
FRCture Documentation

Jessica Creighton

Oct 21, 2020

Contents:

1	Driver Station	3
1.1	Driver Station to RoboRIO	3
1.1.1	UDP	3
1.1.2	TCP	5
1.2	RoboRIO to Driver Station	7
1.2.1	UDP	7
1.2.2	TCP	9
1.3	Driver Station to FMS	12
1.3.1	UDP	12
1.3.2	TCP	14
1.4	FMS to Driver Station	16
1.4.1	UDP	16
1.4.2	TCP	18
1.5	Dashboard	19
1.5.1	Control (UDP port 1164)	19
1.5.2	Status (UDP port 1166)	19
1.5.3	TCP (port 1741)	20
1.5.4	TCP Listening (port 1742)	21
1.6	Logging	21
1.6.1	.dslog	21
1.6.2	.dsevent	23
1.7	Usage Reporting	23
1.7.1	Format	23
1.7.2	Entries	23
1.7.3	Example	26
2	FPGA	29
2.1	Registers	29
2.2	Watchdog	47
2.2.1	Feeding	48
2.2.2	Challenge-response	48

Documenting the results of reversing engineering various proprietary aspects of the [FIRST Robotics Competition](#) system.

This is a *work in progress*. Report any suggestions or corrections on the [issues page](#).

The Driver Station communicates with both the RoboRIO and Field Management System using a set of network protocols.

1.1 Driver Station to RoboRIO

1.1.1 UDP

Sent every 20ms to the RoboRIO, UDP port 1110 (port 1115 when connected to FMS)

Field	Length	Type	Comment
Sequence Num	2	uint16	
Comm Version	1	uint8	0x01 is currently the only observed value
Control	1	<i>Control</i>	
Request	1	<i>Request</i>	
Alliance	1	<i>Alliance</i>	
Tags	n	<i>Tags</i>	

Control

Field	Mask	Comment
E-Stop	x	
FMS Connected	 x . . .	
Enabled	 x . .	
Mode	 x x	0: Teleop, 1: Test, 2: Autonomous

Request

Field	Mask	Comment
Reboot RoboRIO	x...	
Restart Code	x...	

Alliance

Color is Red if `value < 3`, else Blue. Position is `value % 3 + 1`

Tags

Field	Length	Type	Comment
Size	1	uint8	
ID	1	uint8	See table below
Data	n		Depends on tag type

ID	Tag
0x07	<i>Countdown</i>
0x0c	<i>Joystick</i>
0x0f	<i>Date</i>
0x10	<i>Timezone</i>

Countdown 0x07

Field	Length	Type	Comment
Countdown	4	float	In seconds

Joystick 0x0c

Field	Length	Type	Comment
Axis Count	1	uint8	Number of axis bytes to follow
Axes	n	int8	-128 to 127, for each axis
Button Count	1	uint8	Number of buttons
Buttons	n	<i>Button</i>	Bytes containing button data
POV Count	1	uint8	Number of POVs
POVs	2 * n	int16	0 to 360, -1 if not pressed

Button

1 bit per button, so the number of bytes is `count / 8 + ((count == 0) ? 0 : 1)`

Stored **LSB 0**. All bits after the specified number of buttons are 0.

Date 0x0f

This and *Timezone* below are sent once when the DS first connects.

Field	Length	Type	Comment
Microseconds	4	uint32	
Second	1	uint8	
Minute	1	uint8	
Hour	1	uint8	In UTC
Day	1	uint8	
Month	1	uint8	0 is January
Year	1	uint8	Year (starting at 1900)

Timezone 0x10

Field	Length	Type	Comment
Timezone	n	string	Timezone

1.1.2 TCP

Sent over TCP port 1740. Strings are either prefixed with a length (specified) or implied.

