept: Software	User Manual Document	Rev No.	Rev 3.0
· · · · · · · · · · · · · · · · · · ·		Date:	Oct 17, 2022

## 7.12.4 LCD State

The AI280 SDK supports the user to configure the default state of the LCD (either OFF/ON) and this can be done by modifying the below parameter in the configuration file. Please see section 7.12.6 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	CONF_LCD_STA TE	LCD_CONF_ON LCD_CONF_OF F	LCD_CONF_ON	User can configure the LCD state as ON/OFF.

The user can also get/set the default state of the LCD (either OFF/ON) during runtime using the below DB variable

Field ID	Data Type	Permission	Size	Description	Comments
LCD_STATE	DBu8	READ/WRI TE	1	ON/OFF	This field sets and reads back the Turn ON or OFF the LCD display

The below code snippet shows how you can set or get the LCD state,

```
uint8_t state;
/* Get the LCD State value from the DB */
Get_DL(LCD_STATE , &state);
if (LCD_CONF_OFF == state)
    {
       state = LCD_CONF_ON;
       /* Set the LCD state to ON */
       Set_DL(LCD_STATE, &state);
    }
```

## 7.12.5 LCD Brightness

The AI280 SDK supports the user to configure the LCD brightness, and this can be done by modifying the below parameter in the configuration file. Please see section 7.12.6 for sample configuration.

The below configuration means the screen is at 30% brightness level. If the user needs full brightness, then it will need to set it at 100.

Sr. No	Variables	Options	Default State	Description
1	CONF_LCD_BRI GHTNESS	0-100	30%	User can configure the LCD brightness

The user would be able to read and modify the below functionalities of the LCD module via the DB variables and configuration file.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Field ID	Data Type	Permission	Size	Description	Comments
LCD_BRIGHT NESS	DBu8	READ/WRI TE	1	0-100	This field sets the percentage of brightness from 0 to 100 (full brightness) for the LCD.

The sample code gives an example to set the brightness of the LCD.

```
uint16_t brightness_value;
uint8_t state;
void LCDView::brightnessinc()
{
    #if (SDK_SERVICE_LCD == PS_ENABLE)
    brightness_value++;
    if (!(LCD_BRT_MAX >= brightness_value))
    brightness_value = LCD_BRT_MAX;
    Set_DL(LCD_BRIGHTNESS, (uint8_t *)&brightness_value);
    #endif
    }
}
```

The sample code gives an example to get the brightness of the LCD.

\* LCD Task Priority \* osPriorityNone

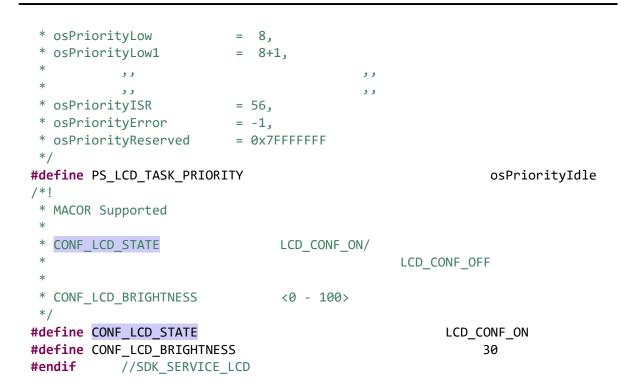
\* osPriorityIdle

```
LCDView::LCDView()
{
#if (SDK_SERVICE_LCD == PS_ENABLE)
      brightness value = 0;
      /* Get the LCD Brightness value from the DB */
     Get_DL(LCD_STATE , &state);
     Get_DL(LCD_BRIGHTNESS, (uint8_t *)&brightness_value);
#endif
}
      7.12.6 LCD Sample Configurations
/*!
* LCD Platform service Enable (PS_ENABLE) / Disable (PS_DISABLE) Macros
*/
#define SDK_SERVICE_LCD
                                                       PS_ENABLE
#if (SDK SERVICE LCD == PS ENABLE)
/*!
* LCD Task Periodicity 100ms
*/
#define PS_LCD_TASK_TIMEOUT
                                                              100
/*!
```

= 0,

= 1,

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022



## 7.13 CAN Module

The SDK supports one can channel CAN1.

Please note that standalone CAN module will be disabled when J1939 is enabled in the configuration file. To use CAN in a standalone mode J1939 has to be disabled in the configuration file.

Below code snippet from the AI280\_config.h that shows the same,

#if ((SDK\_SERVICE\_J1939 == PS\_ENABLE) && (SDK\_SERVICE\_FDCAN == PS\_ENABLE))

#undef SDK\_SERVICE\_FDCAN

#define SDK\_SERVICE\_FDCAN

PS\_DISABLE

#endif

## 7.13.1 CAN Module Configuration Support

The SDK provides the user the ability to enable/disable the CAN functionality by modifying the default configuration file. Please see section 7.13.13 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_F DCAN	PS_ENABLE PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE: -</li> <li>Enables the FDCAN module in the SDK</li> <li>PS_DISABLE: -</li> <li>Disables the FDCAN module in the SDK</li> </ul>

### 7.13.2 CAN Enable/Disable

The SDK provides the user the ability to enable/disable the CAN functionality by modifying the default configuration file. Please see section 7.13.13 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	FDCAN1_ENABL E	PS_ENABLE PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE: -</li> <li>Enables the FDCAN1 module in the SDK</li> <li>PS_DISABLE: -</li> <li>Disables the FDCAN1 module in the SDK</li> </ul>

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Software Oser Manual Document	Date:	Oct 17, 2022	

## 7.13.3 CAN Module Timeout Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware and update it in the DB so that when the user reads the DB, he will receive the latest updated data if any or perform any other routine tasks. This default configuration can be done in the AI280\_config.h. Please see section 7.13.13 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_FDCAN_TAS K_TIMEOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommended Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the database.

## 7.13.4 CAN task Priority

The AI280 SDK supports the below task priorities and the user can modify the task priority for the CAN module in the configuration file. Please see section 6.13.13 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_FDCAN_TAS K_PRIORITY	osPriorityNone, osPriorityIdle, osPriorityLow, osPriorityLow1, osPriorityISR, osPriorityError, osPriorityReserved	osPriorityIdle (Recommended Value)	User can select any one of the priorities based on the application requirement

## 7.13.5 CAN Baudrate

The AI280 SDK supports the below Baud rate and the user can modify the Baud rates for the CAN module in the configuration file. Please see section 7.13.13 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	FDCAN1_BAUDR ATE	BAUDRATE_50K/ BAUDRATE_100/ BAUDRATE_125 /BAUDRATE_250 /BAUDRATE_500 /BAUDRATE_100 0	BAUDRATE_5 0K	User can set the Baudrate for CAN1

The AI280 SDK user can change the baud rate during runtime using the below DB variables,

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Field ID	Data Type	Permis sion	Size	Options	Description
CAN_CH0_BAUDR ATE	DBu32	READ/ WRITE	1	Default 250Kbaud Supported baud rates: AUTO, 50K, 100K, 125K, 250K, 500K, 1M	CAN Channel0 Baud-rate

The below code snippet shows how the baud rate can be changed during the runtime,

```
uint8_t can1_buf[10];
```

```
/*
```

- \* CAN1 Supporting baud rate
- \* BAUDRATE\_50K
- \* BAUDRATE\_100K
- \* BAUDRATE\_125K
- \* BAUDRATE\_250K
- \* BAUDRATE\_500K
- \* BAUDRATE\_1000K

```
*/
```

can1\_buf[0] = BAUDRATE\_250K; Set\_DL(CAN\_CH0\_BAUDRATE, &can1\_buf[0]); break;

## 7.13.6 CAN Identifier Configurations

The AI280 SDK supports the below configuration parameters for the CAN and the user can modify the same in the configuration file. Please see section 7.13.13 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	FDCAN1_IDENTI FIER	User Configurable ID	0x19FEFC FE	User can configure the CAN1 Identifier
2	FDCAN1_IDTYPE	FDCAN_EXTENDED_ID/ FDCAN_STANDRD_ID	FDCAN_ EXTEND ED_ID	User can configure the CAN1 IDTYPE as Extended/Standard
3	FDCAN1_ID	FDCAN_STANDARD_ID/	FDCAN_ STANDA	User can configure the CAN1

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

FDCAN_EXTENDED_ID	RD_ID	ID

## 7.13.7 CAN Channel configurations

The AI280 SDK supports the below filter properties for the CAN module in the configuration file. Please see section 7.13.13 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	FDCAN1_RXBUF FERIND	0-65535	1	User can configure the Rx buffer
2	FDCAN1_TXFRA METYPE	FDCAN_DATA_FRAME/ FDCAN_REMOTE_FRAM E	FDCAN_DATA_ FRAME	User can configure the TX frame type as Data Frame or Remote Frame
3	FDCAN1_DATAL ENGTH	FDCAN_DLC_BYTES_0 FDCAN_DLC_BYTES_1 FDCAN_DLC_BYTES_2 FDCAN_DLC_BYTES_3 FDCAN_DLC_BYTES_4 FDCAN_DLC_BYTES_5 FDCAN_DLC_BYTES_6 FDCAN_DLC_BYTES_7 FDCAN_DLC_BYTES_12 FDCAN_DLC_BYTES_12 FDCAN_DLC_BYTES_16 FDCAN_DLC_BYTES_20 FDCAN_DLC_BYTES_24 FDCAN_DLC_BYTES_32 FDCAN_DLC_BYTES_48 FDCAN_DLC_BYTES_48 FDCAN_DLC_BYTES_48	FDCAN_DLC_B YTES_8	User can configure the length of the data
4	FDCAN1_ERROR STATEIND	FDCAN_ESI_ACTIVE/ FDCAN_ESI_PASSIVE	FDCAN_ESI_AC TIVE	User can configure The errors state as Active or Passive
5	FDCAN1_BITRA TESWITCH	FDCAN_BRS_ON/FDCAN _BRS_OFF	FDCAN_BRS_O N	User can configure the Bit rate switch on/off
6	FDCAN1_FDFOR MATE	FDCAN_CLASSIC_CAN FDCAN_FD_CAN	FDCAN_FD_CA N	User can configure the FDCAN format.
7	FDCAN1_TXEVE NTFIFOCONTRO L	FDCAN_STORE_TX_EVE NTS/ FDCAN_NO_TX_EVENTS	FDCAN_STORE _TX_EVENTS	User can configure the event FIFO control
8	FDCAN1_MESSA GEMARKER	0-65535	0	User can configure message marker

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

	FDCAN1_RECEI			User can configure the
9	VE_TASK_DELA	$0-500 \mathrm{~ms}$	100 ms	delay for the receive task
	Y			

## 7.13.8 CAN Filter Configurations

The AI280 SDK supports the CAN Filter configurations, and the user can modify the same in the configuration file. Please see section 7.13.13 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	FDCAN1_FILT ERINDEX	0-65535	0	User can configure the filter index
2	FDCAN1_FILT ERTYPE	FILTER_DUAL or FILTER_RANGE or FILTER_MASK or FILTER_RANGE_NO_EIDM	FDCAN_FILTER _DUAL	User can configure the filter Type
3	FDCAN1_FILT ERCONFIG	FDCAN_FILTER_DISABLE FDCAN_FILTER_TO_RXFIFO0 FDCAN_FILTER_TO_RXFIFO1 FDCAN_FILTER_REJECT FDCAN_FILTER_HP FDCAN_FILTER_TO_RXFIFO1_HP FDCAN_FILTER_TO_RXBUFFER	FDCAN_FILTER _TO_RXBUFFE R	User can configure the filter
4	FDCAN1_FILT ERID1	0-65535	FDCAN_DEFAU LT_FILTERID1	User can configure the default filter id 1
5	FDCAN1_FILT ERID2	0-65535	FDCAN_DEFAU LT_FILTERID2	User can configure the default filter id 2

The user can also read and write to the below CAN filter properties during the runtime using the CAN DB variables,

Field ID	Data	Permission	Size	Description	Comments
	Туре				
CAN_CH0_FI LTER_INDEX _ENABLE	DBu8	READ/WRITE	1	Enable / Disable receive data with filter. Index range 0-32 (CAN_MODE_EXTENDED_ID   CAN_MODE_BUS_MONITORING_ EXTENDED_ID) Index range 0-64 (CAN_MODE_STANDARD_ID	CAN Channel0 filter index state

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

				CAN_MODE_BUS_MONITORING_ STANDARD_ID)	
CAN_CH0_FI LTER_INDEX _ID	DBu32	READ/WRITE	4	FIFO ID use to filter	CAN Channel0 filter index ID
CAN_CH0_FI LTER_INDEX _IDMASK	DBu32	READ/WRITE	4	ID MASK use to filter	CAN Channel0 index ID Mask

The below code sample show how the filter index is updated from the application,

uint8\_t can1\_buf[10]; can1\_buf[0] = ENABLE; Set\_DL(CAN\_CH0\_FILTER\_INDEX\_ENABLE, &can1\_buf[0]); can1\_buf[0] = 10; Set\_DL(CAN\_CH0\_FILTER\_INDEX\_ID, &can1\_buf[0]);

## 7.13.9 CAN Receive Task Delay

The AI280 SDK supports the configuration of the CAN Task Delay and the user can modify the CAN1 Receive Task Delay for the CAN module in the configuration file. Please see section 7.13.13 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	FDCAN1_RECEIVE_ TASK_DELAY	0-500 MS	100ms (Recommended Value)	User can configure the delay for the receive task

### 7.13.10 CAN Channel Modes and States

The user would be able to query the database to get the mode and current state of the CAN channels using the below CAN module DB variables.

Field ID	Data Type	Permis sion	Size	Options	Description
CAN_CH0 _MODE	DBu8	READ/ WRITE	1	Default CAN_MODE_EXTENDED_ID Options available: CAN_MODE_STANDARD_ID, CAN_MODE_EXTENDED_ID, CAN_MODE_BUS_MONITORING_STAND ARD_ID,	This field sets and reads the CAN Channel0 Modes

Dept: Softwa	re

				CAN_MODE_BUS_MONITORING_EXTEN DED_ID,	
CAN_CH0 _STATES	DBu8	READ	1	CAN States available: CAN_BUS_OFF, CAN_BUS_ON, CAN_STATE_PASSIVE, CAN_STATE_UNCHANGED,	This field is used to read the CAN Channel 0 state
CAN_CH0 _COMM_ STATE_E VENTS	DBu8	READ	1	Communication state events available: STATE_EVENT_NONE, STATE_EVENT_BUS_OFF, STATE_EVENT_BUS_OFF RECOVERY, STATE_EVENT_BUS_ON, STATE_EVENT_PASSIVE, STATE_EVENT_ACTIVE, STATE_EVENT_OVERRUN, STATE_EVENT_OVERRUN, STATE_EVENT_QUEUE_FULL, STATE_EVENT_QUEUE_FULL, STATE_EVENT_QUEUE_EMPTY, STATE_EVENT_DRIVER_ERROR	This field is used to read the CAN Channel0 communicatio n state event

The below code snippet shows how we can access the CAN mode and states,

```
uint8_t can1_buf[10];
/*
 * Get the CAN1 state and update the mode
*/
Get_DL(CAN_CH0_STATES, &can1_buf[0]);
can1_buf[0] = CAN_MODE_STANDARD_ID;
Set_DL(CAN_CH0_MODE, &can1_buf[0]);
/*
 * Get the CAN1 communication state event
*/
Get_DL(CAN_CH0_COMM_STATE_EVENTS, &can1_buf[0]);
```

## 7.13.11 CAN Channel Reset

The user would be able to reset the CAN channel and CAN driver using the below CAN module DB variables.

