# MPC5775B BMS plus VCU Reference Design User Guide



# Contents

Chapter 1 Introduction	3
Chapter 2 Hardware User Guide	4
Chapter 3 Software User Guide	25
Chapter 4 GUI user guide	
Chapter 5 Bootloader User Guide	41

# Chapter 1 Introduction

This user guide gives an overview of hardware and software for the powertrain domain controller reference design based on MPC5775B MCU integrated the BMS and VCU in one ECU. The reference design is intended to provide a mechanism for easy customer evaluation of the Cobra55 family of microprocessors, and to facilitate hardware and software development.

## 1.1 List of acronyms

The following table provides a list and description of acronyms used throughout this document.

#### Table 1. Acronyms and abbreviations

Acronym	Description
PDC	Powertrain Domain Controller
EV	Electric Vehicle
BMS	Battery Management System
VCU	Vehicle Control Unit
SBC	System Basis Chip
TPL	Transform Physical Layer
BCC	Battery Cell Controller
RTC	Real Time Clock
DNP	Do Not Populate
CAN	Controller Area Network
ECU	Electronic Control Unit
MSDI	Multiple Switch Detection Interface
РНҮ	Physical

# Chapter 2 Hardware User Guide

## 2.1 Overview





## 2.2 Board features

The system mainly consists of a master board and several slave boards connected in series with a daisy-chain with TPL.

The master board provides the following key features:

- NXP MPC5775B Microcontroller (416 MAPBGA soldered)
- · Support Daisy chain of BCC device connection with Transformer isolation
- · Integrate NXP Power SBC FS65xx with one CAN physical layer
- Master power regulator status LEDs
- · One user reset switch with reset status LEDs
- · Two user LEDs is connected to GPIO for test
- · Standard 14-pin JTAG debug connector
- · One Mbit Automotive external EEPROMs with SPI
- 1 Automotive tiny Real-Time Clock/calendar with alarm function and I<sup>2</sup>C-bus
- Integrate multiple communication interfaces:
- 1 x 100BASE-T1 Automotive Ethernet interface TJA1101
- 3 x CAN interfaces, two is supported via NXP CANFD transceiver TJA1052 and TJA1145, another is supported via FS65xx.
- 1 x LIN interface MC33662BLEF

MPC5775B BMS Plus VCU Reference Design User Guide, Rev. 0, February 2020

- · 1 x Mini USB / UART FDTI transceiver to interface with MCU
- · Integrate input/output devices:
- 33 channel multiple switch detection interface devices(MC33CD1030)
- Quad high side with SPI Control(MC15XS3400)
- 16-output low side switch with SPI Control(MC33996)
- A 130-pin ECU connector, routing external I/O signals including:
- 10 x ADC input channels
- 16 x Switch input channels
- 2 x PWM input capture channels
- · At least 2 x PWM output channels via HSD and LSD
- 5 x 5V + 1 x 12 V power supply for external sensor
- 4 x HSD output channels
- · 11 x LSD output channels

#### NOTE

14 channels of MC33CD1030 is routed to 130-pin connector and 11 channels of MC33996 is routed to 130-pin connector.

## 2.3 Module introduction

## 2.3.1 Power SBC

#### 2.3.1.1 Board power

The MC33FS6523CAE provides a robust, scalable power management to the MPC5775B MCU with fail silent safety monitoring measures making it fit for ASIL-D. It consists of multiple switching and linear voltage regulators.

When external power, typical 12 V automotive DC power supply with current limit > 600 mA, is applied to jumper J8 connector of master board.

- VBAT pins connects to DC power supplier 12 V+.
- · GND pins connects to DC power supplier GND,
- Keep Jumper J1 onboard.

Five green power LEDs show the presence of the supply voltages as follows:

- LED D55 Indicates that the 12 V external power is enabled and working correctly.
- · LED D56 Indicates that the 3.3 V linear regulator is enabled and working correctly.
- LED D57 Indicates that the 5.0 V linear regulator is enabled and working correctly.
- LED D58 Indicates that the 5.0 V linear regulator dedicated to the internal CAN FD interface is enabled and working correctly.
- LED D59 Indicates that the 1.25 V switching regulator is enabled and working correctly.

The following figure shows the power LED's circuit in detail.



## 2.3.1.2 VCCA, VAUX voltage configuration

The voltage of V<sub>CCA</sub> and V<sub>AUX</sub> is configurable through the external resistor(**R8**) connected to **SELECT pin**. The default value of **R8** on the board is 24K, it means V<sub>CCA</sub>=3.3 V, V<sub>AUX</sub>=5.0 V. The following table show the configuration of V<sub>CCA</sub>, V<sub>AUX</sub> voltage, if the SELECT pin is detected open, the V<sub>CCA</sub> regulators start at their minimum output voltage 3.3 V.

 $V_{CCA}/V_{AUX}$  voltage selection

V <sub>CCA</sub> (V)	V <sub>AUX</sub> (V)	R select	Recommended value
3.3	3.3	< 6.0 kΩ	5.1 kΩ ±5.0%
5.0	5.0	10.8 << 13.2 kΩ	12 kΩ ±5.0%
3.3	5.0	21.6 << 26.2 kΩ	24 kΩ ±5.0%
5.0	3.3	45.9 << 56.1 kΩ	51 kΩ ±5.0%

#### Table 2. V<sub>CCA</sub> and V<sub>AUX</sub> configuration

#### 2.3.1.3 Debug mode

The jumper **J1** is used to configure the DEBUG mode of SBC, keep it onboard as shown in the following figure to activates the Debug mode. Once activated, there is no deep fail-safe state.



## 2.3.1.4 Reset circuit

When the specific condition is reached, the SBC will reset MCU via SBC\_RSTB pin. The following figure shows the board supports multiple MCU reset methods like SBC, JTAG and Switch Reset(SW1).



## 2.3.1.5 Communication interface

FS6523(U1) has a high speed CAN transceiver. It is routed to the MCU FlexCAN\_B port(AC19, AD19). The pin-out of the 130-pin connector (J8) is C11, D11.

The SBC SPI signal is transferred to the MCU through DSPI2 with CS0. MCU uses the interface to control SBC watchdog and so on.



## 2.3.2 MCU power and GND configuration

The V<sub>DDEH</sub> and V<sub>DDE</sub> is MCU I/O supply voltage connected to V<sub>CCA</sub>(3.3V) from SBC, and V<sub>DDA\_EQ</sub> is eQADC supply voltage connected to V<sub>AUX</sub>(5V).

## 2.3.3 MCU external clock circuit

In addition to the internal 16 MHz oscillator, the MCU can also be clocked by external oscillator (Y1). The clock circuitry for the 40 MHz crystal is shown in the following figure. The symbol named "DNP" indicates the device is not mounted to the PCB.



## 2.3.4 JTAG connector

The master board is fitted with 14-pin JTAG debug connector. The following figure shows the 14-pin JTAG connector pinout.