Field	Length	Type	Comment
Size	2	uint16	Including ID
ID	1	uint8	See table below (only if Size > 0)
Tags	n	<i>Tags</i>	

Tags

ID	Tag
0x02	<i>Joystick Descriptor</i>
0x07	<i>Match Info</i>
0x0e	<i>Game Data</i>

Joystick Descriptor 0x02

Sends everything on the USB Devices tab of the DS, whether connected or not. Sent when joystick is plugged in, unplugged, or reordered.

Field	Length	Type	Comment
Index	1	uint8	
Is Xbox	1	bool	True if XBox controller
Type	1	int8	<i>See below</i>
Name	1 + n	string	
Axis Count	1	uint8	
Axis Types	n	bytes	<i>See below</i> (one per axis)
Button Count	1	uint8	
POV Count	1	uint8	

Joystick Type

Num	Type	Num	Type
-1	Unknown	8	XInput Drum Kit
0	XInput Unknown	11	XInput Guitar 3
1	XInput Gamepad	19	XInput Arcade Pad
2	XInput Wheel	20	HID Joystick
3	XInput Arcade	21	HID Gamepad
4	XInput Flight Stick	22	HID Driving
5	XInput Dance Pad	23	HID Flight
6	XInput Guitar	24	HID 1st Person
7	XInput Guitar 2		

Axis Type

Num	Type	Num	Type
0	X Axis	3	Twist Axis
1	Y Axis	4	Throttle Axis
2	Z Axis		

Match Info 0x07

Field	Length	Type	Comment
Competition	1 + n	string	
Match Type	1	enum	0:"None", 1:Practice, 2:Qualifications, 3:Eliminations

Game Data 0x0e

Field	Length	Type	Comment
Game Data	n	string	

1.2 RoboRIO to Driver Station

1.2.1 UDP

Sent every 20ms in response to Driver Station packets, UDP port 1150

Field	Length	Type	Comment
Sequence Num	2	uint16	
Comm Version	1	uint8	0x01 is currently the only observed value
Status	1	<i>Status</i>	Basically an echo if everything is operating properly
Trace	1	<i>Trace</i>	
Battery	2	<i>Battery</i>	
Request Date?	1	uint8	0x01 when first connecting, 0x00 otherwise
Tags	n	<i>Tags</i>	

Status

Field	Mask	Comment
E-Stop	x	
Brownout	. . . x . . .	0: Normal, 1: Voltage Brownout
Code Start	 x . .	0: Running, 1: Initializing
Enabled	 x . .	
Mode	 x x	0: Teleop, 1: Test, 2: Autonomous

Trace

Field	Mask	Comment
Robot Code	. . x	
Is RoboRIO	. . . x	
Test Mode	 x . . .	
Autonomous Mode	 x . .	
TeleOp Code	 x .	
Disabled	 x	

Battery

Received as a uint16, XXYY. Voltage is calculated by:

$$XX + YY/256$$

Tags

Field	Length	Type	Comment
Size	1	uint8	
ID	1	uint8	See table below
Data	n		Depends on tag type

ID	Tag
0x01	<i>Joystick Output</i>
0x04	<i>Disk Info</i>
0x05	<i>CPU Info</i>
0x06	<i>RAM Info</i>
0x08	<i>PDP Log</i>
0x09	<i>Unknown</i>
0x0e	<i>CAN Metrics</i>

Joystick Output 0x01

If multiple joysticks, the extended packets are concatenated together. If a joystick isn't outputting anything, the size will be 1 (to include the packet type ID) and not contain any of the below. So if the 3rd joystick is outputting, it'll output *01 01 01 01 09 01 <below structure>*.

Field	Length	Type	Comment
Outputs	4	<i>Output</i>	
Left Rumble	2	uint16	
Right Rumble	2	uint16	

Output

1 bit per output. Stored **LSB 0**.