Field ID	Data Type	Permis sion	Size	Options	Description
CAN_CH0_RE SET	DBu8	READ/ WRITE	1	TRUE = Resets the CAN controller and Bus Off mode.	This variable supports the reset of the CAN Channel 0

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

CAN_CH0_DF IVER_RESET	DBu8	READ/ WRITE	1	TRUE = Reinitialize the CAN driver if Driver Error is Set.	This variable supports the reset of the CAN Channel 0 driver
--------------------------	------	----------------	---	--	---

The below code snippets show how the user can reset the CAN channel and driver,

```
uint8_t can1_buf[10];
/*
 * Reset the CAN CHANNEL 0
*/
case 4:
    Can1_buf[0] = TRUE;
    Set_DL(CAN_CH0_RESET, &can1_buf[0]);
    break;
/*
 * Reset the CAN Channel Driver 0
*/
case 5:
    Can1_buf[0] = TRUE;
    Set_DL(CAN_CH0_DRIVER_RESET, &can1_buf[0]);
    break;
```

### 7.13.12 CAN module RX/TX

The AI280 SDK allows the users to use the CAN channel to send or receive data. To do so please use the below variables.

To read incoming data over the CAN channel, the user will need to monitor the RX DATA SIZE variable and see if there is any pending data available and if yes read the data.

To send data over the CAN channel, the user will fill the TX buffer and send the data over CAN. The SDK will take care of handling the pending data.

The CAN default data packet size is defined as 64 bytes hence the user is expected to create a buffer of this size while reading the data.

Field ID	Data	Permission	Size	Description	Comments						
	Туре										
CAN1_RX_BY	DBu8	READ	1	CAN1 RX Byte count	CAN Channel0 RX						
TE_COUNT	DBuo	KLAD	1	value	Byte count						
CAN1_TX_BY	DBu8	READ	1	CAN1 TX Byte count	CAN Channel0 TX						
TE_COUNT	DBuo	KLAD	1	value	Byte count						
CAN1_RX_DA					CAN channel0 RX data						
TA_IS_AVAIL	DBu8	READ	READ	READ	READ	READ	READ	READ	1	CAN1 data available flag	available flag
ABLE					available flag						
CAN1_RX_DA	DBu8	READ	1	CAN1 RX data size	CAN Channel0 RX data						
TA_SIZE	DBuo	KLAD	1	CAINT INA UATA SIZE	size						

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

The sample code for sending and receiving the data from CAN is shown below,

```
uint8_t can_rxbuffer1[64];
      uint8 t can1 buf[10];
if(CAN BUS OFF != can1 buf[0])
{
      Get DL(CAN1 RX DATA IS AVAILABLE, &can1 buf[0]);
}
#if (SDK_SERVICE_FDCAN == PS_ENABLE)
uint8 t status;
/* Get the CAN RX status */
Get_DL(CAN1_RX_DATA_SIZE, &status);
if(0 != status)
{
      memset(&can rxbuffer1[0], 0x00, sizeof(can rxbuffer1));
      /* Read the Rx data from the DB */
      Get_DL(CAN1_RX_DATA, (uint8_t *)&can_rxbuffer1[0]);
}
/*To send the data over the CAN bus, set the CAN tx data */
Set DL(CAN1 TX DATA , (uint8 t *)&can1buf[0]);
```

## 7.13.13 CAN Sample Configuration.

```
*
*
*
                FDCAN Module
/*!
* FDCAN Platform service (PS_ENABLE) / Disable (PS_DISABLE) Macros
*/
#define SDK_SERVICE_FDCAN
                                           PS ENABLE
#if (SDK SERVICE FDCAN == PS ENABLE)
#define FDCAN1_ENABLE
                                       PS ENABLE
/*!
* FDCAN Task <u>Periodicity</u>
                         100ms
*/
#define PS_FDCAN_TASK_TIMEOUT
                                       100
/*!
* FDCAN Task Priority
* osPriorityNone
                   = 0,
* osPriorityIdle
                   = 1,
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

<pre>* osPriorityLow</pre>	= 8,		
* osPriorityLow1	= 8+1,		
ر د د		و و	
*			
* osPriorityISR	= 56,	و و	
* osPriorityError	= -1,		
<ul> <li>* osPriorityReserved</li> </ul>			
	= 0X/FFFFFFF		
#define PS_FDCAN_TASK_PRI			osPriorityIdle
<pre>#if (FDCAN1_ENABLE == PS_</pre>	_ENABLE)		
/*			
* BAUDRATE_50K			
* BAUDRATE_100K			
* BAUDRATE_125K			
* BAUDRATE_250K			
* BAUDRATE_500K			
* BAUDRATE_1000K			
*/			
<pre>#define FDCAN1_BAUDRATE</pre>			BAUDRATE_50K
#define FDCAN1_IDENTIFIER	{		Øx19FEFCFE
//0x111			
<pre>#define FDCAN1_IDTYPE</pre>			FDCAN_EXTENDED_ID
<pre>#define FDCAN1_TXFRAMETYF</pre>	ΡE		FDCAN_DATA_FRAME
<pre>#define FDCAN1_DATALENGTH</pre>	ł		FDCAN_DLC_BYTES_8
#define FDCAN1_ERRORSTATE	IND		FDCAN_ESI_ACTIVE
<pre>#define FDCAN1_BITRATESWI</pre>	ТСН		FDCAN_BRS_ON
<b>#define</b> FDCAN1_FDFORMATE			
<b>#define</b> FDCAN1_TXEVENTFIF	OCONTROL		FDCAN_STORE_TX_EVENTS
#define FDCAN1_MESSAGEMAR			0
<b>#define</b> FDCAN1_ID			FDCAN_STANDARD_ID
#define FDCAN1_FILTERINDE	X		0
#define FDCAN1_FILTERTYPE			FDCAN_FILTER_DUAL
#define FDCAN1_FILTERCONF			
FDCAN_FILTER_TO_RXE			
#define FDCAN1_FILTERID1			
FDCAN_DEAFULT_FILTE	RTD1		
#define FDCAN1_FILTERID2			
FDCAN DEAFULT FILTE	PTD2		
#define FDCAN1_RXBUFFERIN			1
/* CAN1 Receive Task Dela			Ŧ
			100
<pre>#define FDCAN1_RECEIVE_TA #ondif</pre>	-		100
<pre>#endif /* FDCAN1_ #endif //SDK_SEDVICE</pre>			
<pre>#endif //SDK_SERVICE</pre>	_FUCAN		

Dept: Software	User Manual Document	Rev No.	Rev 3.0
T		Date:	Oct 17, 2022

## 7.14 J1939

## 7.14.1 J1939 Module Configuration Support

The SDK provides the user the ability to enable/disable the J1939 functionality by modifying the default configuration file. Please see section 7.14.14 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	SDK_SERVICE_J 1939	PS_ENABLE PS_DISABLE	PS_DISABLE	PS_ENABLE: - Enables the J1939 module in the SDK PS_DISABLE: - Disables the J1939 module in the SDK

## 7.14.2 J1939 Module Timeout Configuration

The AI280 SDK user can configure the timeout value of the task such that, every time the timeout occurs the task would go and read the hardware and update it in the DB so that when the user reads the DB, he will receive the latest updated data if any or perform any other routine tasks. This default configuration can be done in the AI280\_config.h. Please see section 7.14.14 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_J1939_TASK_ TIMEOUT	MIN VALUE: 50 MAX VALUE: 500	100ms (Recommended Value)	The user can configure the timeout value of task so that the platform service would go and read the hardware and update the database.

## 7.14.3 J1939 task Priority

The AI280 SDK supports the below task priorities and the user can modify the task priority for the J1939 module in the configuration file. Please see section 7.14.14 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	PS_J1939_TASK_ PRIORITY	osPriorityNone, osPriorityIdle, osPriorityLow, osPriorityLow1, osPriorityISR, osPriorityError, osPriorityReserve d	osPriorityIdle (Recommended Value)	User can configure this macro to default priority

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

### 7.14.4 J1939 Claim Address Enable/Disable

The AI280 SDK supports the address claim functionality can be enabled or disable from the configuration file. Please see section 7.14.14 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	J1939_CLAIM_A DDRESS	PS_ENABLE/ PS_DISABLE	PS_ENABLE	User can enable/disable this macro to J1939 claim address

## 7.14.5 J1939 CAN Enable/Disable

The AI280 SDK supports enabling CAN1 Channel for J1939 and it can be enabled or disabled from the configuration file. Please see section 7.14.14 for sample configuration.

Sr. No	Variables	Options	Default State	Description	
1	J1939_CAN1_EN ABLE	PS_ENABLE/ PS_DISABLE	PS_ENABLE	User can enable/disable J1 CAN1	939

## 7.14.6 J1939 Claim Address

The AI280 SDK supports the address claim for channel CAN1 for J1939 and it can be configured as any value between 0-255 based on your requirement from the configuration file. Please see section 7.14.14 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	J1939_CAN1_CL AIM_ADDRESS	0-255	23	User can configure this macro to the required claim address

### 7.14.7 J1939 CAN Bit Rate

The AI280 SDK supports the bit rate configuration for the CAN1 for J1939 and it can be configured based on your requirement from the configuration file. Please see section 7.14.14 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	J1939_CAN1_BIT RATE	0/250/500/100	250	User can configure this macro to default J1939 CAN 1 BITRATE

## 7.14.8 J1939 Diagnostics Support

The AI280 SDK supports the enabling and disabling the diagnostics support for J1939 and they can be configured based on your requirement from the configuration file. Please see section 7.14.14 for sample configuration. The SDK Currently supports the below messages,

- 1) DM1
- 2) DM2

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Sr. No	Variables	Options	Default State	Description
1	EMTOS_J1939_D M1	PS_ENABLE PS_DISABLE	PS_ENABLE	User can enable/disable the Emotas J1939 DM1
2	EMTOS_J1939_D M2	PS_ENABLE PS_DISABLE	PS_ENABLE	User can enable/disable the Emotas J1939 DM2

### 7.14.9 J1939 Dynamic Address Claim

The AI280 SDK supports the dynamic address claim CAN1 for J1939 and it can be configured based on your requirement from the configuration file. Please see section 7.14.14 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	J1939_CAN1_AD DRESS_CLAIM_ DYNAMIC	PS_ENABLE PS_DISABLE	PS_ENABLE	User can enable/disable CAN1 dynamic address claim

## 7.14.10 J1939 Dynamic Address Claim Next Address Configuration

The AI280 SDK supports the dynamic address claim next address for CAN1 for J1939 and they can be configured based on your requirement from the configuration file. Please see section 7.14.14 for sample configuration.

When dynamic address claim is set to 1, search will start from this value and up to claim the address.

Sr. No	Variables	Options	Default State	Description
1	J1939_CAN1_AD DRESS_CLAIM_ NEXT_ADDRESS	0-255	80	User can configure the start address for address claim in dynamic mode for CAN1

### 7.14.11 J1939 Configure Number of PGN's supported

The AI280 SDK supports the configuration of the number of RX and TX PGNS on CAN1 for J1939 and they can be configured based on your requirement from the configuration file. Please see section 7.14.14 for sample configuration.

Please note that the SDK can support a maximum of 300 PGNs including RX and TX over CAN1 put together.

Sr. No	Variables	Options	Default Value	Description
1	CAN1_NUMBER_ PGNS_RX	MIN VALUE = 0 MAX VALUE = 300	18	User can configure the number of RX PGNS supported on CAN1

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

3	CAN1_NUMBER_ PGNS_TX	MIN VALUE = 0 MAX VALUE = 300	1	User can configure the number of TX PGNS supported on CAN1
---	-------------------------	-------------------------------------	---	--

#### 7.14.12 J1939 PGN and SPN Configuration

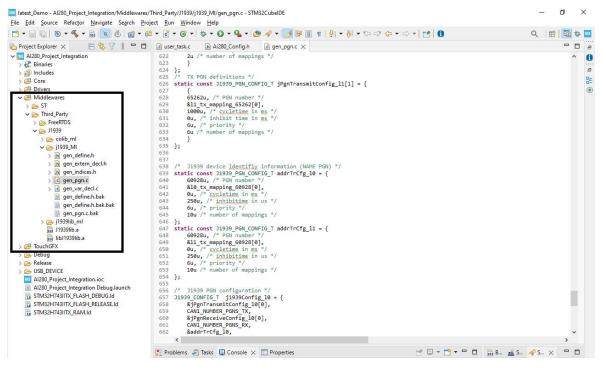
The AI280 SDK supports up to 300 PGN values and the same can be configured in the J1939 stack and their values will be read and updated in the Data layer for the application's access.

The user can replace existing PGNs' or configure new PGN and SPN values for CAN1 for J1939 during compile time and same can be accessed from the DB during runtime.

### 7.14.12.1 J1939 Source Code

The J1939 source and its configuration files are available in the below path as highlighted in the following diagram.

#### Middlewares/Third Party/J1939/\*



# 7.14.12.2 J1939 Supported PGN List

The AI280 SDK supports the below PGN values currently.

Sr. No	PGN	SIGNAL NAME
1	61444	Electronics Engine Controller 1
2	65110	Diesel Exhaust Fluid Tank 1 Information
3	65276	Dash Display
4	65272	Transmission Fluids 1
5	64774	Direct Lamp Control Command 2
7	65213	Fan Drive
8	65237	Alternator Information
9	65252	Shutdown
10	64892	Diesel Particulate filter control 1
11	65128	Vehicle Fluids VF
12	65237	Alternator information
13	65252	Shutdown
14	64998	Hydraulic Braking system
15	65089	Lighting command
16	65274	Brakes 1
17	64586	SCR System Cleaning
18	64523	Electronics Engine Controller 20
19	64525	Fire Pump Statistics 1
20	64529	Total Gaseous fuel information

## 7.14.12.3 J1939 Add PGN Configuration

The first step would be for the user to add the new PGN's in the configuration file. The PGN Configuration is available in the below structure which can be found in the file Middlewares/Third\_Party/J1939/j1939\_ML/gen\_pgn.c. This defines generic PGN structure for CAN1.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

```
/* J1939 PGN configuration */
J1939_CONFIG_T j1939Config_10 = {
    &jPgnTransmitConfig_10[0],
    CAN1_NUMBER_PGNS_TX,
    &jPgnReceiveConfig_10[0],
    CAN1_NUMBER_PGNS_RX,
    &addrTrCfg_10,
    J1939_CAN1_CLAIM_ADDRESS
```

};

The same file has the TX and RX structures where the user can include their own PGN's into the specific channel structure based on their requirement. Please note that for each PGN added, the minimum configuration that needs to be provided is,

- 1) PGN Number
- 2) SPN Mapping structure
- 3) Cycle time in milli seconds
- 4) Inhibit time in milli seconds
- 5) Priority
- 6) Number of mappings

The below code snippet shows sample PGN definitions.