## 2.3.5 USB/UART transceiver

Mini USB connector interfaced with FT232RQ is connected to MCU eSCI\_C port(AD22, AF23), USB to serial UART chip (U17). The USB to serial UART connection is shown in the following figure. Can connect to PC via mini USB to put string on screen or get string from screen by Serial Debug Assistant.



## 2.3.6 CAN transceiver

The master board has two high speed CAN transceivers.

- TJA1052i(U5) is a galvanically isolated interface between CAN controller and the physical two-wire CAN bus, and connected to MCU MCAN0/FlexCAN\_A port (AE19, AF19).
- TJA1045T/FD(U6) is a High-speed CAN transceiver for partial networking, and connected to MCU MCAN1/FlexCAN\_D port (AC20, AD20). TJA1045T/FD must be drove through DSPI. The SPI signal is transferred to the MCU through DSPI0 with CS2, MCU use the interface to configure the PHY.





## 2.3.7 LIN transceiver

The master board is fitted with a NXP MC33662LEF LIN transceiver (U7) connected to MCU eSCI\_A port(M2, M3). Configure MCU eSCI\_A module and assert LIN1\_EN pin to enable the LIN communication.



## 2.3.8 Ethernet PHY

The master board includes support for 100M-base T1 Automotive Ethernet PHY TJA1101(U9), utilizing the normal MII lite. It is clocked by 25 MHz external oscillator (Y2). The default physical address is configured as 0b00xx0 by the pull-down or pull-up resistor on the pin.

A pair of differential signal(EPHY\_TRX\_P, EPHY\_TRX\_M) is routed to 130-pin ECU connector.



## 2.3.9 eQADC interface

The master board enables nine eQADC channels from ANA0 to ANA8. The first three channels are routed to MSDI\_AMUX, HSD\_CSNS and SBC\_MUXOUT port respectively. The others channels are routed to J8 to measure external analog signal, such as pedal, sensor signal, through ADC input signal process circuit (as shown in the following figure).

In addition to MCU\_AIN4 channel supports external 0-12 V analog signal input from J8, other channels only support 0-5 V.



## 2.3.10 Digital signal interface

The master board supports to detect external switching signal, such as gear signal, through clamp circuit to MCU port (MCU\_DIN0-MCU\_DIN5).

In addition to MCU\_DIN4 and MCU\_DIN5 channels support 0/12V digital signal input from J8, other channels only support 0/5V (as shown in the following figure).



## 2.3.11 MSDI

The master board has a CD1030 to detect the closing and opening of multiple switch contacts and analog input signal. The input signal is transferred to the MCU through DSPI2 with CS1.

As shown in the following figure, the master board enables SG0-SG9 as digital input, SP0-SP4 as analog input, these port is connect to J8, and MCU read switch status via DPSI2, analog signal via MSDI\_AMUX through eQADC channel 0 (ANA0).



## 2.3.12 PWM input

The master board has two PWM input circuit to detect external PWM signal, the MCU\_PWM\_IN0/1 is routed to MCU eMIOS channels 1/2. The user can use eMIOS mode Pulse width measurement mode to measure PWM signal, its amplitudes range from 3.3 V to 12 V.



#### 2.3.13 HSD (extreme switch)

The system requires several channels to drive external load. Provide high current driving capability that could control these devices directly.

Generally load as:

- Main Contactor 12 V dc coil voltage rating, 1~2A coil current at ON.
- Pre-Charge Contactor 12 V dc coil voltage rating, 0.5 A coil current at ON.
- Cooling Fans 1 2 V dc, 4.8 A
- Heater 12 V dc 40 W

The master board use an extreme switch devices, NXP Quad high-side switch MC15XS3400(U15), to drive these loads, it is supplied by 12 V dc power. It can provide four channel high side outputs, each channel up to 6 A current.

MCU can configure DSPI2 with CS2 to control it, and also supports the direct control to the output pins (HS0-HS3) via the HSD\_IN0-HSD\_IN3 with GPIO.



## 2.3.14 LSD

The system requires several channels to control external contactor or relays, such as AC power switch, DCDC relay, etc. So the master board is fitted with a NXP 16-output low side switch with a 24-bit serial input control MC33996 (U16), MCU can configure DSPI2 with CS3 to control it. The PWM pin – LSD\_PWM is routed MCU eMIOS channel three and the output of the output PWM frequency is up to 2.0 KHz.



## 2.3.15 TPL

Due to high voltage of battery packs, isolation is very important. This reference design isolate power supply and communications. TPL communication daisy chain between slave boards is above 3750 V dc by device.

The MC33664 is a transceiver physical layer transformer driver designed to interface a MCU conveniently to a high speed isolated communication network. It is needed for dual SPI for communication with MCU by half-duplex.

TPL communication port between master board and slave boards, the baud rate is 2 Mbps. It is linked as daisy chain with one or more slave board. The board has two TPL(U10, U11) routed three MCU DSPI, includes DSPI0 with CS0 for master, DSPI1 with CS0 for TPL1 slave, DSPI3 with CS0 for TPL2 slave.

#### MPC5775B BMS Plus VCU Reference Design User Guide, Rev. 0, February 2020



## 2.3.16 RTC

The master board is fitted with a NXP automotive RTC PCA85063(U3) with  $I^2C$  communication, but the MPC5775B has no  $I^2C$  interface, so it can only be implemented by using the GPIO(V25, V26) to simulate  $I^2C$  protocol. The default devices address is 0x51.



## 2.3.17 NVM

The master board has an external NVM – M95M01, an automotive 1024 Kbit SPI bus EEPROM with high speed clock. MCU can read and write data through DSPI0 with CS1.



## 2.4 130 pin ECU connector

There is a 130-pin ECU connector onboard, consisting of two parts of header M1 and M2.



When the connector is inserted into the cable, the two supporting cables are installed in sequence. The M2 corresponding cable has two pairs of differential twisted pair wires for connecting the BCC slave board. The following figure show the cables installation.



The following table lists all the connections to the master board interface connector .

Table 3.	Board	interface	connect	or c	letai	S
----------	-------	-----------	---------	------	-------	---

Connector	Signal	Description	Connector	Signal	Description
A11	VBAT	Power supply	A21	GND	Digital GND
A12	VBAT	Power supply	A22	GND	Digital GND
A13	VBAT	Power supply	A23	GND	Digital GND
A14	GND	Digital GND	A24	GND	Digital GND
A15	GND	Digital GND	A25	GND	Digital GND

Table continues on the next page ...