Disk Info 0x04

Field	Length	Type	Comment
Free Space	4	uint32	Number of bytes available

CPU Info 0x05

Field	Length	Type	Comment
Num of CPUs	4	float	<i>0x02</i> on the RoboRIO
CPU _n Time Critical %	4	float	
CPU _n Above Normal %	4	float	
CPU _n Normal %	4	float	
CPU _n Low %	4	float	

RAM Info 0x06

Field	Length	Type	Comment
Block	4	uint32	
Free Space	4	uint32	Number of bytes available

PDP Log 0x08

Field	Length	Type	Comment
Unknown	1		
PDP Stats	21		
Unknown	3		

The statistics for each PDP port is 10 bits, padding together for a total of 21 bytes.

```

struct PDP {
    uint16_t port_00 : 10; //
    uint16_t port_01 : 10; // 8 bytes, 40 bits
    uint16_t port_02 : 10; //
    uint16_t port_03 : 10; //
    uint16_t port_04 : 10; //
    uint16_t port_05 : 10; //
    uint16_t pad1    : 4; //

    uint16_t port_06 : 10; //
    uint16_t port_07 : 10; // 8 bytes, 40 bits
    uint16_t port_08 : 10; //
    uint16_t port_09 : 10; //
    uint16_t port_10 : 10; //
    uint16_t port_10 : 10; //
    uint16_t pad2    : 4; //

    uint16_t port_12 : 10; //
    uint16_t port_13 : 10; // 5 bytes, 40 bits
    uint16_t port_14 : 10; //
    uint16_t port_15 : 10; //
}

```

Unknown 0x09

Field	Length	Type	Comment
Unknown	9		

CAN Metrics 0x0e

Field	Length	Type	Comment
Utilization %	4	float	
Bus Off	4	uint32	
TX Full	4	uint32	
RX Errors	1	uint8	
TX Errors	1	uint8	

1.2.2 TCP

Sent over TCP port 1740. Strings are either prefixed with a length (specified) or implied.

Field	Length	Type	Comment
Size	2	uint16	Including ID
ID	1	uint8	See table below (only if Size > 0)
Tags	1		Depends on tag type

Tags

ID	Tag
0x00	<i>Radio Events?</i>
0x01	<i>Usage Report</i>
0x04	<i>Disable Faults</i>
0x05	<i>Rail Faults</i>
0x0a	<i>Version Info</i>
0x0b	<i>Error Message</i>
0x0c	<i>Standard Output</i>
0x0d	<i>Unknown</i>

Radio Events? 0x00

Might just be generic "Error Strings" or something.

Field	Length	Type	Comment
Message	n	string	

Usage Report 0x01

Field	Length	Type	Comment
Team Num	2	string	
Unknown	1	uint8	
Entries	n	<i>Entry Data</i>	

Disable Faults 0x04

Field	Length	Type	Comment
Comms	2	uint16	
12V	2	uint16	

Rail Faults 0x05

Field	Length	Type	Comment
6V	2	uint16	
5V	2	uint16	
3.3V	2	uint16	

Version Info 0x0a

Field	Length	Type	Comment
Type	1	<i>Device</i>	See below
Unknown	2		
ID	1	uint8	CAN ID?
Name	1 + n	string	
Version	1 + n	string	

Empty message (00 00 00 00 00 00) signifies end of list.

Device Type

Num	Type
0	Software (roboRIO image, FRC_Lib_Version)
2	CAN Talon
8	PDP
9	PCM

Error Message 0x0b

Shows up in main Driver Station log. Sequence Number increments along with Standard Output.

Field	Length	Type	Comment
Timestamp	4	float	Seconds since start of log
SeqNum	2	uint16	
Unknown	2		Only seen as 01?
Error Code	4	int32	
Flags	1	<i>Flags</i>	
Details	2 + n	string	
Location	2 + n	string	
Call Stack	2 + n	string	

Flags

Field	Mask	Comment
Error	 x	
isLVcode	 x .	

Standard Output 0x0c

Sequence Number increments along with Error Message.

Field	Length	Type	Comment
Timestamp	4	float	Seconds since start of log
SeqNum	2	uint16	
Message	n	string	

Unknown 0x0d

00 00 04 04 04 04

1.3 Driver Station to FMS

1.3.1 UDP

Sent to *10.0.100.5* UDP port 1160 (from UDP port 1145).