```
/* TX PGN definitions */
static const J1939_PGN_CONFIG_T
      jPgnTransmitConfig_10[CAN1_NUMBER_PGNS_TX] =
 {
    {
      65262u, /* PGN number */
     &l0_tx_mapping_65262[0],
      10000u, /* cycletime in ms */
     Ou, /* inhibit time in ms */
      6u, /* priority */
      6u /* number of mappings */
    }
};
/* RX PGN definitions */
static const J1939_PGN_CONFIG_T
      jPgnReceiveConfig 10[CAN1 NUMBER PGNS RX]=
{
    {
      61444u, /* PGN number */
     &rx_mapping_61444[0],
      10u, /* cycletime in ms */
     Ou, /* inhibit time in ms */
      3u, /* priority */
```

```
9u /* number of mappings */
};
```

#### 7.14.12.4 J1939 Add SPN Configuration

For each PGN included by the user, the supported SPN structure will need to be included in the same file. The SPN definition structure will include the below information,

- 1) DB Buffer with index location
- 2) SPN Data Size
- 3) SPN Number
- 4) Data Type
- 5) If it's a dynamic PGN include the dynamic variable.

For example, we have the PGN 61444 included in the RX structure and hence the PGN's supported by 61444 are defined in the below code snippet.

/\* +++PGN 61444/0xf004 EEC1 Electronic Engine Controller 1 \*/

```
static const J1939_MAPPING_T rx_mapping_61444[] = {
```

{ &dl\_ju8[ENGINE\_TORQUE\_MODE - (START\_J1939U8BIT + 1) ], 4u, 899u, J1939\_DTYPE\_U4 } /\* Engine Torque Mode \*/,

{ &dl\_ju8[ACTUAL\_ENGINE\_PERCENT\_TORQUE\_FRACTIONAL - (START\_J1939U8BIT +
1)], 4u, 4154u, J1939\_DTYPE\_U4 } /\* Actual Engine - Percent Torque
(Fractional) \*/,

{ &dl\_ju8[DRIVERS\_DEMAND\_ENGINE\_PERCENT\_TORQUE - (START\_J1939U8BIT +
1)], 8u, 512u, J1939\_DTYPE\_U8 } /\* Drivers Demand Engine - Percent Torque
\*/,

```
{ &dl_ju8[ACTUAL_ENGINE_PERCENT_TORQUE - (START_J1939U8BIT + 1)], 8u,
513u, J1939_DTYPE_U8 } /* Actual Engine - Percent Torque */,
```

{ &temp\_speed, 16u, 190u, *J1939\_DTYPE\_U16* } /\* Engine Speed \*/,

```
{ &dl_ju8[SOURCE_ADDRESS_OF_DEVICE_ENGINE_CONTROL - (START_J1939U8BIT +
```

1)], 8u, 1483u, J1939\_DTYPE\_U8 } /\* Source Address of Controlling Device for Engine Control \*/,

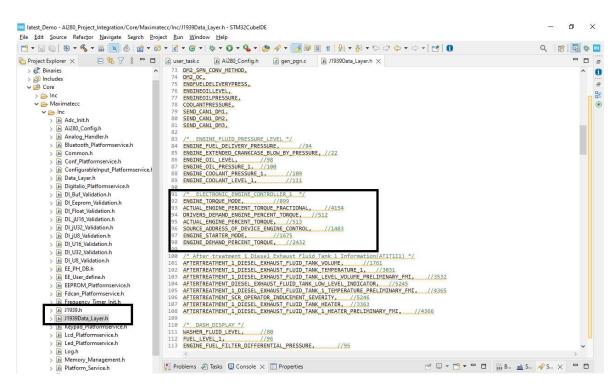
{ &dl\_ju8[ENGINE\_STARTER\_MODE - (START\_J1939U8BIT + 1)], 4u, 1675u, J1939\_DTYPE\_U4 } /\* Engine Starter Mode \*/,

{ &mv\_u8[0], 4u, 10001u, J1939\_DTYPE\_U4 },

{ &dl\_ju8[ENGINE\_DEMAND\_PERCENT\_TORQUE - (START\_J1939U8BIT + 1)], 8u, 2432u, J1939\_DTYPE\_U8 } /\* Engine Demand Percent Torque \*/ };

The #define for the SPN values can be added in the J1939Data\_Layer.h. The path for this file and the defines are highlighted in the below image,

Dept: Software	User Manual Document	Rev No.	Rev 3.0
T	Oser manual Document	Date:	Oct 17, 2022



## 7.14.12.5 Translate the SPN's before storing in DB

Once the user has successfully defined the PGN's and added them in the required structure the SDK will take care of capturing them. But before the SPN's are updated in the DB you will need to add the code specific to the PGN which translates them to a value that can directly be used by the application.

The below code snippet shows a sample translation done in the SDK to support the SPN (Engine speed which is part of the PGN 61444,

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

}

}

The above ENGINE SPEED PGN is the actual engine speed which is calculated over a minimum crankshaft and is at a resolution of 0.125 rpm per bit. It occupies 2 bytes of data and hence is stored in the dl\_ju16 array.

For any new PGN added by the user, a similar translation may be required based on the J1939 stack specification and PGN definition support.

## 7.14.12.6 Access the new SPN's from DB

The application user can directly access the DB variables of each SPN value that he has configured in the stack.

```
uint8_t val;
```

The below code snippet shows the example for the same.

```
/* Get ENGINE_SPEED */
Get_DL(ENGINE_SPEED , (uint16_t*)&val);
/* Get ENGINEOILLEVEL */
Get_DL(ENGINEOILLEVEL, (uint8_t*)&val);
/* Get ENGINEOILPRESSURE */
Get_DL(ENGINEOILPRESSURE, (uint8_t*)&val);
/* Get COOLANTPRESSURE */
Get_DL(COOLANTPRESSURE, (uint8_t*)&val);
```

## 7.14.13 J1939 Diagnostic Message Configuration

The AI280 SDK supports the J1939 diagnostic messages which help the user understand the current state of the device.

## 7.14.13.1 J1939 DM1 and DM2 Support in SDK

The AI280 SDK supports the configuration of the DM1, DM2 messages. These messages are already configured in the stack and the SDK DB has entries for each of these variables. Hence the user can directly access these SPN's from the application by accessing the below DB variables.

Field ID	Data Type	Permission	Size	Description	Comments
DM1_PROTECT _LAMP	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable
DM1_AMBER_ WARN_LAMP	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable
DM1_RED_STO P_LAMP	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable
DM1_MAL_FUN C_IND_LAMP	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

DM1_FLASH_P ROT_LAMP	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable
DM1_FLASH_ AMBER_WARN _LAMP	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable
DM1_FLASH_R ED_STOP_LAM P	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable
DM1_FLASH_M AL_FUNC_IND_ LAMP	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable
DM1_FMI	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable
DM1_SPN_CON V_METHOD	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable
DM1_OC	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable

Field ID	Data Type	Permission	Size	Description	Comments
DM2_PROTECT	DBu8	READ	1	Raw Data	Raw data value from J1939 is
LAMP				from J1939	updated into this DB Variable
DM2_AMBER_	DBu8	READ	1	Raw Data	Raw data value from J1939 is
WARN_LAMP	DDuo	KLAD	1	from J1939	updated into this DB Variable
DM2_RED_STO	DBu8	READ	1	Raw Data	Raw data value from J1939 is
P_LAMP	DBuð	KEAD	1	from J1939	updated into this DB Variable
DM2_MAL_FUN	DBu8		1	Raw Data	Raw data value from J1939 is
C_IND_LAMP	DBuð	READ	1	from J1939	updated into this DB Variable
DM2_FLASH_P	DD.,0		1	Raw Data	Raw data value from J1939 is
ROT_LAMP	DBu8	READ	1	from J1939	updated into this DB Variable
DM2_FLASH_				Raw Data	Raw data value from J1939 is
AMBER_WARN	DBu8	READ	1	from J1939	
_LAMP				110111 J 1939	updated into this DB Variable
DM2_FLASH_R				Raw Data	Raw data value from J1939 is
ED_STOP_LAM	DBu8	READ	1	from J1939	
Р				110111 J 1939	updated into this DB Variable
DM2_FLASH_M				Raw Data	Raw data value from J1939 is
AL_FUNC_IND_	DBu8	READ	1		
LAMP				from J1939	updated into this DB Variable

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

DM2_FMI	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable
DM2_SPN_CON V_METHOD	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable
DM2_OC	DBu8	READ	1	Raw Data from J1939	Raw data value from J1939 is updated into this DB Variable

The user will need to set the two variables from the Application to request the DM messages, shown in the below code snippet,

```
/* Set the CAN channel address and enable the DM1 messages */
    val = 0x84;
    Set_DL(CAN1_DEST_ADD, (uint8_t *)&val);
    val = 1;
    Set_DL(SEND_CAN1_DM1, (uint8_t *)&val);
/* Set the CAN channel address and enable the DM2 messages */
    val = 0x84;
    Set_DL(CAN1_DEST_ADD, (uint8_t *)&val);
    val = 1;
    Set_DL(SEND_CAN1_DM2, (uint8_t *)&val);
```

One the SDK receives the above messages, the DM packets will be process from the J1939 stack and then the values updated in the DB. The below code snapshot shows how the DM variables can be accessed from the touch GFX application.

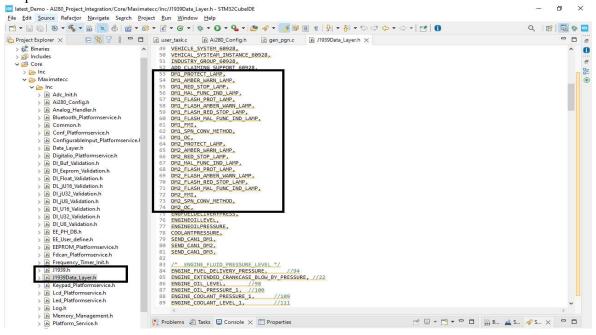
```
/* Get DM1 protect Lamp diagnostic data */
Get_DL(DM1_PROTECT_LAMP , (uint8_t*)&val);
```

### 7.14.13.2 J1939 Additional DM Support

The SDK also supports the user to add support for additional DM messages. To do the same the user will have to configure the stack.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

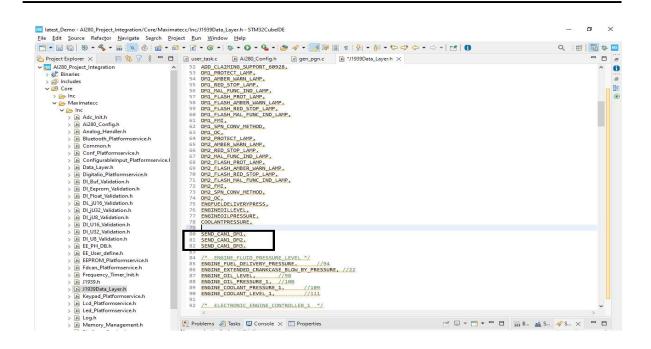
The user can add the SPN definitions in the J1939Data\_layer.h file as shown in the below snapshot.



For example, to add the support for DM2, the following #defines were added in this file,

DM2\_PROTECT\_LAMP, DM2\_AMBER\_WARN\_LAMP, DM2\_RED\_STOP\_LAMP, DM2\_MAL\_FUNC\_IND\_LAMP, DM2\_FLASH\_PROT\_LAMP, DM2\_FLASH\_AMBER\_WARN\_LAMP, DM2\_FLASH\_RED\_STOP\_LAMP, DM2\_FLASH\_MAL\_FUNC\_IND\_LAMP, DM2\_FLASH\_MAL\_FUNC\_IND\_LAMP, DM2\_FMI, DM2\_SPN\_CONV\_METHOD, DM2\_OC, The user will need to add the below variables so that the user can use these variables to enable the DM from the UI.

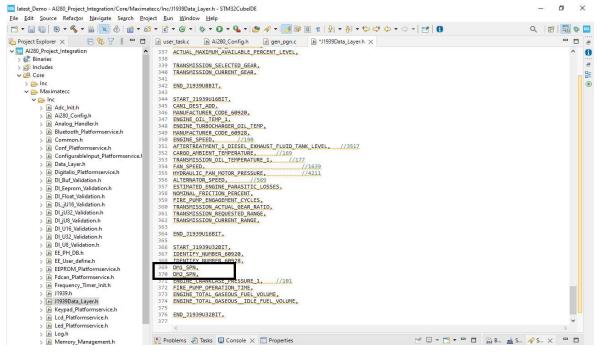
Dept: Software	User Manual Document	Rev No.	Rev 3.0
T		Date:	Oct 17, 2022



For example, to add the support for DM2, the following #defines were added in this file,

#### SEND\_CAN1\_DM2,

The next step would be to add a 32-bit entry as shown in the below image so that the access to the DM variables can be enabled.



Date: Oct 17, 2022	Dept: Software	User Manual Document	Rev No.	Rev 3.0
			Date:	Oct 17, 2022

The next step would be to enable it in the code. Please navigate to j1939.c as shown in the below image and add the details of the new DM message in the **diagnosticReceive** API as shown below. The code snippet shows the entries for DM1 and DM2 messages.