Connector	Signal	Description	Connector	Signal	Description
B11	GND	Digital GND	B21	AGND	Analog GND
B12	GND	Digital GND	B22	AGND	Analog GND
B13	GND	Digital GND	B23	AGND	Analog GND
B14	GND	Digital GND	B24	AGND	Analog GND
B15	GND	Digital GND	B25	AGND	Analog GND
C11	SBC_CAN_L	Flex CAN1	C21	Reserved_AIN0	MSDI SP0
C12	CAN1_L	MCAN0	C22	Reserved_AIN1	MSDI SP1
C13	CAN2_L	MCAN1	C23	CRUSH_SS_AIN	Crush sensor signal
C14	NC	Not connected	C24	Reserved_AIN2	MSDI SP2
C15	LIN1	LIN signal output	C25	Reserved_AIN3	MSDI SP3
D11	SBC_CAN_H	Flex CAN1	D21	ALT_PP_AIN1	Accelerator pedal signal 1
D12	CAN1_H	MCAN0	D22	ALT_PP_AIN2	Accelerator pedal signal 2
D13	CAN2_H	MCAN1	D23	BRK_PP_AIN1	Brake pedal signal 1
D14	NC	Not connected	D24	BRK_PP_AIN2	Brake pedal signal 2
D15	KL15_WAKE	SBC IO0 (Wake-UP pin)	D25	VCM_PPS_AIN	Vacuum pump pressure sensor
E11	GND	Digital GND	E21	AGND	Analog GND
E12	DRIVE_GEAR_DI N	Driver gear switch input	E22	AGND	Analog GND
E13	REVERSE_GEA R_DIN	Reverse gear switch input	E23	AGND	Analog GND
E14	PARK_GEAR_DI N	Parking gear switch input	E24	AGND	Analog GND
E15	NEUTRAL_GEA R_DIN	Neutral gear switch input	E25	AGND	Analog GND
F11	EPHY_TRX_P	Ethernet Differential signal (+)	F21	GND	Digital GND
F12	GND	Digital GND	F22	GND	Digital GND

Table 3. Board interface connector details (continued)

Table continues on the next page...

Connector	Signal	Description	Connector	Signal	Description
F13	GND	Digital GND	F23	GND	Digital GND
F14	BRAKE_SW_DIN	brake switch input	F24	GND	Digital GND
F15	HAND_BRAKE_D IN	Hand brake switch input	F25	GND	Digital GND
G11	EPHY_TRX_M	Ethernet Differential signal (-)	G21	KEY_ON_DIN	MSDI SG0
G12	GND	Digital GND	G22	AC_SW_DIN	MSDI SG1
G13	ALT_PS_5V0	5V supply for Accelerator pedal	G23	CRUISE_CTL_DIN	MSDI SG4
G14	ALT_PS_5V0	5V supply for Accelerator pedal	G24	CHARGE_CC_DIN	MSDI SG3
G15	GND	Digital GND	G25	PTC_SW_DIN	MSDI SG2
H11	GND	Digital GND	H21	PWM_IN0	PWM input
H12	VCM_PPS_5V0	5V supply for vacuum pump pressure sensor	H22	PWM_IN1	PWM input
H13	BRK_PS_5V0	5V supply for Brake pedal	H23	GND	Digital GND
H14	BRK_PS_5V0	5V supply for Brake pedal	H24	GND	Digital GND
H15	CRUSH_SS_12V	12V supply for crush sensor	H25	GND	Digital GND
J11	GND	Digital GND	J21	KEY_START_DIN	MSDI SG5
J12	GND	Digital GND	J22	Reserved_DIN0	MSDI SG6
J13	GND	Digital GND	J23	Reserved_DIN1	MSDI SG7
J14	GND	Digital GND	J24	Reserved_DIN2	MSDI SG8
J15	GND	Digital GND	J25	Reserved_DIN3	MSDI SG9
K11	AC_PWR_RLY_D OUT	LSD output 2	K21	NC	Not connected
K12	WATER_PUMP_ RLY_DOUT	LSD output 1	K22	NC	Not connected
K13	DCDC_EN_DOU T	LSD output 0	K23	NC	Not connected

#### Table 3. Board interface connector details (continued)

Table continues on the next page ...

Connector	Signal	Description	Connector	Signal	Description
K14	LSD_REV_OUT4	LSD output 10	K24	NC	Not connected
K15	LSD_REV_OUT3	LSD output 9	K25	NC	Not connected
L11	AC_OUTLET_EM _RLY2_DOUT	LSD output 5	L21	NC	Not connected
L12	LSD_REV_OUT2	LSD output 8	L22	NC	Not connected
L13	AC_OUTLET_EM _RLY1_DOUT	LSD output 4	L23	NC	Not connected
L14	LSD_REV_OUT0	LSD output 6	L24	NC	Not connected
L15	LSD_REV_OUT1	LSD output 7	L25	NC	Not connected
M11	GND	Digital GND	M21	TPL2_RDTX_N	TPL2 Differential signal (-)
M12	GND	Digital GND	M22	TPL2_RDTX_P	TPL2 Differential signal (+)
M13	GND	Digital GND	M23	NC	Not connected
M14	GND	Digital GND	M24	TPL1_RDTX_N	TPL1 Differential signal (-)
M15	GND	Digital GND	M25	TPL1_RDTX_P	TPL1 Differential signal (+)
N11	AC_PTC_RLY_D OUT	LSD output 3	N21	NC	Not connected
N12	COOL_FAN_DRV _DOUT	HSD output 3	N22	NC	Not connected
N13	HEAT_DRV_DO UT	HSD output 1	N23	NC	Not connected
N14	PRECH_CTT_DO UT	HSD output 2	N24	NC	Not connected
N15	MP_CTT_DOUT	HSD output 0	N25	NC	Not connected

Table 3. Board interface connector details (continued)

## Chapter 3 Software User Guide

This reference design provides two sample application projects.

- S32 SDK(support FreeRTOS)
- AUTOSAR MCAL

## 3.1 SDK project

#### 3.1.1 Software and tools requirement

The following is list of tools and software required for SDK project.

- IDE: S32 Design Studio for Power<sup>®</sup> Architecture Version 2017.R1.
- SDK: S32\_SDK\_S32PA\_RTM\_3.0.0.
- Compiler: GCC E200 VLE GNU Compiler 4.9.4.
- Debugger: P&E multilink (with P&E GDB Server), Lauterbach TRACE32 JTAG debugger (use run.cmm script to debug).
- CAN Adaptor: PCAN-USB Pro.