Field	Length	Type	Comment
Sequence Num	2	uint16	
Comm Version	1	uint8	Only 0x00 has been observed
Status Byte	1	<i>Status</i>	Contains control bits
Team Num	2	uint16	
Battery	2	<i>Battery</i>	Battery voltage
Tags	N	<i>Tags</i>	

Status

Field	Mask	Comment
E-Stop	x	0: Normal, 1: Emergency Stopped
Robot Comms Active	. . x	0: No, 1: Yes
Radio Ping	. . . x . . .	0: No, 1: Yes
Rio Ping	 x . .	0: No, 1: Yes
Enabled	 x .	0: Disabled, 1: Enabled
Mode	 xx	0: TeleOp, 1: Test, 2: Autonomous
Reserved	. x	

Battery

Received as a uint16, XXYY. Voltage is calculated by: $XX + YY/256$

Tags

Field	Length	Type	Comment
Size	1	uint8	
ID	1	uint8	See table below
Data	n		Depends on tag type

ID	Packet Type
0x00	<i>Field Radio Metrics</i>
0x01	<i>Comms Metrics</i>
0x02	<i>Laptop Metrics</i>
0x03	<i>Robot Radio Metrics</i>
0x04	<i>PD Info</i>

Field Radio Metrics (0x00)

Field	Length	Type	Comment
Signal Strength	1	uint8	
Bandwidth Utilization	2	uint16	

Comms Metrics (0x01)

Field	Length	Type	Comment
Lost Packets	2	uint16	
Sent Packets	2	uint16	
Average Trip Time	1	uint8	

Laptop Metrics (0x02)

Field	Length	Type	Comment
Battery Percent	1	uint8	
CPU Percent	1	uint8	

Robot Radio Metrics (0x03)

Decompiling the Offseason FMS code shows “Robot Radio Metrics” in the enum, and a “UDPTagBridgeWifiMetrics”. Neither of which are used or handled, but I assume they are the same thing.

Field	Length	Type	Comment
Signal Strength	1	uint8	
Bandwidth Utilization	2	uint16	

PD Info (0x04)

Seems to be empty for now

Field	Length	Type	Comment

1.3.2 TCP

Sent to *10.0.100.5* TCP port 1750.

Field	Length	Type	Comment
Size	2	uint16	Including ID
ID	1	uint8	See table below (only of Size > 0)
Tags	n		Depends on tag type

Tags

ID	Tag
0x00	<i>WPILib Version</i>
0x01	<i>RIO Version</i>
0x02	<i>DS Version</i>
0x03	<i>PDP Version</i>
0x04	<i>PCM Version</i>
0x05	<i>CANJag Version</i>
0x06	<i>CANTalon Version</i>
0x07	<i>Third Party Device Version</i>
0x15	<i>Usage Report</i>
0x16	<i>Log Data</i>
0x17	<i>Error and Event Data</i>
0x18	<i>Team Number</i>
0x1b	<i>Challenge Response</i>
0x1c	<i>DS Ping</i>

Versions all take the same format

Version

See above table for devices and their associated tag IDs.

Field	Length	Type	Comment
Status	n	string	<bad>, <good>, or <preferred>
Version	n	string	

Usage Report (0x15)

Field	Length	Type	Comment
Team Num	2	uint16	
Unknown	1	uint8	
Entries	n	<i>Entry Data</i>	

See *Usage Reporting* for details about the Entries format.