```
/*
 * registered function for diagnostic requests
 */
static void diagnosticReceive(
           UNSIGNED8
                      canLine,
                                        /* PGN requested */
           UNSIGNED32 pgn,
           UNSIGNED8
                                             /* requested node */
                      srcNode
     )
{
RET_T retVal;
//UNSIGNED32
                 spn;
#if (EMTOS_J1939_DM1 == PS_ENABLE)
     if (pgn == J1939_PGN_DM1) {
           printf("DM1 received\n");
           do {
                 retVal = j1939DiagnosticGet_DM1(canLine,
                      &dl ju8[DM1 PROTECT_LAMP - (START_J1939U8BIT + 1)],
                      &dl_ju8[DM1_AMBER_WARN_LAMP - (START_J1939U8BIT +
1)],
                      &dl ju8[DM1 RED STOP LAMP - (START J1939U8BIT + 1)],
                      &dl ju8[DM1 MAL FUNC IND LAMP - (START J1939U8BIT +
1)],
                      &dl ju8[DM1 FLASH PROT LAMP - (START J1939U8BIT +
1)],
                      &dl ju8[DM1 FLASH AMBER WARN LAMP - (START J1939U8BIT
+ 1)],
                      &dl ju8[DM1 FLASH RED STOP LAMP - (START J1939U8BIT +
1)],
                      &dl_ju8[DM1_FLASH_MAL_FUNC_IND_LAMP -
(START J1939U8BIT + 1)],
                      &dl_ju32[DM1_SPN - (START_J1939U32BIT + 1)],
                      &dl_ju8[DM1_FMI - (START_J1939U8BIT + 1)],
                      &dl_ju8[DM1_SPN_CONV_METHOD - (START_J1939U8BIT +
1)],
                      &dl_ju8[DM1_OC - (START_J1939U8BIT + 1)]);
           } while (retVal == RET_SERVICE_BUSY);
#if (EMTOS_J1939_DM2 == PS_ENABLE)
           //j1939RequestPgn(canLine, J1939 PGN DM2, 0x84);
#endif
```

Dept: Software	User Manual Document	Rev No.	Rev 3.0
•		Date:	Oct 17, 2022

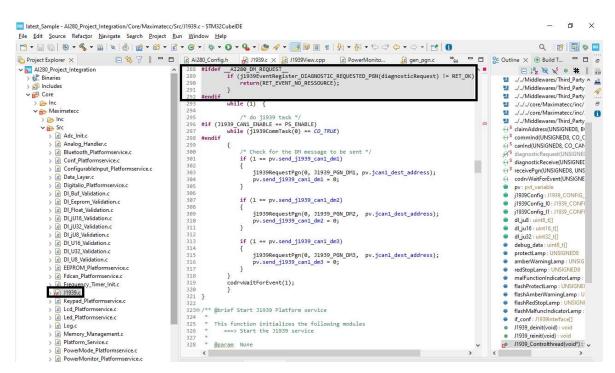
} #endif
<pre>#if (EMTOS_J1939_DM2 == PS_ENABLE)</pre>
if (pgn == J1939_PGN_DM2) {
<pre>printf("DM2 received\n");</pre>
do {
<pre>retVal = j1939DiagnosticGet_DM2(canLine,</pre>
&dl_ju8[DM2_PROTECT_LAMP - (START_J1939U8BIT + 1)],
&dl_ju8[DM2_AMBER_WARN_LAMP - (START_J1939U8BIT +
1)],
&dl_ju8[DM2_RED_STOP_LAMP - (START_J1939U8BIT + 1)],
&dl_ju8[DM2_MAL_FUNC_IND_LAMP - (START_J1939U8BIT +
1)],
&dl_ju8[DM2_FLASH_PROT_LAMP - (START_J1939U8BIT +
1)],
&dl_ju8[ <i>DM2_FLASH_AMBER_WARN_LAMP - (START_J1939U8BIT</i>
+ 1)],
&dl_ju8[ <i>DM2_FLASH_RED_STOP_LAMP - (START_J1939U8BIT</i> +
1)],
&dl_ju8[ <i>DM2_FLASH_MAL_FUNC_IND_LAMP</i> -
( <i>START_J1939U8BIT</i> + 1)],
&dl_ju32[ <i>DM2_SPN</i> - ( <i>START_J1939U32BIT</i> + 1)],
&dl_ju8[ <i>DM2_FMI</i> - ( <i>START_J1939U8BIT</i> + 1)],
&dl_ju8[ <i>DM2_SPN_CONV_METHOD</i> - ( <i>START_J1939U8BIT</i> +
1)],
&dl_ju8[ <i>DM2_OC</i> - ( <i>START_J1939U8BIT</i> + 1)]);
<pre>} while (retVal == RET_SERVICE_BUSY);</pre>
}
#endif
}

Once this is enabled, the SDK will capture the new DM packets and store them in the DB. The TouchGFX user can access them from the DB as described in the previous section.

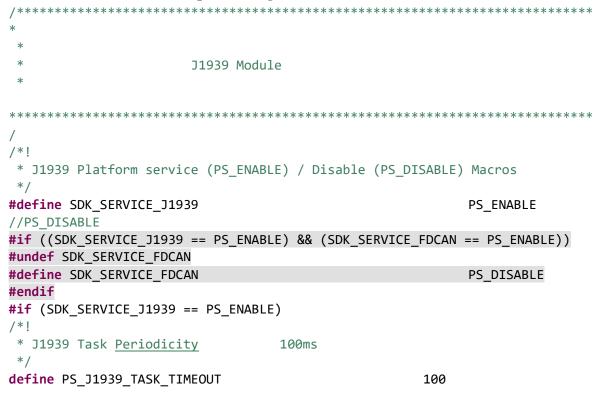
## 7.14.13.3 J1939 DM1 API Configuration

The below API support has been enabled in the J1939 file to be get the DM messages.

Dept: Softwa	re
--------------	----



# 7.14.14 J1939 Sample Configuration



Dept: Software	User Manual Document	Rev No.	Rev 3.0
1		Date:	Oct 17, 2022

/\*! \* J1939 Task Priority = 0, \* osPriorityNone \* osPriorityIdle = 1, \* osPriorityLow = 8, \* osPriorityLow1 = 8+1, ر ر ر ر رر ر ر \* osPriorityISR = 56, \* osPriorityError = -1, \* osPriorityReserved = 0x7FFFFFF \*/ #define PS\_J1939\_TASK\_PRIORITY osPriorityIdle /\*! \* J1939 CLAIM ADDRESS \*/ #define J1939\_CLAIM\_ADDRESS PS\_ENABLE /\* \* State BLE J1939 Debug data \*/ #define BLE J1939 DEBUG DATA PS ENABLE #define J1939\_CAN1\_ENABLE PS ENABLE /\*! \* J1939 CLAIM ADDRESS will be between 0 to 255 \*/ 23 #define J1939\_CAN1\_CLAIM\_ADDRESS /\*! \* J1939\_ADDRESS\_CLAIM\_DYNAMIC will search address when set to 1, Fixed = 0 \*/ #define J1939\_CAN1\_ADDRESS\_CLAIM\_DYNAMIC 1 /\*! \* J1939 ADDRESS CLAIM NEXT ADDRESS When dynamic set to 1, search will starting from this value and up \*/ #define J1939\_CAN1\_ADDRESS\_CLAIM\_NEXT\_ADDRESS 80 /\*! \* J1939\_CAN1\_BITRATE will be between 0/250/500/1000 \*/ #define J1939\_CAN1\_BITRATE 250 /\*! \* J1939 ENABLE/Disable DMx \*/ #define EMTOS J1939 DM1 PS ENABLE #define EMTOS J1939 DM2 PS ENABLE

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

<pre>//#define EMTOS_J1939_DM11 /*!  * J1939 Set number of PGN'S to receive</pre>	PS_EN and transmit	ABLE
<pre>*/ #define CAN1_NUMBER_PGNS_RX #define CAN1_NUMBER_PGNS_TX /*!</pre>	18u 1u	
* J1939 Set COB handlers */		
<pre>#define CAN1_CO_COB_COUNTS #endif //SDK_SERVICE_J1939</pre>	36u	

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 7.15 Throughput Module

The User would be able to use the below functionalities of the Throughput module via the DB variables and configuration file.

## 7.15.1 Throughput Enable/Disable

The SDK provides the user the ability to enable/disable the throughput functionality by modifying the default configuration file. Please see section 7.15.4 for sample configuration.

Sr. No	Variables	Options	Default State	Description
1	THROUGH_PUT- SERVICE	PS_ENABLE/ PS_DISABLE	PS_ENABLE	<ul> <li>PS_ENABLE:</li> <li>Enables the Throughput module in the SDK</li> <li>PS_DISABLE:</li> <li>Disables the Throughput module in the SDK</li> </ul>

# 7.15.2 Throughput Absolute Time

During Run time, the below DB variables are used for displaying the absolute time of each software module.

Field ID	Data Type	Permission	Size	Options	Description
KEYPAD_A BSTIME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
DIO_ABSTI ME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
CI_ABSTI ME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
POWERMO DE_ABSTI ME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
LED_ABST IME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time

Dept: Software	Software User Manual Document	Rev No.	Rev 3.0
*		Date:	Oct 17, 2022

POWER_M ONITOR_A BSTIME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
DEFAULT_ ABSTIME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
BLUETOO TH_ABSTI ME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
RTC_ABST IME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
SW_TIMER _ABSTIME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
EEPROM_ ABSTIME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
WATCHDO G_ABSTIM E	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
LCD_ABST IME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
CAN_ABST IME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time based on their application
J1939_ABS TIME	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time
TOUCHGF X_ABSTIM E	DBu32	READ	1	0- FFFFFFFF	The total time that the task has actually been executing is the Absolute time

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 7.15.3 Throughput Percentage Time

During Run time, the below DB variables are used for displaying the Percentage time of each software module.

Field ID	Data Type	Permission	Size	Options	Description
KEYPAD_P ERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
DIO_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
CI_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
POWERMO DE_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
LED_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
POWER_M ONITOR_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
DEFAULT_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
BLUETOO TH_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
RTC_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

SW_TIMER _ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
EEPROM_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
WATCHDO G_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
LCD_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time Processing Time
CAN_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
J1939_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time
TOUCHGF X_ PERTIME	DBu8	READ	1	0-100	Provides the Percentage of the total Processing Time rather than an absolute time

# 7.15.4 Throughput Sample Configuration

# **8** Application Details

The MAXAI280 SDK S/W release package includes a sample application per module which demonstrates the functionalities of the modules and can be used for reference by the user to understand how to interact with the SDK.

It also includes a graphical demo application which can be used as a reference to understand how to use multiple modules in a single application and tie them to various graphical UI elements that are available in the touch GFX screen.

# 8.1 Sample Application Project Details

This sample application per module gives you a walk-through of the test procedure for each module available. The sample application can be used as a basis to understand what functionalities are available in each module and how the user can interact with the individual modules.

This section includes a brief description about sample application user interface and the minute details of each module, which includes module description, module screen navigation and the test procedures.

## 8.1.1 Introduction

To open the sample application project please follow the same procedure followed to open the blank project file as described in Section 3.

Once you have successfully compiled and flashed the Sample project on the AI280 hardware, you can reboot the hardware to run the application.

Initially after the device is turned ON, A Home Screen will be displayed on the LCD screen which contains the list of all the available modules. The Home Screen sample image is given below:



Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date: Oct 17, 2022	Oct 17, 2022

## 8.1.2 Home Screen Navigation:

In the Home Screen you can find the list of all the modules available in the AI280 SDK i.e., Keypad, Power Monitor, RTC, EEPROM, LCD, Digital o\p, SW Timer, Config input, LED, BLE, Power Mode, J1939, Flash, through put, WD and CAN.

Along with the modules there are four key navigators (i.e., Previous, Next, Enter, Back) which will allow the user to move front, back, up and down, enter into a specific module and exit from the specific module. The four key navigators are operated using the four built-in buttons, which are located to both sides of the screen (left and right) and are represented as Key1, Key2, Key3, and Key4 respectively. Each button has a specific functionality which is mentioned in below table.

Button Name	Functionality
Key1	Go to previous
Key2	Go to next
Key3	Enter/Select
Key4	Back

## 8.1.3 Keypad Module

The keypad module sample application is shown in the below screen,



Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date: Oct 17, 2022	Oct 17, 2022

## 8.1.3.1 Module Description:

This Keypad module is basically designed to test the functioning of all the four keys present on the device. The testing can be done for each key (i.e., All the four keys) to check whether it's working properly based on their specific functionality.

## 8.1.3.2 Module Navigation:

To go to the Keypad module, In the Home Screen navigate to the keypad module using Key 1 and Key 2 and then select the keypad using Key3 i.e., Enter, which will take you to Keypad Functional screen.

On this Screen you will find four Keypad options i.e., Keypad1, Keypad2, Keypad3 and Keypad4. Each Keypad consists of two Blocks below them, where one block is used for short press test update and the other Block is used for long press test update.

Note: Short Press denotes a single click on the key.

Long Press denotes click and hold the key.

Test Case	Keypad Action	Description	Test Procedure	Expected Result
KeyPad 1	Short Press	This key is used to turn ON the Backlight which are present on the keys	Click on key1 and check if the light on all the keys turns ON.	Light on the button Glows
KeyPad 1	Long Press	This key is used to turn OFF the Backlight which are present on the keys	Click and Hold Key1. Then check if the light on all the keys turns OFF.	Light on the button Turns OFF
KeyPad 1	Continu ous Press	This key is used to turn OFF the Backlight which are present on the keys	Click and Hold Key1. Then check if the light on all the keys turns OFF.	Light on the button Turns OFF
KeyPad 2	Short Press	This is specifically used to check whether the key is functioning properly.	Click on Key2 and then check if the short press block below key2 gets highlighted	Keypad2ShortpressblockpresentonLCDwillbeHighlighted
KeyPad 2	Long Press	This is specifically used to check whether the key is functioning properly.	Click and Hold on Key2.Then check if long press block below key2 gets highlighted	Long press block present on the LCD will be Highlighted

## 8.1.3.3 Module Test Procedure:

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

KeyPadContinuThis is specifically used to check whether the key is functioning properly.Click and Hold on key2.Then check if long press block below key2 gets highlightedContinuous press block present or the LCD will b HighlightedKeyPadShortThis is specifically used to check whether the key is functioning properly.Click on Key3 and then check if the short press block below key3 gets highlightedShort press block the LCD will b HighlightedKeyPadLong functioning properly.This is specifically used to check whether the key is functioning properly.Click and Hold on block below key3 gets key3.Then check if long press block below key3 gets highlightedCong press block present on th LCD will b HighlightedKeyPadLong functioning properly.This is specifically used to functioning properly.Click and Hold on long press block below key3 gets highlightedLong press block HighlightedKeyPadContinu functioning properly.This is specifically used to functioning properly.Click and Hold on long press block below key3 gets highlightedKeyD will b HighlightedKeyPadShort functioning properly.This is specifically used to functioning properly.Click on Key4 and then short pressShort press block block present or long press block below key3 gets highlightedKeyPadShort functioning properly.This is specifically used to check whether the key is functioning properly.Click on Key4 and then short press block block below key4 gets highlightedShort press block functioning properly. <t< th=""><th>2 KeyPad</th></t<>	2 KeyPad
Pressfunctioning properly.long press block below key2 gets highlightedthe LCD will b HighlightedKeyPad 3Short 	KeyPad
KeyPadShortThis is specifically used to check whether the key is functioning properly.Click on Key3 and then check if the short pressHighlightedKeyPadLongThis is specifically used to functioning properly.Click and Hold on long press block below key3 gets highlightedLCD will be HighlightedKeyPadLongThis is specifically used to check whether the key is functioning properly.Click and Hold on long press block below key3 gets highlightedLong press block present on the long press block below key3 gets highlightedKeyPadContinuThis is specifically used to check whether the key is functioning properly.Click and Hold on long press block below key3 gets highlightedContinuous press block present on the LCD will be key3 gets highlightedKeyPadShortThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short pressShort press block block present on the LCD will be key3 gets highlightedKeyPadShortThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short pressShort press block present on the block below key4 gets highlighted4PressThis is specifically used to check whether the key is functioning properly.Click on Key4 and then block below key4 gets highlighted	•
KeyPad 3Short PressThis is specifically used to check whether the key is functioning properly.Click on Key3 and then check if the short pressShort press bloc present on th LCD will b HighlightedKeyPad 3Long PressThis is specifically used to check whether the key is functioning properly.Click and Hold on Long press blockLong press bloc present on th LCD will b HighlightedKeyPad 3Long PressThis is specifically used to check whether the key is functioning properly.Click and Hold on Long press block below key3 gets highlightedLong press bloc present on th LCD will b HighlightedKeyPad 3Continu ous PressThis is specifically used to check whether the key is functioning properly.Click and Hold on LCD will b key3 gets highlightedContinuous press block present of tong press block below key3 gets highlightedKeyPad 4Short PressThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short press block below key4 gets block below key4 gets highlightedShort press bloc Highlighted	•