NOTE

There is an issue in SDK drivers source code. Please modify the macro named PIT\_CLOCK\_NAMES from PITRTI0\_CLK to PER\_CLK in line 178 of the *MPC5775B\_features.h* header file in which <u>SDK/platform/devices/</u><u>MPC5775B/include/</u>\_

Project Explorer 🛛 🕞 🛸 🖓 Compo	nent Inspector - pin mux 🛙			Basic Advanced 🍱 🗇 🗸 🍟 🗖	B Out 🛙
PDC 5775B SDK Z7 0: Debug	Eunctional Properties Methods Se	ttings			
> S Binaries		i i i i i i i i i i i i i i i i i i i		C ( D )	An outline is
>   Includes	Ode Options	6 H C H		Generate Report	available.
> @ Generated_Code	Ipsed O Pins Show Only Cor	ntigurable Signals		HIML Report	
Project_Settings		AN Nexus SILL SRX SoC eMIOS	sci		
> @ SDK					
> 🖴 Sources Signals		Pin/Signal Selection	Direction	Selected Pin/Signal Name	
> 😂 include 🛛 🗸 🗸 CAN_	0				
> 😑 Debug 🛛 👘 Flex	CAN Receiver Input	No pin routed	Input		
> 🖉 Debug_RAM Flex	CAN Transmitter Output	No pin routed	Output		
> 🖉 Documentation 🛛 🗸 🗸 CAN_	1				
> 🖉 SourceInsight 🛛 🛛 🕹 Flex	CAN Receiver Input	PC[22]	Input	PC[22]	
% ProcessorExpert.pe Flex	CAN Transmitter Output	PC[21]	Output	PC[21]	
☑ README.txt	2				
Flex	CAN Receiver Input	No pin routed	Input		
Components - PDC_5775B 🛛 👘 Flex	CAN Transmitter Output	No pin routed	Output		
🗏 🕭 🖏 👻 🗸 🗸 CAN_	3				
Generator_Configurations Flex	CAN Receiver Input	No pin routed	Input		
WPC5775B_416 Flex	CAN Transmitter Output	No pin routed	Output		
PreekTOS:FreekTOS					
Processors					
> @ Cpu:MPC5775B_416					
Components	s 🔊 Tasks 📮 Console 💥 🔲 Proper	ties 🗷 Debugger Console 🎄 Debug 📟 Dis	assembly 🖞 History 🖼 Progress	🗎 🖬 📴 I	- 🗉 👻 📑 🕶
> B Referenced_Components		ales a bebugger console a bebug - bis	ussenibly a history orregress		
> @ pin_muxPinSettings Could no	ot set item value - unsupr	ported number format (comp: Mcan	Com1. item: phaseSeg1CBT0.	value: '0') - Unknown char: '	
> 12 clockMan1:clock_manager Could no	ot set item value - unsupp	oorted number format (comp: Mcan	Com1, item: phaseSeg2CBT0,	value: '0') - Unknown char: '	
intMan1:interrupt_manager Could no	ot set item value - unsupp	oorted number format (comp: Mcan	Com1, item: propSegCBT0, v	alue: '0') - Unknown char: '	
> @ esci2:esci					
> 🔟 pit:pit					



## 3.1.2 File structure

There are two folders in the top-level directory named PDC\_5775B\_SDK, which contain two projects - core 0 projects (in PDC\_5775B\_SDK\_Z7\_0 folder) and core 1 projects (in PDC\_5775B\_SDK\_Z7\_1 folder). The provided demo codes only run on core 0. The following table shows the file structure.

#### Table 4. SDK project files

Folders/Files	Descriptions
.settings	contains customized project settings.
bin	contains generated object file.
bin/bootloader_bin	contains bootloader binary files.
Documentation	contains automatically generated documents by S32DS.

Table continues on the next page ...

#### MPC5775B BMS Plus VCU Reference Design User Guide, Rev. 0, February 2020

#### Table 4. SDK project files (continued)

Folders/Files	Descriptions
Generated_Code	contains all generated codes by ProcessorExpert.pe tool.
include	contains the application header files with pdc_ prefix.
Project_Settings	contains two version (bootloader and non-bootloader) files - linker, startup code and debugger configurations.
SDK	contains the SDK source codes used in this project.
Sources	contains the application source codes, related drivers folders.
ProcessorExpert.pe file	ProcessorExpert.pe tool.
run.cmm	is a script used for Lauterbach debugger

NOTE

Please unpack "Bootloader\_Settings.zip" in Project\_Settings folder to replace *linker\_flash.ld* and *startup\_MPC5775B.S* for bootloader support project. If you do not want to enable bootloader, unpack "Non\_Bootloader\_Settings.zip".

#### Table 5. Source files

Files	Descriptions
Files with pdc prefix	contain codes for PDC applications.
Files with bcc prefix	contain drivers of MC3377x, include MC33772, MC33771B and MC33771C.
Files with sbc prefix	contain drivers of MC33FS65xx.
Files with boot prefix	contain codes for bootloader.
Files with lin prefix	contain codes for LIN driver implement since SDK has no LIN driver yet.
Files with i2c prefix	contain codes for I <sup>2</sup> C protocol.
Files with rtc prefix	contain codes for PCA85063 driver.
Files with msdi prefix	contain drivers of CD1020, CD1030, MC33978 and MC34978.
Files with nxp prefix	contain nxp_helpers codes used in SBC drivers.

## 3.1.3 Downloading exist elf file

The following steps must be followed to download the exist elf file to MCU using S32DS.

1. Launch the S32DS IDE, click on the Flash icon, the Flash Configurations window appears.

ြဲ Project Explorer 🛛				
	W Flash Configurations			
	Create, manage, and run configurations			
		Configure launch settings from this dialog:		
	GDB PEMicro Interface Debugging	Press the 'New' button to create a configuration of the selected type. Press the 'Dunlicate' button to conv the selected configuration		
		<ul> <li>Press the 'Delete' button to remove the selected configuration.</li> </ul>		
		🕸 - Press the 'Filter' button to configure filtering options.		
		- Edit or view an existing configuration by selecting it.		
		Configure launch perspective settings from the <u>'Perspectives'</u> preference page.		
5- Components ⊠				
	Filter matched 1 of 9 items			
	(?)		Flash	Close

2. Create a new configuration, adjust its name and browse for elf/srec/hex file.

Hash Configurations		×				
Create, manage, and run configurations		5				
	Name: PDC_5775B_FLASH					
type filter text	🗎 Main 🔅 Debugger 🕨 Startup 🦆 Source	Common # OS Awareness				
<ul> <li>Is in the second second</li></ul>	Project:					
		<u>B</u> rowse				
	Specify the number of additional object files y	you wish to program: 0 Generate Object File Fields				
		~				
	C/C++ Application:					
	C:\Users\NXF47394\Desktop\release_02\PDC-5	SW-Package-v1.5.0\MCU\PDC_5775B_SDK\PDC_5775B_SDK_Z7_0\b				
		<u>V</u> ariables Searc <u>h</u> Project <u>Br</u> owse				
	Build (if required) before launching					
	Build Configuration: Select Automatically	~				
	O Enable auto build	O Disable auto build				
	Use workspace settings	Configure Workspace Settings				
Filter matched 2 of 10 items		Re <u>v</u> ert Appi <u>y</u>				
0		Flash Close				
Figure 29. Create new configuration						