Log Data (0x16)

Field	Length	Type	Comment
Trip Time	1	uint8	This is round trip time, actual trip time is half of this
Lost Packets	1	uint8	
Voltage	2	<i>Battery</i>	XXYY, XX + YY / 256
Robot Status	1	<i>Status</i>	
CAN	1	uint8	Value is halved
SignalDb	1	uint8	Value is halved
Bandwidth	2	Bandwidth	round(float(uint16) / 256, 2)

Status

Field	Mask	Comment
Brownout	x	
Watchdog	. x	
DS TeleOp	. . x	
DS Auto	. . . x . . .	
DS Disable	 x . .	
Robot TeleOp	 x . .	
Robot Auto	 x .	
Robot Disable	 x	

Error and Event Data (0x17)

Field	Length	Type	Comment
Message Count	4	uint32	Number of messages to process
Timestamp	8	uint64	Number of seconds since 1904/01/01 00:00:00 GMT
Unknown	8		86 48 b0 00 00 00 00 00
Log Message	4 + n	string	

Team Number (0x18)

Sent immediately upon TCP connection. Sending this triggers a challenge response.

Field	Length	Type	Comment
Team Num	2	uint16	Looks like it takes over the FMS's Driver Station context associated with that team number. Would probably be bad to send another team's number from your own station

Challenge Response (0x1b)

Used to verify if the Driver Station is "official". Trivial to figure out the correct response from the off-season FMS, which isn't incredibly useful.

Field	Length	Type	Comment
Response	n	string	

As of 2017, there's 23 possible responses. These change year to year.

ID	Response
0	Johnny Five
1	Data
2	ED-209
3	Bishop
4	NXT
5	Optimus Prime
6	Roomba
7	Rosie
8	The Terminator
9	HAL 9000
10	Sojourner
11	R2-D2
12	C-3PO
13	Wall-E
14	Curiosity
15	Opportunity
16	Spirit
17	Cylons
18	Stinky
19	Iron Giant
20	RCX
21	EV3
22	ASIMO

The correct response uses the Random Value sent by the *Challenge Question*.

```
id = ( (team number) * (random value) ) % 23
```

DS Ping (0x1c)

Just an empty tag. Sent by DS to FMS as a heartbeat.

Field	Length	Type	Comment

1.4 FMS to Driver Station

1.4.1 UDP

Sent to the DS every 500ms on UDP port 1121 by the normal FMS, or to UDP port 1120 by the offseason FMS. Sending to 1120 causes the DS to display a prompt asking to approve FMS control, whereas sending to 1121 starts FMS control immediately.

Field	Length	Type	Comment
Sequence Num	2	uint16	
Comm Version	1	uint8	Only 0x00 has been observed
Control Byte	1	<i>Control</i>	Contains control bits
Request Byte	1	Request	Stays at 0x00, not used
Alliance Station	1	<i>Alliance Station</i>	Represents the station the DS connects to
Tournament Level	1	<i>Tournament Level</i>	What level of competition this is
Match Number	2	uint16	Represents the number of current match
Play Number	1	uint8	Increments if there's a replay
Date	10	<i>Date</i>	
Remaining Time	2	uint16	Time left in current mode
Tags	n	Tags	Offseason FMS has code for sending tags but none to send

Control

Field	Mask	Comment
E-Stop	x	0: Normal, 1: Emergency Stopped
Enabled	 x . .	0: Disabled, 1: Enabled
Mode	 x x	0: TeloOp, 1: Test, 2: Autonomous

Alliance Station

Computed by taking station [1, 3] and subtracting 1 for red, or adding 2 for blue. Color is Red if `value < 3`, else Blue. Position is `value % 3 + 1`

Tournament Level

Level	Value	Comment
Match Test	0	Only used during FMS setup for testing purposes
Practice	1	
Qualification	2	
Playoff	3	

Date

Field	Length	Type	Comment
Microseconds	4	uint32	
Second	1	uint8	
Minute	1	uint8	
Hour	1	uint8	In UTC?
Day	1	uint8	
Month	1	uint8	0 is January
Year	1	uint8	Year (starting at 1900)

1.4.2 TCP

10.0.100.5, listening on TCP port 1750.