3Presscheck whether the key is functioning properly.check if the short press block below key3 gets highlightedpresent on the LCD will b Highlighted3PressThis is specifically used to check whether the key is functioning properly.Click and Hold on key3.Then check if long press block below key3 gets highlightedLong press bloc present on the LCD will b HighlightedKeyPad 3Continu ous PressThis is specifically used to check whether the key is functioning properly.Click and Hold on key3 gets highlightedLong press block present on the long press block below key3 gets highlightedKeyPad 4PressThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short press block below key4 gets block below key4 gets highlightedShort the LCD will b Highlighted	•
KeyPad 3Long PressThis is specifically used to check whether the key is functioning properly.Click and Hold on long press block below key3 gets highlightedLCD will b HighlightedKeyPad 3Continu ous PressThis is specifically used to check whether the key is functioning properly.Click and Hold on long press block below key3 gets highlightedLong press block HighlightedKeyPad 3Continu ous PressThis is specifically used to functioning properly.Click and Hold on long press block below key3 gets highlightedContinuous press block present on the LCD will b key3 gets highlightedKeyPad 4Short PressThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short press block below key4 gets block below key4 gets highlightedShort present on the LCD will b Highlighted	3
KeyPad 3Long PressThis is specifically used to check whether the key is functioning properly.Click and Hold on long press block below key3 gets highlightedLong press block present on th long press block below key3 gets highlightedKeyPad 3Continu ous PressThis is specifically used to check whether the key is functioning properly.Click and Hold on long press block below key3 gets highlightedLCD will b HighlightedKeyPad 3Continu ous PressThis is specifically used to functioning properly.Click and Hold on long press block below key3 gets highlightedContinuous press block present of long press block below key3 gets highlightedKeyPad 4Short PressThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short press block below key4 gets block below key4 gets highlightedHighlighted	
KeyPad 3Long PressThis is specifically used to check whether the key is functioning properly.Click and Hold on Key3.Then check if long press block below key3 gets highlightedLong press bloc present on th LCD will b HighlightedKeyPad 3Continu ousThis is specifically used to check whether the key is functioning properly.Click and Hold on long press block below key3 gets highlightedContinuous press block present or the LCD will b highlightedKeyPad 4Short PressThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short press block below key4 gets block below key4 gets highlightedShort press block the LCD will b Highlighted	
3Presscheck whether the key is functioning properly.Key3.Then check if long press block below key3 gets highlightedpresent on th LCD will b HighlightedKeyPad 3Continu ous pressThis is specifically used to functioning properly.Click and Hold on long press block below key3 gets highlightedContinuous press block present of the LCD will b HighlightedKeyPad 4Short PressThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check whether the key is block below key4 gets block below key4 getsShort press block highlighted	
3Presscheck whether the key is functioning properly.Key3.Then check if long press block below key3 gets highlightedpresent on th LCD will b HighlightedKeyPad 3Continu ous pressThis is specifically used to functioning properly.Click and Hold on long press block below key3 gets highlightedContinuous press block present of the LCD will b HighlightedKeyPad 4Short PressThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check whether the key is block below key4 gets block below key4 gets highlightedShort press block press	KeyPad
KeyPadContinu ous Pressfunctioning properly.long press block below key3 gets highlightedLCD will b Highlighted3ous PressCheck whether the key is functioning properly.Click and Hold on long press block below key3 gets highlightedContinuous press block present o long press block below key3 gets highlightedKeyPad 4Short PressThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short press block below key4 gets block below key4 gets highlightedShort press block Highlighted	•
KeyPadContinuThis is specifically used to check whether the key is functioning properly.Click and Hold on Key3.Then check if long press block below key3 gets highlightedHighlightedKeyPadShortThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short pressShort press block pressKeyPadShortThis is specifically used to check whether the key is functioning properly.Click on Key4 and then block below key4 getsShort press block pressHighlightedHighlightedClick on Key4 and then block below key4 getsShort press block present on the block below key4 getsClick on Key4 gets block below key4 gets	
KeyPad 3Continu ous PressThis is specifically used to check whether the key is functioning properly.Click and Hold on Key3.Then check if long press block below key3 gets highlightedContinuous press block present of the LCD will b HighlightedKeyPad 4Short PressThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short pressShort press block pressKeyPad 4Short PressThis is specifically used to functioning properly.Click on Key4 and then block below key4 gets block below key4 getsClick on the LCD will b Highlighted	
3ous Presscheck whether the key is functioning properly.Key3.Then check if long press block below key3 gets highlightedblock present of the LCD will b HighlightedKeyPad 4Short PressThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short pressShort press block present on th block below key4 getsLCD will b highlightedLCD will b Highlighted	KeyPad
Pressfunctioning properly.long press block below key3 gets highlightedthe LCD will b HighlightedKeyPadShortThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short pressShort press block present on th LCD will b highlighted	•
KeyPadShortThis is specifically used to check whether the key is functioning properly.Click on Key4 and then check if the short pressShort press bloc present on th block below key4 getsLCD will b highlighted	
4 Press check whether the key is functioning properly. check if the short press present on the block below key4 gets highlighted below key4 ge	
functioning properly. block below key4 gets highlighted Highlighted	KeyPad
highlighted Highlighted	4
<b>KevPad</b> Long This key is used to Exit from Click and Hold on Back\Exit	
	KeyPad
4 Press the Keypad Functionality Key3. Then check if the	4
screen and go to Home Screen Screen gets exited from	
Keypad functionality	
and goes to home	
screen.	
KeyPad Continu This key is used to Exit from Click and Hold on Back\Exit	
4 ous the Keypad Functionality Key3.Then check if the	KeyPad
Press screen and go to Home Screen Screen gets exited from	•
Keypad functionality	•
and goes to home	•
screen.	•

#### 8.1.4 Power Monitor

The power monitor module sample application is shown in the below screen,



## 8.1.4.1 Module Description:

This Power Monitor test is basically used to provide the temperature, battery level and ignition status updates.

## **8.1.4.2** Module Navigation:

To go to Power Monitor Test screen, from Sample Application Screen navigate to Power Monitor block using Key1 and Key2, then by using Key3 enter into the Power Monitor Test Screen

In Power Monitor Test screen there are 3 blocks available they are:

- 1) IGNITION STATE. (IGN\_STATE)
- 2) TEMPERATURE LEVEL. (TEMP\_LEVEL)
- 3) BATTERY LEVEL. (BAT\_LEVEL)

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Soltware		Date: Oct 17, 202	Oct 17, 2022

8.	1.4.3 Module Test P	rocedure:		
Test Case	Description	Test Procedure	Expected Result	Units
IGNITION STATE	This functionality is used for testing whether the external device supply is ON or OFF	to provide some external	supply provided is turned On then it will give	_
TEMPERAT URE LEVEL	This functionality provides the update about the temperature level present in the device area.	navigatetoTEMP_LEVELthenclick on Key3 (Enter).	is located will be	Celsius
BATTERY LEVEL.	This functionality updates the battery level of the device.	Using Keyland Key2 navigate to BAT_LEVEL then click on Key3 (Enter). Now check the result that is displayed at the blank space.	of the device will be	mv

## 8.1.5 RTC

The RTC module sample application is shown in the below screen,



# 8.1.5.1 Module Description:

RTC is the real time clock which supports the RTC configuration for the AI280 board and also supports two alarms.

# 8.1.5.2 Module Navigation:

To go to RTC, from "Sample Application" navigate to RTC using key1 and Key2 then click on key3 to enter into RTC screen.

RTC has the three blocks as given below:

- 1) RTC Time
- 2) ALARM A
- 3) ALARM\_B

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Software			Oct 17, 2022

#### 8.1.5.3 Sub Screens:

The three blocks contain Sub Screens which are used to set the time or alarm. These sub screens are,

**RTC screen**: The RTC screen consist of HR: MIN:SEC: WKDAYS:DD:MM: YY blocks which is used to set the Hour's, Minutes, Seconds, Week Days, Date, Month, Year respectively using the Key1 and Key4 Short Press. And to display it on the main Screen of RTC long press Key4.



**ALARM\_A screen:** The ALARM\_A screen is routed by selecting ALARM\_A from RTC main screen. This screen consists of HR: MIN:SEC: WK\_DAY, which are again for Hour's, Minutes, Seconds, Week Days respectively and can be set using the Key1 and key4 Short Press. And there is an empty block available below which Displays ALARM\_OCCURED message when ALARM\_A is triggered in the platform.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
Dept. Software		Date:	Oct 17, 2022



**ALARM\_B screen:** The ALARM\_B screen is routed by selecting ALARM\_B from RTC main screen. This screen consists of HR: MIN:SEC: WK\_DAY, which are again for Hour's, Minutes, Seconds, Week Days respectively and can be set using the Key1 and key4 Short Press. And there is an empty block available below which Displays ALARM\_OCCURED message when ALARM B is triggered in the platform.



ð.	1.5.4 Module Test	Procedure:	
Test Case	Description	Test Procedure	Expected Result
RTC Time	This functionality is used to set the time of the device	Using keyl navigate to RTC Time block and then enter using key3.Now in the Sub Screen, by using Keyl set all the required details then by a short press on key4 you can set the time and later give a long press exit from the present	The updated time will be displayed on the screen.
ALARM_A	This functionality is used to set alarm on the device	screen and go to RCT main screen. Using keyl navigate to Alarm A block and then enter using key3.Now in the Sub Screen, by using Keyl set all the required details and then by a short press on Key4 set the alarm.	When the alarm gets triggered in the platform the empty Block under ALARM_A will display ALARM_OCCUR ED message
ALARM_B	This functionality is used to set alarm on the device	Using keyl navigate to Alarm B block and then enter using key3.Now in the Sub Screen, by using Keyl set all the required details and then by a short press on Key4 set the alarm.	When the alarm gets triggered in the platform then the empty Block under ALARM_B will display ALARM_OCCUR ED message.

# 8.1.5.4 Module Test Procedure:

# 8.1.6 LCD

The LCD module sample application is shown in the below screen,



# 8.1.6.1 Module Description:

LCD functionality is basically designed to check if the screen is functioning properly. The verification is done based on two factors one is by turning ON and OFF the screen and the other by increasing or decreasing the brightness of the screen.

# 8.1.6.2 Module Navigation:

To go to LCD test, from "Sample Application" screen select LCD using Key1 and Key2 and later enter into the LCD Test Screen by using Key3. In the LCD Test there is a session which is used to update the brightness value of the display.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 8.1.6.3 Module Test Procedure:

Test Case	Keypad Action	Description	Test Procedure	Expected Result
Brightness	Increase	This functionality is	Click on	The brightness
Value		used to increase the	key1(Brightness+) and	value would be
		brightness of the	check the results	increase from the
		Device		previous value
Brightness	Decrease	This functionality is	Click on	The brightness
Value		used to decrease the	key2(Brightness-) and	value would be
		brightness of the	check the results	decrease from the
		Device		previous value
LCD	-	This functionality is	Click on the ON\OFF	The LCD display
<b>ON\OFF</b>		used to Turn ON/OFF	button given on the	will turn ON in
		the LCD Screen	LCD Screen.	case of ON and
				Turn Off in case of
				OFF

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 8.1.7 Digital Output

The digital output module sample application is shown in the below screen,



# 8.1.7.1 Module Description:

There are two digital pins in the device digital high and digital low. This Digital Output module is used to set this pin values.

# 8.1.7.2 Module Navigation:

To go to the Digital Output screen, from "Sample Application" Screen select Digital Output using Key1 and Key2, then enter the Digital Output screen using key3.On the Digital output screen there are 4 blocks they are as given below:

High Side Low Side Status ON Status OFF **Note:** There are three configurable modes available in Digital output they

Mode	High pin	Low pin
High side	1	0
Low side	0	1
Open Drive	0	0

# 8.1.7.3 Module Test Procedure:

Test Case	Description	<b>Test Procedure</b>	Expected Result

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Status OFF	This functionality is used to set the system into Open Drive configuration mode	Using Key1 and Key2 navigate to Status OFF block and click on Key3. Now check the results	The system will go to OPEN Drive state which means the high pin and low ping will be low.
Status ON	This functionality is used to set the Digital output pin state to the previous state which was present before the system was set as OFF status	Using Key1 and Key2 navigate to Status OFF block and click on Key3. Now check the results	The pin status gets updated according to the previous state.
High side	This is used to show what pin state the system is operation on at present.	This will be tested along with status ON\OFF	High Side block in the display will get highlighted and the high pin will be 1 and low pin will be 0.
Low side	This is used to show what pin state the system is operation on at present	This will be tested along with status ON\OFF	Low Side block in the display will get highlighted and low pin will be 1 and high pin will be 0.
Open Drive	This is used to set the system into Open Drive	This will be tested along with status ON\OFF	The system will go to OPEN Drive state which means the high pin and low ping will be low.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

## 8.1.8 Software Timer

The software timer module sample application is shown in the below screen,



# 8.1.8.1 Module Description:

The software timer is mainly designed to interact with the timer module. This will be consisting of timer reset, timer trigger and also to read the current timer counter value. There are a total of six timers available from Software Timer1 to Software timer6.

## 8.1.8.2 Module Navigation:

To go to Software Timer, from "Sample Application" screen navigate to Software Timer Using Key1 and Key2, later click on Key3 to enter to Software Timer Test screen. In this screen there will be six blocks available which are used for six different timers.