3. Select MCU and specific core you are targeting.

cate, manage, and ran configurations		
Plugin has not been registered. Some functiona	ality may not be available.	
	Name: PDC_5775B_FLASH	
pe filter text	🕒 Main 🎋 Debugger 🕒 Startup 🦅 Source 🗖 Common 🥮 OS Awareness	
GDB PEMicro Interface Debugging	Software Registration	
New_configuration	Please register your software to remove this message.	
	Register now	
	DEMicro Interface Settings	
	Interface: UISB Multilink USB Multilink FX Embedded OSBDM/OSITAL X Compatible Hardware	
	Det USD1 Multilik Usigerel SV Dev C (DEME22234)	
	Port: USB1 - Multilink Universal FX Rev C (PEME32324) V Reliesh	
	Device Name: MPC5775B Core: Z7_0	
	Specify IP Specify Network Card IP	
	Additional Options	
	Advanced Options	
	Hardware Interface Power Control (Voltage> Power-Out Jack)	

4. Click on Flash button to proceed to programming. As soon as the device is programmed it gets disconnected.

🖹 Problems 🧟 Tasks 📮 Console 🛱 🔲 Properties 🗟 Debugger Console 🖘 Progress 🖋 Search 🚦 History 🎂 Git Staging 🎘 Git Repositories 🦧 Problem Details 🚟 Disassembly
<terminated> PDC_5775B_FLASH [GDB PEMicro Interface Debugging] C:\NXP\S32DS_Power_v2017.R1\eclipse\plugins\com.pemicro.debug.gdbjtag.ppc_1.8.3.201812312131\win32\pegdb</terminated>
Reset script (C:\NXP\S32DS_Power_v2017.R1\eclipse\plugins\com.pemicro.debug.gdbjtag.ppc_1.8.3.201812312131\win32\gdi\P&E\s32e200_mpc5777c.mac) completed.
MPC5777C Device detected. No breakpoints currently set. Disconnected from "127.0.0.1" via 127.0.0.1 Target Disconnected.
Figure 31. Flash programming done

#### 3.1.4 Basic workflow of demo project

MCU Init initializes system clock, pins, eDMA and UART. Then it starts to create BSP task, initialize the remaining modules in BSP task, run BSP\_TestOnce() function for some module that need to run only once. Finally it creates other tasks that system requires. Start the RTOS scheduler.

SDK project uses FreeRTOS to implement multitasking. The project mainly has four tasks VCU, BMS, BSP and UDS. Software timer is used to trigger BMS (50ms), VCU(10ms), UDS(100ms) and BSP(300ms) tasks. The callback function of the system software timer is called every 10ms, and schedules multitasking in the timer callback.



- At power up, the SBC watchdog should properly be refreshed during the first 256ms open window.
- VCU task checks gear position, accelerator pedal position, brake pedal position and VCU faults information. Then it calculates the expected output torque. It determines the output, that is, reduce torque and shut down some accessories according to system fault level.
- BMS task monitors stack voltage, cell voltage, temperature and BMS fault information. Support daisy chain of MC33771C or MC33771B or MC33772.
- UDS task monitors specific messages on CAN bus once every 100ms. Once having received commands on CAN bus, UDS Task will execute UDS protocol and jump to bootloader.

**NOTE** The default GCC optimization is O2 in the demo project.

## 3.2 MCAL project

## 3.2.1 Software and tools required

The following software and tools are required for MCAL project.

- Configuration tool: EB tresos Studio 20.1.0.
- MCAL Drivers: MPC5777C\_MCAL4\_0\_RTM\_HF4\_1\_0\_1.
- Compiler: Green Hills Compiler 2018.5.
- Debugger: Lauterbach TRACE32 JTAG Debugger.
- CAN Adaptor: PCAN-USB Pro.

#### NOTE

There is an issue in MCAL4.0. Please replace *Adc\_Eqadc.c* with the provided one in fix folder to enable ADC calibration functions.

MPC5775B BMS Plus VCU Reference Design User Guide, Rev. 0, February 2020





## 3.2.2 File structure

#### Table 6. MCAL project files

Folders/Files	Descriptions
.metadata	contains log information of Tresos.
bin	contains generated object files.
bin/bootloader_bin	contains bootloader binary files.
cfg	contains configuration files generated by Tresos.
fix	contains files to fix bugs.
include	contains application header files with pdc_ prefix.
make	contains makefiles used for building the application.
src	contains application source code files.
toolchains	contains files needed to build with various toolchains (startup, linker command files).
Tresos	contains the Tresos project with the application configuration.
makefile	is the MCAL sample application makefile.

Table continues on the next page ...

#### Table 6. MCAL project files (continued)

Folders/Files	Descriptions
Modules	specifies which modules are compiled and linked.
make.bat	launches the make command.
launch.bat	contains user-defined paths for compilation and launches the make.bat file.
clear.bat	clears all generated and built files.
compiler_used.s	specifies which complier is used.
run.cmm	is a script used for Lauterbach debugger

**NOTE** For bootloader supported project, please unpack "Bootloader\_Settings.zip" in toolchains older to replace *autosar\_flash.ld, Startup\_vle.c* and *Vector\_vle\_core.s.* If you do not want a bootloader, unpack Non\_Bootloader\_Settings.zip.

## Table 7. Source files

Files	Descriptions
Files with pdc prefix	contain codes for PDC applications.
Files with bcc prefix	contain drivers of MC3377x, include MC33772, MC33771B and MC33771C.
Files with sbc prefix	contain drivers of MC33FS65xx.
Files in boot folder	contain codes for bootloader.
Files with i2c prefix	contain codes for I2C protocol.
Files with rtc prefix	contain codes for PCA85063 driver.
Files with msdi prefix	contain drivers of CD1020, CD1030, MC33978 and MC34978.
Files with nxp prefix	contain nxp_helpers codes used in SBC drivers.

## 3.2.3 Downloading the exist elf file

For downloading the exist elf file using TRACE32 follow these stps.

1. Launch the TRACE32. Select File > Run Script..., from the menu bar.

#### Software User Guide

A TRACE32 PowerView for PowerPC #1	- 1	×
The Fold View Van Break Run CPU Misc Trace Perf Cov MPCSXXX Window Heln		
dt Run Scrint		
E fait Script		
Search for Script		
Automatic Scripts on Start		
Br Owen Elle		
B load Ele		
The Tune File.		
31 Dump File		
a second		
Pinter Settings		
C Window Screenshot to File		
X Edit		
EB:B::CD.DO C:\Users\NXF53225\Desktop\bitbucket\vds-powertrain-domain-controller\MCU\PDC_5775B_MCAL\run.cmm (C:\Users\NXF53225\Desktop\bitbucket\vds-powertrain-domain-controller\MCU\PDC_5	775B_MCAL)	
EB::B::CD.DO "C:\Users\NXF53225\Desktop\Project\PDC\Version_MCAL\MPC5775B_MCAL_35Code alignment(include test format)\run.cmm" (C:\Users\NXF53225\Desktop\Project\PDC\Version_MCAL\MPC5775B_	MCAL_35Code a	ilignment(
SV:00818340 7A12FFFC e_beq 0x81833C		
SY:00818344 (301 Se_102 r0,0X0C(r1) ; r0,12(r1) SY:00818346 0090 se_mtlr r0		
\$V:00818348 C036 se_lwz r3,0x0(r6) ; r3,0(r6)		
SV:0081834A 20/1 Se_a01 r1,0x8 ; r1,8 SV:0081834C 0004 Se_b1r		
SV:0081834E 0000 se_illegal		
SY:00818350 [243 LLN_HW_L.Se_CHD11 75,0X5 ; 75,5 SY:00818352 [7401000 e ble 0x81835C		
i. <		
B::/B::Data.Load.Elf ./Debug/PDC_57758_SDK_Z7_0.elf /Quad /verify		
components trace Data Var List PERF SYStem Step Go Break sYmbol Frame Register FPU o	ther prev	ious
running	MIX	UP
	17:48	
	央 2019/12/10	1
Figure 36. Run script		