Field	Length	Type	Comment
Size	2	uint16	Including ID
ID	1	uint8	See table below (only of Size > 0)
Tags	n		Depends on tag type

Tags

ID	Tag
0x00	<i>WPILib Version</i>
0x01	<i>RIO Version</i>
0x02	<i>DS Version</i>
0x03	<i>PDP Version</i>
0x04	<i>PCM Version</i>
0x05	<i>CANJag Version</i>
0x06	<i>CANTalon Version</i>
0x07	<i>Third Party Device Version</i>
0x14	<i>Event Code</i>
0x19	<i>Station Info</i>
0x1a	<i>Challenge Question</i>
0x1c	<i>Game Data</i>

Versions all take the same format

Version

See above table for devices and their associated tag IDs. Seems only the “preferred” DS version is sent to the Driver Station.

Field	Length	Type	Comment
Status	n	string	<bad>, <good>, or <preferred>
Version	n	string	

Event Code (0x14)

Field	Length	Type	Comment
Event Code	1 + n	string	Event name

Station Info (0x19)

Field	Length	Type	Comment
Alliance Station	1	<i>Alliance Station</i>	
Station Status	1	<i>Station Status</i>	

Station Status

Num	Status	Comment
0	Good	Sent if FMS Driver Station context is valid. Causes the DS to display “FMS Connected”
1	Bad	Sent if the Driver Station is plugged into the wrong station. Causes the DS to display “Wrong station. Go to X”
2	Waiting	Sent if FMS Driver Station context is invalid. Causes the DS to display “FMS Waiting”

Challenge Question (0x1a)

Field	Length	Type	Comment
Question	1 + n	string	<i>Coolest robot EVER!??</i>
Random Value	2	uint16	Random number [1000, 2000] associated with the FMS Driver Station context

Offseason FMS allows for sending multiple questions, but the responses are only checked against the first one (as of 2018).

See the *Challenge Response* for possible responses.

Game Data (0x1c)

Field	Length	Type	Comment
Data	1 + n	string	

1.5 Dashboard

TODO: Based on LabVIEW Dashboard code. Need to actually listen and see if they are exact copies

The Driver Station uses a variety of ports to communicate with any Dashboard software. Either connects to these ports localhost or to them on the specified Remote Dashboard IP.

1.5.1 Control (UDP port 1164)

Simply a copy of the packets sent to the RoboRIO

1.5.2 Status (UDP port 1166)

Simply a copy of the packets sent from the RoboRIO

1.5.3 TCP (port 1741)

Might have the other tags, but at least have the ‘Robot IP’ and ‘DB Mode’ tags special

Field	Length	Type	Comment
Size	2	uint16	Including ID
ID	1	uint8	See table below (only of Size > 0)
Tags	n		Depends on tag type

ID	Tag
0x00	<i>Error Messages</i>
0x04	<i>Disable Counts</i>
0x05	<i>Rail Faults</i>
0x07	<i>Match Info</i>
0x08	<i>Robot IP</i>
0x09	<i>DB Mode</i>

Error Messages (0x00)

Field	Length	Type	Comment
Message	n	string	

Disable Counts (0x04)

Field	Length	Type	Comment
Comms	2	uint16	
12V	2	uint16	

Rail Faults (0x05)

Field	Length	Type	Comment
6V	2	uint16	
5V	2	uint16	
3.3V	2	uint16	

Match Info (0x07)

Field	Length	Type	Comment
Competition	1 + n	string	
Match Type	1		0: “None”, 1: Practice, 2: Qualifications, 3: Eliminations
Match Number	2	uint16	
Replay Number	1	uint8	

Robot IP (0x08)

Field	Length	Type	Comment
IP	4	IPv4 Address	

DB Mode (0x09)

Field	Length	Type	Comment
Mode	1	<i>Mode</i>	

Mode

Field	Mask	Comment
Docked	 x . .	
Simulated	 x	

1.5.4 TCP Listening (port 1742)

The Driver Station listens on TCP port 1742 and sends the following JSON:

```
{ "robotIP":0, "dashboardIP":3232235818, "FMS Connected":false, "Robot Connected":false, "RobotCode":false }
```

This is sent whenever one of the values changes.