## 8.1.8.3 Sub Screen:

**Sub Screen\_1:** From Software test screen when a timer is selected it will be redirected to Sub Screen\_1 which has a name of the related Timer number. In this screen there are two blocks available they are:

- 1) Single Shot
- 2) Inactive

Dept: Software	<b>User Manual Document</b>	Rev No.	Rev 3.0
		Date:	Oct 17, 2022



# 8.1.8.4 Module Test Procedure:

Test Case	Keypad Action	Description	Test Procedure	Expected Result
Single/ Inactive	Short press	This functionality is used to stop the timer	User can navigate to the timer using key1(NEXT) or key2(PREVIOUS) and can select the any one timer by pressing key3(ENTER), User can go back to previous screen using key 4(BACK).	Timer gets stopped

Dept: Software	<b>User Manual Document</b>	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

		Once the user navigates to the SW Timer 1, and press the key3 (ENTER), SWTIMER_1 GUI screen appears, where user can set the timer according to the requirement.	
Continuous/ Active	Long Press	 •	

## 8.1.9 Configurable Inputs

The below screen shot shows the configurable input screen.



# 8.1.9.1 Module Description:

The Configurable input are basically designed to configure the various input channels and then read the latest Configurable Input values according to the initial configuration done. The channels which are available for configuring are voltage, resistance, digital STB, digital STG, resistance and frequency. The MAXAI280 supports 5 configurable inputs.

## 8.1.9.2 Module Navigation:

To go to configurable Input, from "Sample Application" screen navigate to Config Input block using Key1 and Key2, now using Key3 enter to Configurable Input Test Screen.

In Configurable Inputs test screen, there are five blocks available which has five AI values from AII Value to AI5 Value. This block represents different input channels they are as given below:

- 1) Voltage
- 2) Digital STB
- 3) Digital STG
- 4) Resistance
- 5) Frequency

Note: From AI1 to AI5 the configurable inputs can be assigned to below mentioned channel:

1) Voltage\_32v

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

- 2) Voltage\_6v
- 3) Frequency
- 4) Resistance
- 5) Digital STG
- 6) Digital STB

# **8.1.9.3 Module Test Procedure:**

	8.1.9.5 Widdule Test Trocedure.				
Test Case	Description	Test Procedure	Expected Result		
AI1 to AI5	AI1 to AI5 are values which are assigned to the configurable input channels 1 to 5 that are available for configurable inputs functionality	Using Key1(Long press) navigate to the specific block from AI1 to AI5 then by using Key1(Short press select the desired channel) and then check UI above the channel for the updated value based on the configuration. If you navigate to AI1 you will be able to configure it as resistance, voltage low and high, frequency, digital input. For example, if you configure it as resistance, you will see the input resistance value on the screen.	•		

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

## 8.1.10 LED

## 8.1.10.1 Module Description:

The LED functionality is designed to test the LED light which is In-Built in the device. The LED testing can be done in three modes they are:

- 1) Turn On
- 2) Turn OFF
- 3) Blink

The MAXAI280 supports 2 LEDs. They are,

- 1) RED
- 2) AMBER

#### **8.1.10.2 Module Navigation:**

To go to LED screen, from "Sample Application" screen navigate to LED using Key1 and Key2.Later enter into the LED test screen using Key3. In this screen there are two functionalities available, they are:

- 1) RED LED
- 2) AMB LED

## 8.1.10.3 Sub Screen:

The two functionalities which are available in LED test screen, are again having individual sub screen, they are explained in detail below:

**Sub Screen\_1:** When the user selects from RED LED and AMB LED, Sub Screen\_1 will be opened which again has two functionalities, they are:

- 1) ON/OFF
- 2) BLINK

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022



**Sub Screen\_2:** When the user selects the ON and OFF functionality then Sub Screen\_2 will be displayed on this screen it updates if the LED is ON/OFF.



Dept: Software	are User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

**Sub Screen\_3:** When the user selects the Blink functionality then Sub Screen\_3 will be displayed on this screen it updates if it's turned On and at what speed will the LED blink.



**8.1.10.4 Module Test Procedure:** 

Test Case	Mode	Description	Test Procedure	Expected Result
RED LED	ON/OFF	This functionality is used to Turn ON the RED LED present on the Bottom right Corner of the device	From the main Screen (LED test Screen) select RED LED Using Key1 and Key2, user will be navigated to Sub Screen_1 there select ON\OFF using Key1 and Key2, Now the user will be redirected to Sub Screen_2, now using Key1 turn On /OFF the LED based on requirement.	LED light Glow's. And if the user selects OFF mode, then the LED light
RED LED	BLINK	This functionality is used to blink the LED.	From the main Screen (LED test Screen) select RED LED Using Key1 and Key2, user will be navigated to Sub Screen_1 there select Blink using Key1 and Key2, Now the user will be redirected to Sub Screen_2, now	If the User selects the Blink On mode, then the LED starts blinking at the specified rate.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

		I		
			using key2 and Key3 increase or decrease the blink speed and	
			Using Key1 set Blink ON\OFF	
AMB LED	ON/OFF	This functionality is used to Turn ON the AMB LED present on the Bottom right Corner of the device	From the main Screen (LED test Screen) select AMB LED Using Key1 and Key2, user will be navigated to Sub Screen_1 there select ON\OFF using Key1 and Key2, Now the user will be redirected to Sub Screen_2, now using Key1 turn ON /OFF the LED based on requirement.	If the User Selects ON mode, then the LED light Glow's. And if the user selects OFF mode, then the LED light which was glowing will turn OFF
AMB LED	BLINK	This functionality is used to blink the LED.	From the main Screen (LED test Screen) select AMB LED Using Key1 and Key2, user will be navigated to Sub Screen_1 there select Blink using Key1 and Key2, Now the user will be redirected to Sub Screen_2, now using key2 and Key3 increase or decrease the blink speed and Using Key1 set Blink ON\OFF	If the User selects the Blink On mode, then the LED starts blinking at the specified rate.

Dept: Software User Manual Documer	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

## 8.1.11 Power Mode



# 8.1.11.1 Module Description:

The Power Mode Module is basically designed to check the low power functional mode of the device. There are three different functionality modes available for Power Mode Module which indicates the present state of the device. Currently the AI280 devices supports only the Stop mode and the user can enter/exit the stop mode functionality based on their requirement.

To exit the stop mode functionality the user can configure one of the below inputs,

- 1) RTC
- 2) Keypad
- 3) Ignition

#### **8.1.11.2 Module Navigation:**

To go to the Power Mode Test, from "Sample Application" screen select Power Mode Test by using Key1 and Key2, after that enter into Power Mode Test Screen by using Key3. In the Power Mode Test user can test the Stop mode functionality. This would enable the user to enter the low power mode.

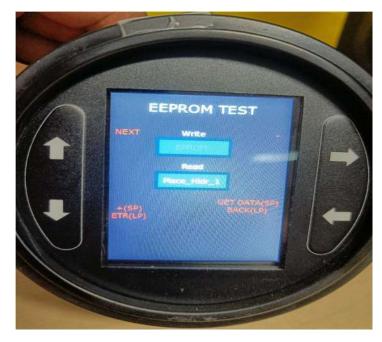
0.	8.1.11.3 Module Test Procedure:			
Test Case	Description	Test Procedure	Expected Result	
Stop Mode	This functionality is used to stop all the functions inside the system and it is waiting into the same mode until an interrupt will occur and activate the device.	Using Key1 and Key2 navigate us to Stop Mode; now by pressing Key3 will set the device into Stop mode.	The running functionality will be kept on halt, and the system will be set to Stop mode i.e., system will turn OFF	
Keypad	This is a wake-up source which is used to activate the system from the power mode	Click on any one of the keys then the system will turn on Note: Keypad 4 is not working	The device will be activated back	
RTC	This is a wake-up source which is used to activate the system from the power mode	The user needs to set a timer (After how long the system has to restart) in the configuration file and wait till the timer reached the desired time gap.	The device will be activated back	
Ignition	This is a wake-up source which is used to activate the system from the power mode	When there is any activity performed in the power monitor module then the system will be activated	The device will be activated back	
CAN	This is a wake-up source which is used to activate the system from the power mode	The CAN must send the signals to the device in case the user wants to activate the system using CAN	The device will be activated back	

# 8.1.11.3 Module Test Procedure:

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

## 8.1.12 EEPROM

The EEPROM module sample application is shown in the below screen,



# 8.1.12.1 Module Description:

The functionality is basically designed to store the data in EEPROM memory and read the same when required. The user will provide the input value to the place holder and output will be stored into the EEPROM memory. User can provide values up to 65535 place holders.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022



## **8.1.12.2 Module Navigation:**

To go to the EEPROM Test, from "Sample Application" screen select EEPROM Test by using Key1 and Key2, after that enter into the EEPROM Test Screen by using Key3. In the EEPROM Test user can interact with the EEPROM module for below functionality.

- 1) To write and store the data in EEPROM memory.
- 2) To read the stored data from EEPROM memory.

	Description	Test Procedure	Expected Result
Test Case			
Write	This functionality is	Using Key1 and Key2	The input data provided at
Mode	used to write and store	navigates to Write Mode; now	the place holder will be
	the data into the	by pressing Key3 will set the	stored at the EEPROM
	EEPROM memory.	device to the write mode.	memory.
Read	This functionality is	Using Key1 and Key2	User can read the data
Mode	used to read the stored	navigates to Read Mode; now	stored at the EEPROM
	data from the EEPROM	by pressing Key3 will set the	memory.
	memory.	device to the read mode.	

## 8.1.12.3 Module Test Procedure:

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

## 8.1.13 Watchdog

The watchdog module sample application is shown in the below screen,



# 8.1.13.1 Module Description:

The functionality is basically designed to monitor the state of the device. Watchdog reset depends on the Pre-scaler. The Pre-scaler value will be provided within the range of 4 to 256.

Once the user provides the pre-scaler value 256, the system will reset after 50 seconds. User has to go to the config.h file to enable or disable any property on the Board.

# 8.1.13.2 Module Navigation:

To go to the Watchdog Test, from "Sample Application" screen select Watchdog Test by using Key1 and Key2, after that enter into the Watchdog Test Screen by using Key3. The user can select any watchdog (WD0 – WD9). In the Watchdog Test user can interact with the Watchdog module for below functionality.

- 1) To enable/disable the watchdog functionality.
- 2) To feed the watchdog manually.

8.1.13.3 Module Test Procedure:					
Test Case	Description	Test Procedure	Expected Result		
Watchdog	This functionality is used to enable a property on the system.	navigates to Watchdog	Watchdog, whatever the default state is available		
Watchdog Ping	used to feed or refresh	Once the user selects any watchdog from WD0 to WD9 and hit stop ping, it will stop the feeding.	Hardware will go into reset mode after a few seconds.		

Dept: Software User Manu	al Document Rev No.	<b>Rev 3.0</b>
	Date:	Oct 17, 2022

#### 8.1.14 BLE

### 8.1.14.1 Module Description:

The functionality is basically designed to establish a BLE connection between two devices and send/receive data. To communicate using Bluetooth connection between the two devices, users can pass the BLE terminal commands and get the acknowledgement on the same screen.



#### 8.1.14.2 Module Navigation:

To go to the BLE Test, from "Sample Application" screen select BLE Test by using Key1 and Key2, after that enter into the BLE Test Screen by using Key3. In the BLE Test, user can interact with the BLE module for below functionality:

- 1) Scan
- 2) Connect
- 3) Read
- 4) Write

#### 8.1.14.3 Steps to Scan and Connect:

The User can use the BLE Application through mobile phone using below mentioned steps.

**Step 1:** Once the user opens the BLE application, they need to click on START SCANNING. Here the user will find MAX device name advertised as "maxAI12345678" as shown below.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Fa	stBLE		
	START SCANNING		
<b>ሙ</b> "	49:D0:2A:92:A7:D6	-13 ·M	NEC
≱»	74:C4:52:82:C7:98	-42 M	
₿»	CB:57:54:A5:1B:E7	-96 M	
<b>₿</b> "	22:2A:6D:AD:14:51	-82 M	
<b>₿</b> »	61:25:18:D7:E9:70	-95 M	CON NEC
₿"	maxAl12345678 00:A0:50:00:88:8F	-89 M	
₿"	0E:A8:46:FE:22:11	-96 M	
<b>₿</b> "	3A:12:E3:ED:49:BE	-97 M	
<b>₿</b> "	FB:5F:03:F3:21:2C	-96 M	
<b>₿</b> "	4E:2E:E0:6E:02:81	-84 M	
₿"	TVS00131371 00:13:43:79:0D:10	-94 M	
<b>₿</b> "	Amazfit Bip Watch D3:CB:58:02:E0:0C	-93 M	
<b>∦</b> "	73:9E:D4:A4:56:AD	-93 M	

Users can connect these devices by clicking on the CONNECT option. Once it is done the user will navigate to the SERVICE screen.

# FastBLE

		START SCANNING			
<b>ች</b> "	49:D0:2A:92:A7:D0	5	-\3	NEC	
\$"	74:C4:52:82:C7:98	1 I	-42 .M		
₿"	CB:57:54:A5:1B:E7	7	-96 M		
<b>∦</b> "	22:2A:6D:AD:14:5	r.	-82 M		
₿"	61:25:18:D7:E9:70		-95 M		
₿"	0E:A8:46:FE:22:11		-96 M		
₿"	3A:12:E3:ED:49:BE		-97 M		
₿»	FB:5F:03:F3:21:20		-96 M		
₿"	4E:2E:E0:6E:02:81		-84 .M		
₿»	TVS00131371 00:13:43:79:0D:10		-94 M		
₿»	Amazfit Bip Wat D3:CB:58:02:E0:00		-93 M		
₿"	73:9E:D4:A4:56:AI	5	-93 M		
÷	maxAI1234567 00:A0:50:00:B8:8F	Connecte	d DISC ONN	ENT ER	

**Step 2:** In SERVICE Screen the user will find Service (0). Once the user will select the Service (0), it will move to the characteristic screen as shown below:

Service list

 Device broadcast name: maxAl12345678

 MAC:00:A0:50:00:B8:8F

 Service(0)

 00001800-0000-1000-8000-00805f9b34fb

 Service type (main service)

 Service(1)

 00001801-0000-1000-8000-00805f9b34fb

 Service type (main service)

 Service(2)

 da70c284-1041-4f12-8c99-6cbd8039d48a

 Service type (main service)

**Step 3:** In the characteristic screen user will find characteristic (0). Once the user selects the characteristic (0), the user will move to BLE terminal screen as shown below.

← Characteristic list	
Characteristic (0)	
c2246f54-9539-4222-97f6-1c14a654a33c Characteristic( READ )	>
Characteristic (1)	
c2246f54-9539-4222-97f6-1c14a654a33c	>
Characteristic( READ )	

**Step 4:** In the BLE Terminal Screen, user will be able to write or read data from the MAXAI280 device using the BLE.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

← Cor	nsole	
8060f101815	3	WRITE
	READ	
	ט טט טט טט טט טט טט טט טט טט טט טט טט ט	1000010000
00 00 80 60 f1 03 c1 0 00 00 00 00 00 0	0 12 a7 00 00 00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00

# 8.1.14.4 Steps to Read and Write:

**Step 1:** The user will write the message in the mobile application in the form of Hexadecimal and that message will be sent to the GUI

WRITE

Step 2: That message will be received to the GUI screen which will be as show below

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022



**Step 3:** If the user wants to send the message from the GUI, then, from the GUI user can select any of the available default messages using Key2 and later by clicking on Key3 user can set that message.



Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

**Step 4:** Once the message is sent from the GUI the User has to click on Read button on the mobile application and the message will be displayed on the screen in Hexadecimal format.

← Console	
Please enter the HEX format command	WRITE
READ	
80 61 51 42 48 00 65 00 62 00 62 00 66 00 20 00 77 00 66 0 00 00 00 00 00 00 00 00 00 00 00 00 0	

# **8.1.14.5 Module Test Procedure:**

Test Case	Description	Test Procedure	Expected Result