2. Select the script named run.cmm (in root directory) to download elf. file into flash.

rganize 🗙 New folder					8== 👻 🗂
= Dicturar		Data med <sup>26</sup> - 1	Tune	Size	
Pictures	A Name	Date modified	Туре	Size	
Desktop	.metadata	2019/12/9 9:29	File folder		
OBC	bin	2019/12/10 16:24	File folder		
Plan	cfg	2019/12/9 9:29	File folder		
ubuntu	fix	2019/12/9 9:29	File folder		
	include	2019/12/10 10:36	File folder		
CHECKIC HAI	make	2019/12/9 9:29	File folder		
OneDrive - Personal	Si	2019/12/10 14:09	File folder		
Documents	src	2019/12/10 16:24	File folder		
图片	toolchains	2019/12/9 9:29	File folder		
应用	I Iresos	2019/12/9 9:29	File folder	12 KB	
- N// ( 0000	run.cmm	2019/10/11 14:04	CIVINI FILE	12 ND	
NXL60822					
3D Objects					
E Desktop					
😫 Documents					
🕹 Downloads					
h Music					
E Pictures					
Videos					
SDisk (C:)					
	~				
A Makeral.					
File name: run.cmm					PRACTICE script (*.cmm)
					Open Cancel
				1 sunning	
= 🖛 🧟 👝 🏹					
- HI 🗖 🎇 🌔 🧲		N 🌏 🥃 🖊 🖆		x ^	、

3. Click Go button to run.

#### Software User Guide



#### 3.2.4 How to build

1. Launch the EB tresos Studio and select File > Import, from the IDE menu. Choose import source page appears. Click Browse and select the project folder \Tresos\workspace, click Finish.

88 Import			—		×
Import Projects				T	-
Select a directory to se	arch for existing Ec	lipse projects.			
Select root directory:	C:\Users\NXF532	25\Desktop\Project\	PDC\Version_MCAL\N	Browse	
O Select archive file:				Browse	
Projects:					
PDC_5775B_MCA	L (C:\Users\NXF53	225\Desktop\Project	\PDC\Version_MCAL\N	Select All	
				Deselect All	
				Refresh	
<			>		
Copy projects into v	vorkspace				
Working sets					
Add project to wor	king sets			<b>C</b> 1 .	
Working sets:			~	Select	
			ri d	Canaal	
(?)	< Back	Next >	Finish	Cancel	

2. Double click the project name PDC\_5775B\_MCAL and click generate button.

	EB tresos 20.1.0 - workspace: C:\Users\NXF File Edit Search Project Window Help ☐ ▼ □ □ ◎ ▼ □	53225\works
	<ul> <li>Project Explorer 3</li> <li>PDC_5775B_MCAL</li> <li>Ecuc (PA, MPC5777C)</li> <li>config</li> <li>output</li> </ul>	
Figure 40. Generate code		

- 3. Fill in the right compiler in compiler\_used.s.
- 4. Fill in user-defined paths for compilation in launch.bat. Paths includes TRESOS\_DIR, MAKE\_DIR, GHS\_DIR or DIAB\_DIR or IAR\_DIR, PLUGINS\_DIR.

```
@echo off
::You have to uncomment and set the following variables if they are not already set
:: uncomment line below if you do not set TRESOS DIR over environment
::TRESOS
SET TRESOS DIR=C:\EB20\tresos
::MAKE
SET MAKE DIR=C:\NXP\S32DS Power v2017.R1\utils\msys32\usr
::GHS
SET GHS DIR=C:\ghs\comp 201854
::DIAB
::SET DIAB DIR=C:/Tools/WindRiver/diab/5.9.4.7
::IAR
::SET IAR DIR=C:/Tools/IAR Systems/Embedded Workbench 7.3/arm/
::Path to the plugins folder
SET PLUGINS DIR=C:\NXP\AUTOSAR\MPC5777C MCAL4 0 RTM HF4 1 0 1\eclipse\plugins
::SSC
::SET SSC ROOT=
Figure 41. Lauch.bat file
```

5. Run build.bat in command window. Errors, warnings and compilation information can be check here.

C:\Windows\System32\cmd exe			_	Π	×
malace Nathing to he days for 'huil	1'				
make: Nothing to be done for built Print Size: DDC 5775B MCAL olf	α.				
section	dec	hex			
. ramcode	60	0x003c			
. data	4622	0x120e			
.mcal data	506	0x01fa			
.bss	21404	0x539c			
.mcal_bss	3584	0x0e00			
.acfls_code_ram	96	0x0060			
.mcal_bss_no_cacheable	2304	0x0900			
.rcw	8	0x0008			
.vletext	90856	0x162e8			
.mcal_text	66566	0x10406			
. rodata	1091	0x0443			
. ROM. ramcode	60	0x003c			
. ROM. data	4622	0x120e			
. KOM. mcal_data	506				
.mcal_const_cig	9268	0x2434			
. mcal_const	1430	0x0059C			
.aclis_code_rom	15000	0x2000			
isrweetbl core	19000	0x0194			
Total · 2	292	0x0124 0x26/b1			
Create S19 file: PDC 5775B MCAL s1	9	0,50101			
Create bin file: PDC 5775B MCAL bi					
gmemfile: warning: hole of size 0x	 3h5628	8 hetween			
.acfls code rom ending at	0x82a	d8 and			
.isrvectbl starting at Oxb	e0000.				
C:\Users\NXF53225\Desktop\bitbucke	et\vds-	-powertrain-domain-controller\MCU\PDC_5775B_MCAL>			$\sim$
Figure 42. Build result					

6. You can run clear.bat to clear output objects in unexpected situations and build again.

**NOTE** Recommended optimization option for this demo is -Osize if Green Hills compiler is used.

#### 3.2.5 Basic workflow of memo project

MCU Init is executed first. It initializes system clock, external interrupts, ports and UART.



BSP task takes four responsibilities:

- 1. BSP Task initializes the remaining peripherals after MCU Init, including DSPI used for initializing SBC watchdog.
- 2. BSP Task initializes SBC watchdog which needs to be initialized within at most 256ms from power on. This is why SBC Task shall be executed before starting task scheduler.
- 3. BSP Task refreshes SBC watchdog regularly after initialization.
- 4. BSP Task contains test functions which shall be executed once or regularly.

