CAN Interface Application Note

PulsON® 440

TIME DOMAIN®
Cummings Research Park
4955 Corporate Drive Suite 101
Huntsville, AL 35805 USA
http://www.timedomain.com
Tel: +1 256.922.9229
+1 888.826.8378
Fax: +1.256.922.0387

320-0326A
September 2015

Copyright

Trademarks

Rights

Introduction

The software interface to Time Domain's PulsON 440 (P440) UWB platforms is defined by an Application Programming Interface (API). There are three such APIs, one for ranging and network applications, one for monostatic radar, and one which can be used either for bistatic radar operation or as an RF propagation tool. These documents are listed below, are summarized in the *P440 Data Sheet – User Guide*, and are available on the Time Domain website.

Ranging and Networks:	320-0313x *RangeNet API Specification*
Monostatic Radar:	320-0298x *MRM API Specification*
Bistatic/propagation:	320-0305x *CAT API Specification*

This software interface can be used, with minor changes, on any of the physical interfaces provided by the P440. These interfaces include Ethernet, USB, Serial, SPI, and CAN. This document provides information on using the API through the CAN interface.

The document *Using the USB and Serial Interfaces*, also available from Time Domain's web site, explains using the API through those interfaces in detail.

The P440 CAN interface uses the J1939 protocol. This application note assumes familiarity with J1939 and the various terms used in the J1939 standard.

This application note also defines CAN-specific API messages which can be used to set the baud rate of the CAN bus and the J1939 node address of the radio.

CAN Message Protocol

CAN and J1939 have a limit of 8 data bytes in the payload. Some RCM API messages fit within this 8 data byte limit, while others do not. RCM messages that are $\leq$8 bytes are transmitted with a Proprietary A PGN of 0xEF00. Messages larger than 8 bytes are transmitted using J1939's transport protocol (specifically the Broadcast Announce Message, or BAM). In BAM, an initial PGN of 0xEC00 defines the number of bytes and packets in the message, and the message itself is transmitted in multiple 0xEB00 PGN packets.

For example, to retrieve the status and version information from a radio, a connected host sends an RCM_GET_STATUSINFO_REQUEST message and the radio responds with a RCM_GET_STAUSINFO_RESPONSE message. The request message is 4 bytes long, so it can be sent within a PGN 0xEF00 message. The response, however, is 64 bytes long and must be sent using BAM. A sample RCM_GET_STATUSINFO_REQUEST follows in hex, along with an explanation of the contents:

18EF8001F0010001
18 - default priority of 6
EF80 - PGN EF00 with destination address of 80
01 - source address of 01
F001 - RCM_GET_STATUSINFO_REQUEST message number
0001 - RCM API message ID

And the radio's response is shown below in hex:

10ECFF802040000AFF01EF00
10EBFF8001F1010001020A00
10EBFF800200020500A30C15
10EBFF800308185A00027441
10EBFF80040004000000009C
10EBFF80053135303731352D
10EBFF800672633239000000
10EBFF800700000000000000
10EBFF800800000000000000
10EBFF800900000000000000
10EBFF800A00FFFFFFFFFFFF

Here is an explanation of each message:

10ECFF802040000AFF01EF00
10 - priority of 4
ECFF - PGN EC00 with broadcast destination address
80 - source address of radio
20 - control byte 20, Broadcast Announce Message
4000 - total data size, LSB first (64 decimal bytes)
0A - total number of packets
FF - reserved, always FF
01EF00 - PGN, LSB first. EF01 is PGN

10EBFF8001F1010001020A00
10 - priority of 4
EBFF - PGN EB00 with broadcast destination address
80 - source address
01 - packet number 1
F101 - RCM_GET_STATUSINFO_CONFIRM message number
0001 - RCM API message ID
02 - RCM Version Major
0A - RCM Version Minor
00 - MSB of RCM Version Build

10EBFF800200020500A30C15
10 - priority of 4
EBFF - PGN EB00 with broadcast destination address
80 - source address
02 - packet number 2
00 - LSB of RCM Version Build (complete version is 2.10.0)
02 - UWB Kernel Major
05 - UWB Kernel Minor
00A3 - UWB Kernel Build (complete version is 2.5.163)
0C - FPGA Firmware Version
15 - FPGA Firmware Year

10EBFF800308185A00027441
10 - priority of 4
EBFF - PGN EB00 with broadcast destination address
80 - source address
03 - packet number 3
08 - FPGA Firmware Month
18 - FPGA Firmware Day (complete version is 1508180C)
5A000274 - Serial Number
41 - Board Revision (ASCII 'A')