1.6 Logging

The Driver Station saves 2 types of log files to disk (in C:\Users\Public\Documents\FRC\Log Files) for later viewing. A .dslog and a .dsevents.

The log populates the list with any file ending with .dslog, but the Driver Station helpfully names them with the current date and time. (e.g. 2018_04_22 23_39_25 Sun.dslog)

1.6.1 .dslog

Stored in Big Endian

Field	Length	Type	Comments
Header	20	<i>Header</i>	
Entries	n	<i>Entry</i>	Array of n entries

Header

Field	Length	Type	Comments
Version	4	uint32	
Unknown	4		
Timestamp	4	uint32	Seconds since 1904/1/1
Fractional Seconds?	1	uint8	Divide by 256 to get decimal part of timestamp?
Unknown	7		

Entry

Field	Length	Type	Comments
Trip Time	1	uint8	$x / 15 = ms$
Packet Loss %	1	uint8	$x * 4$
Battery	2	Battery	$xx + yy/256$
RoboRIO CPU %	1	uint8	$x / 2$
Trace	1	<i>Trace</i>	
CAN Percent	1	uint8	$x / 2$
Wifi dB	1	uint8	$x / 2$
Wifi Mb	1	uint8	
Unknown	2		
PDP Stats	21	<i>PDP</i>	
Unknown	3		

Trace

Field	Mask	Comment
Brownout	x	
Watchdog	. x	
DS Teleop	. . x	
Unused	. . . x . . .	
DS Disabled	 x . .	
Robot Teleop	 x .	
Robot Auton	 x .	
Robot Disabled	 x	

PDP

Current for each PDP port is stored as 10 bits, (mostly) packed together. 0-5 are packed and padded into the first 64 bits, 6-11 are packed and padded into the next 64 bits, 12-15 are packed into the next 40 bits. Each of the 10 bit numbers is the current of the port * 8.

Ports	Bits	Total Bytes
0, 1, 2, 3, 4, 5	60 bits + 4 padding	8 bytes
6, 7, 8, 9, 10, 11	60 bits + 4 bits padding	8 bytes
12, 13, 14, 15	40 bits	5 bytes

1.6.2 .dsevent

TODO

1.7 Usage Reporting

- *Format*
- *Entries*
- *Example*

1.7.1 Format

Not quite sure what happens if the “feature” has a parentheses in it.

BNF

(..ish. Too lazy to make it actually proper)

```

<extended> ::= ">" | ""
<type> ::= <letter>
<instance> ::= <integer>
<context> ::= <integer>
<feature> ::= <text>

<opt-context> ::= "" | ":" <integer>
<opt-feature> ::= "" | "(" <feature> ")"

<entry> ::= <extended> <type> <instance> <opt-context> <opt-feature>
<entry-list> ::= <entry> | <entry> <entry-list>
<report> ::= "V1" <entry-list>

```

PCRE

```

(? 'extended'> ) ? ( ? 'type' [A-Za-z] ) ( ? 'instance' \d+ ) ( ? : : ( ? 'context' \d+ ) ) ? ( ? : \ ( ( ?
'feature' .* ? ) \ ) )

```

1.7.2 Entries

Entries with blank instances are either unused, or reported by an external library

Letter	Resource	Instance	Context	Feature
A	Controller			
B	Module			
C	Language	<i>Language Type</i>		
D	CANPlugin			
E	Accelerometer	Channel		Built-in accelerometer if using
F	ADXL345	1: SPI, 2: I2C		