Task Scheduler calls three tasks cyclically. The calling periods are 10ms, 50ms and 300ms respectively for VCU Task, BMS Task and BSP Task. UDS Task runs freely instead of scheduled by Task Scheduler because UDS has strict timing requirements and there is no OS to guarantee the schedule can meet UDS timing requirements. Once having received commands on CAN bus, UDS Task will execute UDS protocol and jump to bootloader.

VCU Task checks gear position, accelerator & brake pedal position and VCU faults. After considering BMS faults and VCU faults, VCU Task controls the output, that is, reduces output torque and shuts done accessories by LSD.

BMS Task monitors stack voltage, cell voltage, temperature and BMS fault information. Support daisy chain of MC33771C or MC33771B or MC33772.

#### NOTE

- 1. There is no OS implemented in MCAL project to schedule tasks, one task will only be executed after the previous one is finished.
- 2. There is a limitation in ETH module. You need to install ETH\_RxCbk callback into plugin codes to enable ETH receive function.
  - a. Copy the following code into EthIf\_RxIndication ETH\_RxCbk(CtrIldx, FrameType, IsBroadcast, PhysAddrPtr, DataPtr, LenByte)
  - a. Include pdc\_eth.h in EthIf\_Cbk.c.

# Chapter 4 GUI user guide

This GUI tool is used to monitor system via CAN bus, which can display the system state real time and send reset command to board. It need to use PCAN-USB Pro adapter to connect board with PC. Hands-on instructions are listed below.



- 1. Select CAN port and Baud rate (500 Kbps for this demo) of CAN transmission and then click Start button. Click Stop button when you want to disconnect the CAN adapter.
- 2. Battery voltage will be displayed, including stack voltage and cell voltage(average, maximal and minimal). Temperature(average, maximal and minimal) is also displayed. This GUI supports up to 10 battery stacks.
- 3. NTC parameters can be changed according to populated NTC.
- Clicking Global Reset button resets BMS and VCU task together. While BMS Reset and VCU Reset button reset BMS and VCU task separately.
- 5. Total voltage, current (not measured in this demo) and fault information is displayed.
- 6. This indicator becomes red when faults occur otherwise becomes green.
- 7. Output torque level (0, Low, Mid and High) is shown.
- 8. This indicator appears when handbrake is pulled up.
- 9. Gear position (R, N, and D) is shown. E will appear when the gear shift fails .
- 10. Positions of accelerate pedal and brake pedal is shown. If measured values are out of range, an error image will appear.
- 11. Basic system information is shown on status bar.

All BCC data are displayed on the BCCDATA page (the following figure), where the first row indicates the status of the CID, including OK, timeout, CRC, etc. The last four rows indicates BCC communication errors and BCC faults.

#### GUI user guide

IVBMSVCU	BCCDATA																				
	CID-1	CID-2	CID-3	CID-4	CID-5	CID-6	CID-7	CID-8	CID-9	CID-10	CID-11	CID-12	CID-13	CID-14	CID-15	CID-16	CID-17	CID-18	CID-19	CID-20	
tatus	Ok	Ok	Ok	Ok																	
rror Num	0	0	0	0																	
itack(V)	15.581	15.574	15.486	15.522																	
cell-1(V)	1.133	1.132	1.126	1.128																	
Cell-2(V)	1.133	1.132	1.129	1.127																	
Cell-3(V)	1.131	1.131	1.126	1.129																	
Cell-4(V)	1.131	1.133	1.125	1.127																	
Cell-5(V)	1.131	1.131	1.127	1.127																	
Cell-6(V)	1.131	1.133	1.128	1.125																	
Cell-7(V)	1.133	1.132	1.128	1.127																	
Cell-8(V)	1.131	1.132	1.127	1.128																	
Cell-9(V)	1.131	1.132	1.127	1.126																	
ell-10(V)	1.131	1.130	1.127	1.127																	
Cell-11(V)	1.131	1.131	1.126	1.126																	
Cell-12(V)	1.132	1.130	1.125	1.125																	
Cell-13(V)	1.129	1.131	1.124	1.125																	
Cell-14(V)	1.129	1.129	1.123	1.124																	
Current(A)																					
Chip(C)	21	22	22	21																	
N0(C)	21	22	21	22																	
AN1(C)	22	21	21	22																	
N2(C)	21	21	22	22																	
AN3(C)	21	21	21	22																	
AN4(C)	21	22	22	22																	
AN5(C)	22	21	22	22																	
ANG(C)	22	21	22	22																	
ADC1A BG(V)	1,180	1,180	1,179	1,180																	
ADC1B BG(V)	1,180	1,181	1.180	1,179																	
	0.0000	0.0000	0.0000	0.0000																	
CANL Connecto		MOLLS	M M						CC Interde	TDI							DCC Turn	. MC2277	10		
CAN: Connècte	u	IVICU S	vv version	: 33205 SI	UK			В	CC Interfa	ce: TPL							всс туре	: MC3377	IC .		

# Chapter 5 Bootloader User Guide

A bootloader is provided to update application firmware via CAN bus in case there are no tools to download application directly. It uses PCAN-USB Pro adapter to connect board with PC, executes its function by OpenBus tool. Hands-on instructions are listed below. Refer to the documentation "Unified bootloader user guide" to understand more details about bootloader source code.

- 1. Download bootloader into MCU: Download MPC577XX\_CAN\_bootloader.elf into flash by PE in S32DS or Lauterbach with run.cmm.
- Allocate flash memory of the application. Unpack Bootloader\_Settings.zip to replace original files, then compile the APP project.
- 3. Launch the OpenBus

🖲 TOOLS:			
	ISO-CAN-UDS-TOOL	ISO-LIN-UDS-TOOL	General Tools
	ENTER	ENTER	ENTER
	Uart Log Helper	SJA1105Q-TOOL	ISO-ETH-UDS-TOOL
	ENTER	ENTER	ENTER
			)

- 4. Calculate CRC for the binary file to be transferred:
  - Click Enter button on General Tools block.
  - · Click Upload Firmware button and select the binary file to be transferred.
  - Select CRC-16(DNP) algorithm.
  - · Click Calculate button to generate CRC value.

··· OpenBus	- 6 X
	$\mathbf{x}$
Caculate CRC Tools:	
Upload Firmware	
PDC_5775B_MCALbin Result	×
CRC-16 Seed CRC-1 <sup>0x62,0xe4</sup>	
	Close
	©2018-2019 NXP Semiconductors.
Figure 47. CRC-16(DNP) example	

- 5. Edit configuration file can\_uds.json in bootloader\_bin folder:
  - Fill in the CRC value calculated in Step 4.



• Fill in the correct paths of MPC577XX\_flash\_drv.bin and app.bin (here PDC\_5775B\_MCAL.bin).