Scan	This Functionality is used to detect the device and display the device name on the mobile application	User needs to open the FastBLE application in the phone, click on the START SCANNING option. User needs to check for the MAX device named "maxAI12345678". Once it is detected, then we need to click on "CONNECT" option. To establish a connection between two devices, the user needs to send the start request packet along with header packet attached.	packet is received from the user, system will send the positive response packet back to the user to give the confirmation message as the
Connect	This functionality is used to establish a connection between the device and the mobile application	Once the device is detected through the mobile application, user need to connect the device by	be established between the mobile
		clicking on "Connect" which	device

Dei	nt:	Software
	<b>y</b>	Solumate

		is present on the mobile application	
Write	This functionality is designed such that when the user sends a hexadecimal data from the mobile application on that, data will be received to the GUI in string format.	User needs to enter the hexadecimal value at the write column which is present in the mobile application and check the result in the GUI	GUI which will be the converted form
Read	This functionality is used to send message from the GUI and will be received by the mobile application.	User can select the default messages (There are four default message) which has to be sent to the mobile application using key2() and set it by pressing key4(OK) Now in the mobile application, click on "Read" command and check the results.	The message sent from the GUI will be converted to hexadecimal and

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

### 8.1.15 CAN

#### 8.1.15.1 Module Description:

The CAN is a control area network which is basically used to control the ECU (Electronic control unit). Can acts as a master controller and based on the functionality it sends the request and receives the messages.

#### **8.1.15.2 Module Navigation:**

To go to CAN, from "Sample Application" screen navigate to CAN using Key1 and Key2, and enter into CAN test Screen using Key3. In CAN test screen there are two blocks present they are:

#### $CAN_1$



#### 8.1.15.3 Sub Screen

**Sub Screen\_1**: When the user selects from CAN\_1 in CAN test screen user will be redirected to Specified functionality test screen which will be out Sub Screen\_1. In this Sub Screen\_1 there are two packets available they are:

Send Pckt

Rx Pckt

#### Steps to send and Receive packets:

For CAN1:

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

Step 1: The required message will be sent in the form of a packet and the packet will be sent to the device via CAN Terminal.

Appearan		Mo																
	Event Type		Receive N	S	D	P	Dir	D	Data	-							J1939 Interp	re
noine1 Engine1	CAN Frame CAN Frame CAN Frame CAN Frame	Engine1 Engine1 Engine1		00 00 00 FE	E FF FF	6 6 6 6	Tx Tx Tx Tx	8 8 8	00 F C0 8	6 00 F FA 7 16	26 10	FF 00 CO	CA 00 87	FE 00 16	00 00 10		[TP] BAM	Р.
ngine1 I_Engine1	CAN Frame CAN Frame 11939 Frame CAN Frame	Engine1 Engine1 Engine1 Engine1	Instrument	00 00 00 00	17 FF FF	3666	Tx Tx Tx Tx Tx	8 8 22 8	CO 0 03 0 54 5 00 0	0 00	00	00	00	00 FF	00 FF	FF	 [TP] Sequenc	H 3

Step 2: When the packet will be received by the device the packet message will be in hexadecimal format.



Dept: Software	User Manual Document	Rev No.	Rev 3.0
Depersonenare		Date:	Oct 17, 2022

Step 3: User needs to select any of the message which is available by default using Key2 and Key3 and set it as the sending packet by clicking on Key4.



Step 4: The message which was sent from the device will be received at the CAN terminal as shown below.

Appearance		MC																		
1 1	Event Type		Receive N	Sale	D	P	Dir		De									9	1939 Inte	rpn
Engine1 ox	CAN Frame CAN Frame	Engine1 Engine1 Engine1		00	作作作	6 6 6	T× T× T×	B B B	20	16 FF	15 1 00 0 FA 2	04 F	FF (	CA 00	FE 00	00		E	TP] BAM	
	CAN Frame CAN Frame CAN Frame	Engine1 Engine1	Instrument	00	LI 17 FF	03 6	Tx Tx Tx	8 8 8	04	00	10 J FA C FF F	D C	00 I	FO	FF	00 FF		E	TP] Seque	nce:
	01030 Frame CAN Frame CAN Frame	Engine1 Engine1	Contraction of the local distance of the loc	00	FF	0 6	Tx Tx Rx	22 8	00	00	6F 0	00 2	20 0	00	00	EO	晤			
																918				

0.1.1	5.4 Module Test Procee				
Test Case	Description	Test Procedure	Expected Result		
CAN1	This Functionality is used to check the packet, which was sent from the externally connected CAN analyser.	In the CAN test screen go to CAN1 using key1 and Key2 then enter the CAN1 Testing screen using Key3. And check the result	The message packet which was sent from the CAN analyser will be will be received at the Rx Packet area in string format		
CAN State	This functionality is used to read the state of the CAN	When the user clicks on Keyl for once then the CAN state will be read.	The was just written as an example for user to understand the usage of can state DB variable.		
Filter Index/CAN mode	This functionality is used to update the Filter details and the mode in which is operation on.	When The user double clicks on Key1 then the CAN state will be updated.	The user can check if the CAN filter is enabled and the filter index is 20 and the standard mode is set in the DB for can channel 1.		
CAN Baud rate	This functionality is used to set the device Baud rate	When The user clicks on Key1 for Tree times, then The CAN Baud rate function will be updated.	The user can check if the CAN 1 baud rate is updated to 250K.		
CAN Drive Reset	This functionality is used to reinitialize the applicant.	When The user clicks on Keyl for four times, then The CAN Drive Reset function will be implemented.	The user can check this update on the DB variable		
CAN Reset	This functionality is used to set the device into power down mode	When The user clicks on Keyl for five times, then The CAN Reset function will be implemented.	The user can check this update on the DB variable		

# **8.1.15.4 Module Test Procedure:**

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 8.1.16 J1939

#### 8.1.16.1 Module Description:

J1939 module is used to interface with the J1939 stack and receive the PGN functionality values and update the values to the GUI. J1939 is also used to provide Diagnosis message to the user.

#### **8.1.16.2 Module Navigation:**

To go to JI939 functionality, from "Sample Application" screen navigate to J1939 using Key1 and Key2, later enter into J1939 screen using Key3.In J1939 test screen there are different PGN present they are:

- 1) EngFuelDeliveryPress
- 2) EngineOilLevel
- 3) EngineOilPressure
- 4) Coolant Pressure

There are three Diagnosis message available they are:

- 1) DM1
- 2) DM2



Dept: Software	User Manual Document	Rev No.	Rev 3.0		
		Date:	Oct 17, 2022		

**Sub Screen\_1**: When the user selects DM1 using Key1 in J1939 test Screen and enters to Sub Screen\_1 by clicking Key3 then a screen consisting for eight diagnosis messages will be displayed which are listed as follows:

**Sub Screen\_2**: When the user selects DM2 using Key1 in J1939 test Screen and enters to Sub Screen\_1 by clicking Key3 then a screen consisting for four diagnosis messages will be displayed which are listed as follows:



Note: To go back to main screen (J1939 Test screen) User need to click on Key4.

Test Case	Description	Test Procedure	Expected Result
DM1	Ŭ	User can set to DM1 using Key2 and Key3 to enable the SDK to capture diagnostic message to the database through CAN which was updated to CAN via J1939	The SDK will start capturing the diagnostic message that's and update those on to the screen.

#### 8.1.16.3 Module Test Procedure:

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

DM2	DM as in Diagnosis message are provided for the user to send messages to the database through CAN	User can set to DM2 using Key2 and Key3 to enable the SDK to capture diagnostic message to the database through CAN which was updated to CAN via J1939	The SDK will start capturing the diagnostic message that's and update those on to the screen.
EngFuelDeliver yPress	This functionality is used to provided data about the Engine Fuel	For testing purpose, the device can be connected to the external CAN analyser to receive this PNG values	The Fuel delivery pressure value will be displayed on the specified space in the UI.
EngineOilLevel	This functionality is used to provided data about the engine Oil level	For testing purpose, the device can be connected to the external CAN analyser to receive this PNG values	. The Oil level value will be displayed on the specified space in the UI.
EngineOilPress ure	This functionality is used to provided data about the Oil Pressure	For testing purpose, the device can be connected to the external CAN analyser to receive this PNG values	The Oil pressure value will be displayed on the specified space in the UI.
Coolant Pressure	This functionality is used to provided data about the coolant pressure	For testing purpose, the device can be connected to the external CAN analyser to receive this PNG values	The Coolant pressure value will be displayed on the specified space in the UI.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

### 8.1.17 Throughput

#### **8.1.17.1 Module Description:**

The main functionality of through put is to constantly update the absolute time and percentage time used by each module until the device is working

#### **8.1.17.2 Module Navigation:**

To go to Throughput, from "Sample Application" screen navigate to Throughput using Key1 and Key2. Later enter the throughput test screen using Key3. In this throughput test screen, all the modules are listed for which there are two functionalities which are being updated, they are:

1) Absolute Time 2) Percentage Time



8.1.17.3 Module Test Procedure:

Test Case	Description	Test Procedure	Expected Result
Absolute Time	the total 'time' that the task has been executing (the total time that the task has	This functionality is designed to constantly update the Absolute Time used by the individual module in the UI without any intervention from the user.	be constantly

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

	to select a suitable time base for their application.		
Percentage Time	provide essentially the same information but as a percentage of the total	This functionality is designed to constantly update the Absolute Time used by the individual module in the UI without any intervention from the user.	-

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 8.2 Details of Demo Application

The Demo application is a combined TouchGFX application which will provide insight into how we can combine the different services of the SDK and write a wholesome application.

#### 8.2.1 Difference between Sample Application and Demo Application

The Sample Application was written to help the AI280 SDK User to understand the functionalities of the individual modules and use them as per their requirement. The home screen helps navigate to all the available modules present on the "Sample Application" screen, which can be tested by entering into a specific module whereas in case of Demo Application there are five screens available which has all the modules integrated within the screens based on their functionality. And the screens can be switched using the panel button functionality mentioned below in 7.2.2.

### 8.2.2 Panel Button Functionality

Initially when the device is turned ON, the main interface is displayed which would be the screen 1, now to shift from one screen to another screen and to interact with each screen the below keys are available,

Block Name	Function	Key press instructions	Description
Key1	Back	Short Press	This key is used to go back to the previous screen.
Key2	Inc++	Short Press	This key is used to increment the value of a specific module
Key2	SET	Long Press	This key is used to update the changes
Key3	Dec	Short Press	This key is used to decrement the value of a specific module
Key3	SEL NEXT	Long Press	This key is used to select the next module
Key4	Next SCR	Short Press	This key is used to go to the next Screen

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

#### 8.2.3 Demo App Screen 1

The below image shows the integrated UI screen from which the user can verify the following software modules.

- 1) J1939
- 2) RTC

#### 8.2.3.1 Screen 1 Description:

Screen-1 contains Gauge and RTC. The gauge has a pointer value that will vary based on engine speed and engine speed will be updated based on CAN and J1939.

The RTC has a digital Clock to display the data.



In the above image all the sections of the screen are shown using the arrows, as each section is functioning for different test case such as RTC is for time, J1939 for Engine Speed.

Module Name	Screen Sections	Description	Test Procedure	Results	Range
RTC	Time	clock value which will be displayed on	There is no specific test, rather the time will be updated based on the real time data	will be displayed on the Digital	-

#### 8.2.3.2 Screen 1 Test procedure:

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

J1939	Engine	This functionality is	When the system is	The gauge value -
	Speed	used to update the	connected to an	varies based on
		engine Speed based	external device via	the speed of the
		on J1939 which gets	CAN, the CAN	system.
		updated through	channel will send the	
		CAN	speed details to	
			J1939 and that value	
			will be updated on	
			the UI without any	
			intervention from the	
			user.	

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 8.2.4 Demo App Screen 2

The below image shows the integrated UI screen from which the user can verify the mentioned software modules.

#### 8.2.4.1 Screen 2 Descriptions:

Screen-2 consists of two modules which are mentioned below:

#### (1) Digital Output

(2) Ignition Indication

The below illustration provides details of screen 2. The screen has two partitions the first partition contains Digital Output. The other partition contains the Ignition. The digital output comprises of two pin values which is HIGH side and LOW side and one open drive, these pins decide the value of the output. The value 1 indicates the HIGH side and value 0 indicates the LOW side and open drive indicates open. The power monitor comprises the ignition indicator which indicates whether the ignition is ON/OFF. When the indicator is turned ON the power monitor is enabled and when it is turned OFF the power monitor is disabled. When both the high side and Low side pins are OFF (value is 0), the state of the Digital Output is Open drive.



In the above screen, the user can see Output block which has two functionality one is for High Side and the other is for Low Side, in this both we have ON and OFF options. Next is the LCD brightness which has a slider that is used to show the brightness level. Next to LDC brightness is Ignition which has OFF/ON option. On to the other side of the Screen there is Input block available which has four configuration properties and their respective counter values. The user can verify the below functionality on Screen 2 of the sample app.

	8.2.4.2 Scre	en 2 Test Procedure	•
Module Name		Description	Test Procedure
		The digital output comprises of two	U

	Name	Sessions	Description	Test Procedure	Result	Units
c	Digital Output	Output	The digital output comprises of two pin values which is HIGH side, LOW side and Open drive these pins decide the value of the output. The value 1 indicates the HIGH side and value 0 indicates the LOW side. If both the value of HIGH side and LOW side are 0, the digital output is in Open drive	can probe the digital output pins by verifying whether the Digital Output high pin should be 1 and Digital Output low Pin should be 0. User can navigate to	The High Side functionality present in the Output block will be Turned ON	-
				Low Side: - User can probe the digital output pin to by verifying the Digital Output High pin should be 0 and Digital Output low Pin should be 1. User can navigate to Low side by long pressing Key2 and set it using Key3(long press)	present in	-