10EBFF8004000400000000B4
10 - priority of 4
EBFF - PGN EB00 with broadcast destination address
80 - source address
04 - packet number 4
00 - BIT Test result
04 - Board Type (P440)
00 - Transmitter Type (FCC)
000000B4 - Temperature in 0.25 degC (45.00 degC)

10EBFF800531353037313352D
10 - priority of 4
EBFF - PGN EB00 with broadcast destination address
80 - source address
05 - packet number 5
3135303731352D – first 7 characters of package version string

10EBFF800672633239000000
10 - priority of 4
EBFF - PGN EB00 with broadcast destination address
80 - source address
06 - packet number 6
72633239000000– next 7 characters of package version string

10EBFF800700000000000000
10 - priority of 4
EBFF - PGN EB00 with broadcast destination address
80 - source address
07 - packet number 7
00000000000000– next 7 characters of package version string

10EBFF800800000000000000
10 - priority of 4
EBFF - PGN EB00 with broadcast destination address
80 - source address
08 - packet number 8
00000000000000– next 7 characters of package version string

10EBFF800900000000000000
10 - priority of 4
EBFF - PGN EB00 with broadcast destination address
80 - source address
09 - packet number 9
00000000 – last 4 characters of package version string (complete string is 150715-rc29)
000000 – first 3 bytes of status

10EBFF800A00FFFFFFFFFFFF
00 – last byte of status (status is 0)
FFFFFFFFFFFF - unused bytes are FF

CAN API Messages

RCM_SET_CAN_CONFIG_REQUEST (0xF091)

API: Internal
Message type: REQUEST (Host)
Corresponding Message type: RCM_SET_CAN_CONFIG_CONFIRM (Radio)

Purpose: This message configures the CAN parameters in the radio. The CAN configuration is stored in non-volatile memory. The radio must be restarted in order for the new configuration to take effect.

Packet Definition:

#	Parameter	Type	Definition
0	RCM_SET_CAN_CONFIG_REQUEST (0xF091)	UINT16	Message type
1	Message ID	UINT16	Associates request to confirm and info messages

2	CAN Baud Rate	UINT8	0 – 125 Kbps, 1 – 250 Kbps, 2 – 500 Kbps, 3 – 1Mbps. The default baud rate is 250 Kbps.
3	CAN Address	UINT8	The address that the radio will use on the CAN bus. The default address is 128.

RCM_SET_CAN_CONFIG_CONFIRM (0xF191)

API: Internal
Message type: CONFIRM (Radio)
Corresponding Message type: RCM_SET_CAN_CONFIG_REQUEST (Host)

Purpose: This message is sent by the radio to the Host in response to a RCM_SET_CAN_CONFIG_REQUEST message previously received from the host. Its purpose is to confirm successful operation of the RCM_SET_CAN_CONFIG_REQUEST.

Packet Definition:

#	Parameter	Type	Definition
0	RCM_SET_CAN_CONFIG_CONFIRM (0xF191)	UINT16	Message type
1	Message ID	UINT16	Associates request to confirm packets
2	Status	UINT32	0 = successful, non-zero = error

RCM_GET_CAN_CONFIG_REQUEST (0xF092)

API: Internal
Message type: REQUEST (Host)
Corresponding Message type: RCM_GET_CAN_CONFIG_CONFIRM (Radio)

Purpose: This is a request message sent by the Host to the radio to retrieve the current CAN configuration.

Packet Definition:

#	Parameter	Type	Definition
0	RCM_GET_CAN_CONFIG_REQUEST (0xF092)	UINT16	Message type
1	Message ID	UINT16	Associates request to confirm packets

RCM_GET_CAN_CONFIG_CONFIRM (0xF192)

API: Internal
Message type: CONFIRM (Radio)
Corresponding Message type: RCM_GET_CAN_CONFIG_REQUEST (Host)

Purpose: This message is sent by the radio in response to a RCM_GET_CAN_CONFIG_REQUEST from the host. It provides the current CAN configuration information.

Packet Definition:

#	Parameter	Type	Definition
0	RCM_GET_CAN_CONFIG_CONFIRM (0xF192)	UINT16	Message type
1	Message ID	UINT16	Associates request to confirm and info messages
2	CAN Baud Rate	UINT8	0 – 125 Kbps, 1 – 250 Kbps, 2 – 500 Kbps, 3 – 1Mbps. The default baud rate is 250 Kbps.
3	CAN Address	UINT8	The address that the radio will use on the CAN bus. The default address is 128.
4	Reserved	UINT16	
5	Status	UINT32	0 = successful, non-zero = error