Figure 49. Path of MPC577XX\_flash\_drv.bin example



Figure 50. Path of PDC\_5775B\_MCAL.bin example

- 6. Connect PCAN-USB pro:
  - Click Enter button on ISO-CAN-UDS-TOOL block.
  - Click Connect Device on the right-top.
  - Select and fill values as shown below and then click Connect button. Note that target address and source address are fixed as 0x55 and 0x35.

	ISO-CAN-UDS-TOOL		ISO-LIN-UDS-TOOL		General Tools	
	Uart Log Helper		SJA1105Q-TOOL		ISO-ETH-UDS-TOOL	
NP OpenBus					©2018-2019 NXP Sem	ucondu ×
NP	OpenBus					
ISO-CAN-UDS-TOOL				2	Connect Device	×
Load From Config File						
Choose Service						
Schedule Service List:	F	Export Clear Erro	or Delay(ms): 0	Cycle(>0): 1	Excute Schedule Service	
Schedule Service List:		Export Clear Erro	or Delay(ms): 0	Cycle(>0): 1	Excute Schedule Service	ours.
•• OpenBus	)penBus	Export Clear Erro	n Delay(ms): 0	Cycle(>0): 1	Excute Schedule Service	ours. ×
CopenBus     ISO-CAN-UDS-TOOL	)penBus	Export Clear Erro	or Delay(ms): 0	Cycle(>0): 1	Excute Schedule Service C2018-2019 NXP Semiconduct  -  Connect Device	ors. ×
Schedule Service List:     Schedule Service List:     ISO-CAN-UDS-TOOL     Load From Config File	)penBus	ect	Delay(ms): 0	Cycle(>0): 1	Connect Device	ors. ×
CopenBus      Schedule Service List:      Schedule Service List:      Schedule Service List:      Schedule Service      Choose Service      C	)penBus 	ect ce: US d: 50	Delay(ms): 0	Cycle(>0): 1	C2018-2019 NXP Semiconduct Connect Device	ors. ×
OpenBus     ISO-C AN-UDS-TOOL     Lead From Config File     Choose Service     Schedule Service List:	)penBus • Devia	eet ce: US d: 55 al Address: 55	X           SBBUS1           V           1           35	Cycle(>0): 1	C2018-2019 NXP Semiconduct - 0 Connect Device	
Coonsus	D <b>penBus</b> Com • Devid • Spee Physice	ect ce: US d: 50 al Address: 17	or       Delay(ms):       0         SBBUS1       ~         0 kBlt/sec       ~         35       RA         A       SA       RA	Cycle(>0): 1 Cycle(>0): 1	Excute Schedule Service C2018-2019 NXP Semiconduct Connect Device Excute Schedule Ser	Vice
Choose Service List: Choose Service Schedule Service List:	DipenBus Puesto Puesto Addres	ect ce: US d: 50 al Address: TA ss Protocol: IS	x     0       SBBUS1     ×       0 kBit/sec     ×       i     35     RA       i     SA     RA       0_15765_2_29B     ×	Cycle(>0): 1 Cycle(>0): 1	Connect Device	vice
OpenBus     ISO-CAN-UDS-TOOL     Lead From Config File     Choose Service     Schedule Service List:	D <b>penBus</b> Devi Devi Spee Physice Addres	ect ce: US d: 50 al Address: 77 ss Protocol: 15	X           SBBUS1           X           3BBUS1           X           3C           RA           SA           RA           CONNECT	Cycle(>0): 1 Cycle(>0): 1	C2018-2019 NXP Semiconduct -  Connect Device Excute Schedule Ser	Vice

MPC5775B BMS Plus VCU Reference Design User Guide, Rev. 0, February 2020

7. Load config files which links the target binary files. Click Load From Config File button and select can\_uds.json you just modified in step 5.

- Cyerba	- o x
NCP OpenBus	
ISO-CAN-UDS-TOOL	Disconnect
Load Fran Contry File Choose Service	
Schedule Service List:	Zurat Courther Delayary 0. Cycle/0. 1. Zurate Schwarz Schwarz
	C2013-2019 NNP-Semicoadorters
Figure 52. Load from config file	

- 8. Execute bootloader procedure:
  - Click Execute Schedule Service button to execute bootloader procedure.
  - Wait the process to succeed.
  - You can repeat Step 4, 5 and 8 to download apps through bootloader again.

<image/>	₩ OpenBus					- ō ×
Loc from Coning File   Choose Service     Schedule Service List:     Export   Clear Error   Delay(ms):   0   Cycle(>0):   1   Export     Clear Error     Delay(ms):     0   Cycle(>0):     Choose Service     Choose Service <th>NP OpenB</th> <th>IS</th> <th></th> <th></th> <th></th> <th></th>	NP OpenB	IS				
Load From Config File Choose Service Schedule Service List: Export Clear Error Delay(ms): 0 Cycle(>0): 1 Excute Schedule S	ISO-CAN-UDS-TOOL					Disconnect
Choose Service       Image: Choose Service         Schedule Service List:       Image: Choose Service         Export       Clear Error       Delay(ms):       0       Cycle(>0):       1       Excute Schedule Service Service						
Schedule Service List: Export Clear Error Delay(ms): 0 Cycle(>0): 1 Excute Schedule S Construction Service List: Clear Error Delay(ms): 0 Cycle(>0): 1 Cycle(>	Choose Service					
Export       Clear Error       Delay(ms):       0       Cycle(>0):       1       Excute Schedule S         Comparison       Comparison       Comparison       Comparison       Comparison       Comparison         Comparison       Comparison       Comparison       Comparison       Comparison       Comparison         Comparison       Comparison       Comparison       Comparison       Comparison       Comparison         Comparison       Com	Schedule Service List:					Ŷ
©2018-2019 NXP Semico		Export	Clear Error	Delay(ms): 0	Cycle(>0): 1	Excute Schedule Service
©2018-2019 NXP Semico						
©2018-2019 NXP Semico						
©2018-2019 NXP Semico						
©2018-2019 NXP Semico						
CONSTRUCTION AND CONSTRUCTION OF A DECEMBER OF A DECEMBER OF A DECEMBER OFOADOFOA DECE						©2018-2019 NYP Semiconductors
ure 52. Table preparties window	ruro 52. Toble properties winds	.,				©2010-2017 NAL Semiconductors.

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/ SalesTermsandConditions.

While NXP has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer's applications and products, and NXP accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, I2C BUS, ICODE, JCOP, LIFE VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, AltiVec, C-5, CodeTEST, CodeWarrior, ColdFire, ColdFire+, C-Ware, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorlQ, QorlQ Qonverge, Ready Play, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, SMARTMOS, Tower, TurboLink, UMEMS, EdgeScale, EdgeLock, elQ, and Immersive3D are trademarks of NXP B.V. All other product or service names are the property of their respective owners. AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

© NXP B.V. 2020.

All rights reserved.

For more information, please visit: http://www.nxp.com For sales office addresses, please send an email to: salesaddresses@nxp.com

> Date of release: February 2020 Document identifier: MPC5775BDCRDUG



# **Mouser Electronics**

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

NXP: RDVCU5775EVM