5				<b>Open Drive: -</b> User can probe the digital output pin to by verifying the Digital Output High pin should be 0 and Digital Output low	The Open drive functionality can be observed by GUI (both high side	

#### 8717 S on ) Tost Dr 4

N

Expected

Units

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

			Pin should be 0.	and low side
			User can navigate to	GUI should
			Open Drive (High	be OFF) and
			side and low side	user can
			OFF) by long	verify by
			pressing Key2 and	probing the
			set it using	digital io
			Key3(long press)	pins state.
Power	Ignition	The power monitor	User can navigate	The Ignition -
Monitor	Indication	comprises the	the Ignition	will switch
		indicator mode	ON/OFF.	modes to
		which provides two		ON/OFF
		basic functionalities		based on
		ON/OFF mode.		Power
		When the indicator		Monitor.
		is turned ON the		
		power monitor is		
		enabled and when it		
		is turned OFF the		
		power monitor is		
		disabled.		

#### 8.2.5 Demo App screen 3

Screen-3 consists of only configurable input module.



The configurable input is designed to configure various input channels. It is used to update the latest configurable input values based upon the previously configured channel properties. The different configurable inputs that are currently available are Digital, resistance, voltage and frequency. The blocks in the figure represents different configurable inputs.

Module Name	Screen Sessions	Description	Test Procedure	Expected Result	Units
Configurable Input	Input	The configurable input is designed to configure various input channels. It is used to update the latest configurable input values based upon the previously configured channel properties. The different configurable inputs that are currently	<ul> <li>each property for a particular channel and then view the data in the adjacent location.</li> <li>Digital: - User can configure the</li> <li>Digital property for AI1 to AI5 channel Using Key2(Short</li> </ul>	the digital input will be updated on	0-5

# 8.2.5.1 Screen 3 Test Procedure:

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

available are Digital, resistance, current and frequency.	Key3(Short press) and set the specific AI value using Key3		
	Resistance: - User can configure the resistance property for AI1 to AI5 channel Using Key2(Short press) and Key3(Short press) and set the specific AI value using Key3	The value of the Resistance input will be updated on the GUI counter.	Ohm's
	Frequency: - User can configure the frequency property for AI1 to AI5 channel Using Key2(Short press) and Key3(Short press) and set the specific AI value using Key3	The counter value of the frequency input will be updated	Hz
	Voltage: - User can configure the Voltage property for AI1 to AI5 channel Using Key2(Short press) and Key3(Short press) and set the specific AI value using Key3	The value of the Voltage input will be updated on the GUI counter.	V

#### 8.2.6 Demo App Screen 4

The below display is the integrated UI screen from which the user can select the following software modules.

- 1) SW Timer
- 2) EEPROM
- 3) Power mode

#### 8.2.6.1 Screen 4 Descriptions:

Screen 4 is designed using three software module which includes Software timer, EEPROM and Power Mode.

The topmost block which is available on the screen is for Software Timer which has three functionalities:

- (1) Increase Timeout
- (2) Decrease Timeout
- (3) Current timeout

In the later part there are two sections, EEPROM Power mode is available. In the EEPROM block the Timeout counter is read from the EEPROM and displayed and for Power mode the Stop functionality is available.



Dept: Software	User Manual Document	Rev No.	Rev 3.0
, i i i i i i i i i i i i i i i i i i i		Date:	Oct 17, 2022

# 8.2.6.2 Screen 4 Test procedure:

Module	Screen	Functionality	Description	Test Procedure	Expected
Name	Section	_			Result
Software	SW	Increase	The software	User can increase	Time out
Timer	Timer	Timeout	timer is used to set timeout value for the device software. Where increase timeout is used to increase the time out value	the timeout value by short pressing of the key-2(Short Press).	value will be increased and displayed under Current timeout
Software	SW	Decrease	The software	User can decrease	Time out
Timer	Timer	Timeout	timer is used to set timeout value for the device software. Where decrease timeout is used to decrease the timeout value	the timeout value by short pressing the key-3.	value will be decreased and displayed under Current timeout
Software Timer	SW Timer	Current Timeout	The software timer is used to set timeout value for the device software. where current timeout is used to show the current timeout value	The current value is auto generated based on the increment or decrement operation performed by the user.	The current timeout value will be updated

EEPROM	EEPROM	Timeout Counter	EEPROM stores the software timer data and the same can be read back.	The timeout counter value will be directly fetched form the EEPROM database which will be dependent on the Software timeout.	The Timeout counter value will be displayed.
Power Mode	Power mode	Stop Mode	This functionality is used to stop all the functions inside the system and it is waiting into the same mode until an interrupt will occur and activate the device.	User can set the power mode as STOP by long pressing of the key-3 the board will switch to the reset mode. User can come back to the wake- up state by using any one of the wake-up sources. They are: • Keypad • RTC • Ignition • CAN	functionality will be kept on halt, and the system

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 9 BLE Mobile Test Application

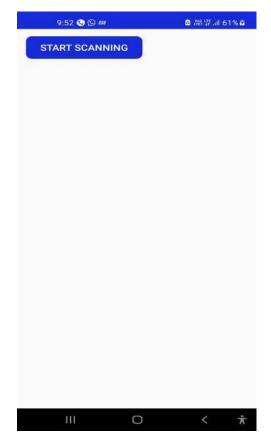
The BLE test application is an android based mobile application which can be used to communicate with the MAXAI280 hardware via BLE for testing/debugging purpose. It supports the below functionalities.

Read/write to all the DB Variables supported by the SDK Direct Memory location Read.

In this section we will walk you through the BLE App screens and how to use the functionalities of the BLE App.

# 9.1 Scan Screen

Install the .apk file into your Android Mobile. After installation, open the App. The app home screen is shown as below.



Press the Start Scanning button. The list of Maximatecc AI280 devices available will be displayed on the screen as shown below.

Dept: Software	User Manual Document	Rev No.Rev 3.0	Rev 3.0
		Date:	Oct 17, 2022

If bluetooth is turned OFF on your mobile phone you will get a notification to turn on Bluetooth and location before using the BLE functionality.

Once the scanning starts you will see the below screen with the list of devices.



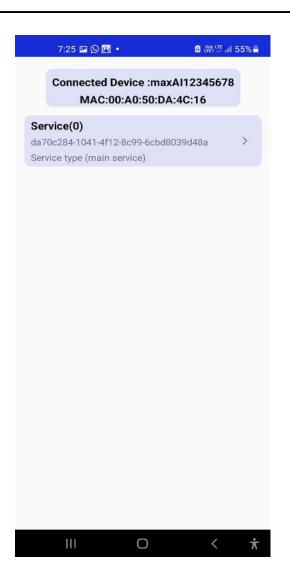
Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date: Oct 17, 202	Oct 17, 2022

# 9.2 Connect Screen

Press the "**Connect**" button" to connect to the MAXAI280 device. Once the device is connected, "**Connected**" button status will be shown. Press the enter button to go to the next screen. To disconnect the device from BLE communication, the user can press the "**Disconnect**" button.

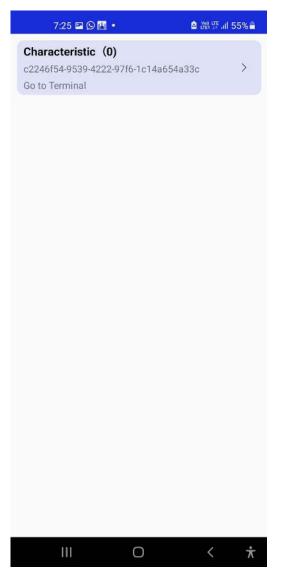


After the connection is successful, the connected device names and services will be shown on the next screen. Press the service UUID on the list and it will move to the next screen.



Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

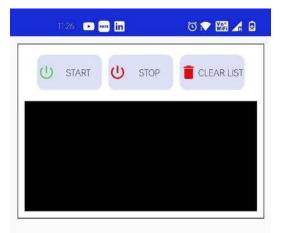
The screen below shows the characteristic UUID on the list. Press the characteristics on the list and it will move to the GUI Screen to communicate with the device.



Dept: Software	User Manual Document	Rev No.	Rev 3.0
· · · · · · · · · · · · · · · · · · ·		Date:	Oct 17, 2022

# 9.3 GUI Screen

The main GUI Screen for the device is shown below.



#### Figure 2

Press the START button to start with the testing. The second half of the screen will be populated with the options once the test has started and the start button is highlighted as shown below.

12:04 🖂 😅 🥥 🗹 👘 🕉 🖼 🛣 🛣 🖉
U START U STOP 🕇 CLEAR LIST
$ \begin{array}{l} R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  f1  01  6e  00  47  87  00  00 \\ R_{\rm XC}  80  60  10  16  60  04  78  10  00 \\ R_{\rm XC}  80  60  10  16  80  46  10  16  80  47  87  00  00 \\ R_{\rm XC}  80  60  10  16  80  46  10  16  80  47  87  00  00 \\ R_{\rm XC}  80  60  10  16  80  47  87  10  00 \\ R_{\rm XC}  80  60  10  16  80  47  87  10  00  00 \\ R_{\rm XC}  80  60  10  16  80  47  87  10  00  00 \\ R_{\rm XC}  80  60  10  16  80  47  87  10  00  00 \\ R_{\rm XC}  80  60  10  16  80  47  87  10  00  00 \\ R_{\rm XC}  80  10  16  80  16  16  80  10  16  80  10  16  80  10  16  80  10  16  80  10  16  80  10  16  80  10  16$
Data Layer ID 47
Buffer 01 READ WRITE
Data By Memory Address
Address 1011 Size 03
READ
Generic Data to Send

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date: Oct 17, 2022	Oct 17, 2022

# 9.4 Read/Write DB Variable Screen

The screen for Read/Write DB Variable is as below. The screen marked by RED shows the Terminal for RX/TX communications of the DB variables. The screen marked by BLUE shows the Read/Write DB Variable Screen.

12:05 🕗 📾 🧐 🛩 🛛 🕤 📳 🕷 📽 📽 🕯	
U START U STOP 🛑 CLEAR LIST	
Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00         Rx: 80 60 f1 03 c1 00 12 o7 00 00	
Data By Common Identifier	
DB Variables LCD_STATE -	
Data Layer ID 47	
Buffer 01 READ WRITE	
Data By Memory Address	
Address 1011 Size 03	
READ	

To read the data from the AI280 SDK DB, select the appropriate SDK module for accessing the module's DB variables on the drop-down list. Select the DB variables in the DB drop-down list.

Please click the READ Button for the selected DB variable. The terminal will reflect the communication between the device and the Mobile App. The values present on the device shall also be reflected on the Terminal. To send the updated data to the device, manually type the value

Dept: Software	Dept: Software User Manual Document	Rev No.	Rev 3.0
Depti Solemare		Date:	Oct 17, 2022

in the buffer text box and then click the WRITE Button. The Terminal will reflect the communication between the device and the Mobile App.

12:05 🖂 💷 🥱 🗾 🔞 🎼 🕷	¢⁴97 ₽
U START U STOP 🕇 CLEAR	RLIST
$ \begin{array}{l} Rx: 80\ 60\ f1\ 03\ c1\ 00\ 12\ a7\ 00\ 00\\ Rx: 80\ 60\ f1\ 03\ c1\ 00\ 12\ a7\ 00\ 00\ 00\ 00\ 00\ 00\ 00\ 00\ 00\ 0$	
Data By Common Identif	ier
LCD_STATE	
LCD_BRIGHTNESS	
KEYPAD_01_WAKEUP	
E KEYPAD_02_WAKEUP	WRITE
KEYPAD_03_WAKEUP	
KEYPAD_04_WAKEUP	
f IGN_WAKEUP	03
RTC_WAKEUP	

In the LCD module, the LCD state variable is selected. This variable as defined in the section 6.12.4 is used to turn on the LCD.

Once the DB variable is selected, the appropriate DB field ID value will be displayed in the Datalayer ID text box available in the area marked by "Data By Common Identifier".

Dept: Software	User Manual Document	Rev No. Rev 3	Rev 3.0
		Date:	Oct 17, 2022

12:05 📿 📾 🦁 🗹	ⓒ ﷺ ≵ ಔ •⅔ 2
U START U	STOP
$ \begin{array}{c} \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 12 & o7 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & 10 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 & c1 & 00 \\ \dot{R}_{c} & 80 & 60 & f1 & 03 & c1 & 00 \\ \dot{R}_{$	
Data By Comi	mon Identifier
DB Variables LCD_STATE	•
Data Layer ID	47
Buffer 01	READ WRITE
Data By Mer	mory Address
Address 1011	Size 03
READ	

To enable the LCD\_STATE in the Buffer text box, hit the write button. The value gets written to the Data base in the AI280 module and the LCD turns ON.

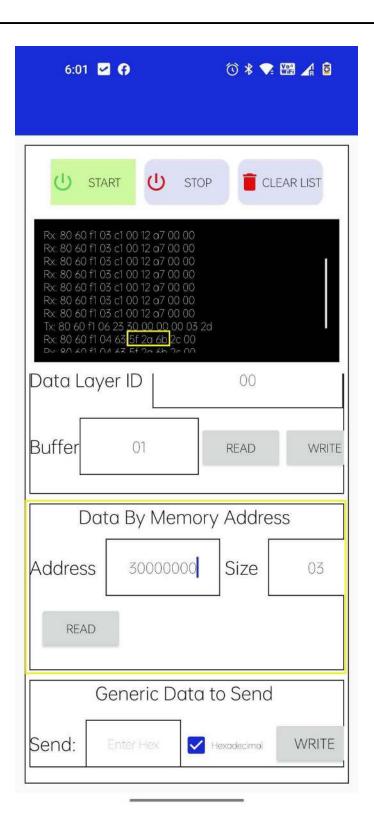
Dept: Software	User Manual Document	Rev No.         Rev 3.0           Date:         Oct 17, 2022	Rev 3.0
			Oct 17, 2022
		•	4

# 9.5 Read/Write by Memory Address Screen

The screen marked by yellow shows the Memory Access Area. If the user needs to read any memory location, he can directly provide the address in the address field and the size value in the size text field and then click read. The data received from the device would be listed in the TX/RX area. In the above illustration the "Data By Memory address" section has an address and size field.

To read the data of the Memory Location, enter the Address and the Size of the variable under consideration. Click the READ Button for the Selected Memory Address. The terminal will reflect the communication between the device and the Mobile App. The values present on the device shall also be reflected on the Terminal. For example, if we want to read 3 bytes from the address 0x30000000, we will update the address and size as shown below and then click read. The result can be got from the TX/RX area.

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022



Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

The first 4 bytes in the Tx/Rx data carries the header field and the 5<sup>th</sup> byte in each variable represents the request packet. The next few bytes represent the Data, the last byte denotes the checksum value.

Tx : 80 60 fl 06 23 30 00 00 00 03 2d

RX : 80 60 f1 04 63 5f 2a 6b 2c 00

Please see the below screen which shows that the value at memory location 0x30000000 is "5f 2a 6b" as received in our response packet above.

	0x30000000			~	Go	New Tab	
	0x30000000 <t< th=""><th>raditional&gt;</th><th>0x30000000 &lt; Traditional &gt; X</th><th></th><th>histopend</th><th></th><th></th></t<>	raditional>	0x30000000 < Traditional > X		histopend		
	0x30000000	5F 2A 6B 2D	84 E7 92 84 95 58 A4 48 7A 98 1A A1 E8 A2	* <b>k</b> ç	.X¤H	z¦è⊄	^
t	0x30000012	A0 1E 5B 06	90 DF EB 50 AD 93 58 EB 59 C7 8D DA 6F EC	.[ßëP	-,Xë	YÇ.Úoì	
	0x30000024	5F 5B B8 00	A9 44 5C 29 09 BF 84 53 47 F3 12 51 28 B7	[OD\)	. ¿. S	Gó.Q(·	
	0x30000036	B2 37 8C 44	70 E8 23 CE 2F D1 C5 E6 56 78 50 DE 63 77 2	7.Dpè#Î	/ÑÅæ	V{P⊵cw	
	0x30000048	95 7F 94 F7	11 69 E1 CC 11 47 20 68 72 B7 A1 8C 3B 60 .	÷.iáÌ	.G h	r•[.;`	-
	0x3000005A	C5 2A 16 6E	FF 96 46 6C 71 17 E5 63 8A 61 AC C5 F9 21 Å	*.nÿ.Fl	q.åc	.a-Åù!	
	0x3000006C	78 4F D6 AF	A3 0F 1F 10 4C 94 30 26 40 3E E3 59 48 86 x	00 <sup>-</sup> £	L.08	@≻ãYH.	
	0x3000007E	C3 D1 B8 DC	16 D5 7C 96 23 B1 B1 32 A5 61 9F 1A 0E 11 Å	Ŵ,Ū.Õ .	#±±2	¥a	
	0x30000090	B4 1C 53 F0	1E F5 83 89 5A 75 49 58 F3 6A C0 6A B6 96	. <b>Sð.</b> ö.¹	ZuIX	ójÀj9.	
5	0x300000A2	81 65 D5 6D	87 97 84 82 8E 33 F9 E9 3C 42 7E 7A 0A 40 .	eÕm	.3ùé	<b~z.@< td=""><td></td></b~z.@<>	
	0x300000B4	9D CD F8 CF	D8 90 F8 0F B6 CE E7 2F 25 9D ED A6 D1 76 .	Í¢ĪØ.¢.	¶Îç∕!	%.í¦Ñ∨	
	0x300000C6	AØ 38 89 CA	85 FC 90 96 60 8A C7 D6 35 63 91 35 8E BB	8.Ê.ū	`.¢ö	5c.5.»	
3	0x300000D8	<b>09</b> 86 67 04	FD FB FF 6F 21 F4 60 05 7C 37 57 BD 77 EC .	.g.ýůÿo	!ô`.	7W%wi	
	0x300000EA	BB 12 B4 7C	67 8C 40 68 79 8A 4D E4 1A 01 33 0C C4 83 »	. 1 g.@h	y.Mä	<mark>З</mark> .Ä.	
-	0x300000FC	FE 8A D2 F7	45 9C D2 F1 13 A9 BB FF B6 9A 62 DA E8 25 þ	.Ò÷E,Òñ	.©»ÿ	J.búè%	
	0x3000010F	17 18 F4 C8	A1 4F D4 12 85 21 36 61 6B 4A B1 31 D1 63 .	.äÈ:0Ô.	. 16a	k∃+1Ñc	V

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 9.6 Generic Data to send

The screen marked by black shows the Generic Data to send section where any generic BLE hex data can be sent to the device.

For example, we will send a hex data to the BLE device.

TX : 80 60 51 06 50 41 49 34 33 30

The first 4 bytes are header, the next is the request packet followed by the data.

18:29			© •	· 🔐 🖅 🎸
U START	Ċ	STOP	C C	LEAR LIST
Tx: 80 60 ft 01 81 53 Rx: 80 60 ft 03 ct 00 Tx: 80 61 51 06 50 41 Rx: 80 60 ft 03 ct 00	49.34 3	3 30		
KEY_PRESS_01				-
Data Layer ID	)		002B	
Buffer 0	1		READ	WRITE
Data By	/ Me	mory	Addre	ess
Address	1011		Size	03
READ				
Generic Data to Send Send: 806151065041493 43330 Hexadecimal WRITE				

Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

When this data the received by the MAXAI280 board, it is translated to text and displayed in the UI by the Bluetooth App as shown in the below image.

In the below illustration it is understood that the given hexadecimal value is converted into (string value) and the same is been displayed on the device.



Dept: Software	User Manual Document	Rev No.	Rev 3.0
		Date:	Oct 17, 2022

# 9.7 Clear list and Stop testing

Selecting clear list would clear the RX/TX terminal so that the user can see the latest data.

Press the STOP button to stop the testing as shown in the below image.