Table 1 – continued from previous page

Letter	Resource	Instance	Context	Feature
G	AnalogChannel	Channel		
H	AnalogTrigger	Channel		
I	AnalogTriggerOutput	Trigger Index	<i>Trigger Type</i>	
J	CANJaguar			
K	Compressor	PCM ID		
L	Counter	Index	<i>Counter Mode</i>	
M	Dashboard			
N	DigitalInput	Channel		
O	DigitalOutput	Channel		
P	DriverStationCIO			
Q	DriverStationEIO			
R	DriverStationLCD			
S	Encoder	FPGA Index	0: 1X, 1: 2X, 2: 4X	
T	GearTooth	Channel		D if direction sensitive. Not used in C++
U	Gyro	Channel		
V	I2C	Device Address		
W	Framework	<i>Framework Type</i>		
X	Jaguar	Channel		
Y	Joystick	Port		
Z	Kinect			
a	KinectStick			
b	PIDController	Instance num (starting at 1)		
c	Preferences	0		
d	PWM	Channel		
e	Relay	Channel, (+128 if reversable)		
f	RobotDrive	Num Motors	<i>Drive Type</i>	
g	SerialPort	0		
h	Servo	Channel		
i	Solenoid	Channel		
j	SPI	Instance num (starting at 1)		
k	Task			
l	Ultrasonic	Channel		
m	Victor	Channel		
n	Button			
o	Command	1 (kCommandScheduler)		
p	AxisCamera	Handle		
q	PCVideoServer	Handle		
r	SmartDashboard	0		
s	Talon	Channel		
t	HiTechnicColorSensor			
u	HiTechnicAccel			
v	HiTechnicCompass			
w x	SRF08 AnalogOutput	Channel		
y z	VictorSP PWMTalonSRX	Channel Channel		
>A	CANTalonSRX	Device ID + 1		
>B	ADXL362	<i>SPI Port</i>		
>C	ADXRS450	<i>SPI Port</i>		
>D	RevSPARK	Channel		
>E	MindsensorsSD540	Channel		

Table 1 – continued from previous page

Letter	Resource	Instance	Context	Feature
>F	DigitalFilter	Channel		
>G	ADIS16448			
>H	PDP			
>I	PCM			
>J	PigeonIMU	Device ID + 1		
>K	NidecBrushless	Channel		
>L	CANifier	Device ID + 1		
>M	CTRE_future0	Talon device ID + 1		
>N	CTRE_future1	Device ID + 1		
>O	CTRE_future2	Device ID + 1		
>P	CTRE_future3	Device ID + 1		
>Q	CTRE_future4			
>R	CTRE_future5			
>S	CTRE_future6			

Language

ID	Language
1	LabVIEW
2	CPlusPlus
3	Java
4	Python
5	DotNet

Trigger Type

ID	Type
0	InWindow
1	State
2	RisingPulse
3	FallingPulse

Counter Mode

ID	Mode
0	TwoPulse
1	Semiperiod
2	PulseLength
3	ExternalDirection

Framework

ID	Framework
1	Iterative
2	Simple
3	CommandControl

Robot Drive Type

ID	Type
1	ArcadeStandard
2	ArcadeButtonSpin
3	ArcadeRatioCurve
4	Tank
5	MecanumPolar
6	MecanumCartesian

SPI Port

ID	Port
0	OnboardCS0
1	OnboardCS1
2	OnboardCS2
3	OnboardCS3
4	MXP

1.7.3 Example

V1K0N0Y0c0i0q0r0N1W1Y1b1i1j1o1q1C2i2i3A4>A31>O31>A32>O32>A33>O33>A34>O34>A35>O35>A36>O36f2

Entry	Description
V1	Magic String? (Not an I2C at least)
K0	Compressor (PCM 0)
N0	Digital Input (Channel 0)
Y0	Joystick (Port 0)
c0	Preferences
i0	Solenoid (Channel 0)
q0	PCVideoServer (Handle 0)
r0	SmartDashboard
N1	Digital Input (Channel 1)
W1	Framework (Iterative)
Y1	Joystick (Port 1)
b1	PID Controller (#1)
i1	Solenoid (Channel 1)
j1	SPI (#1)
o1	Command

Continued on next page

Table 2 – continued from previous page

Entry	Description
q1	PCVideoServer (Handle 1)
C2	Language (C++)
i2	Solenoid (Channel 2)
i3	Solenoid (Channel 3)
A4	Controller (RoboRIO probably)
>A31	CANTalonSRX (ID 31)
>O31	CTRE_future2 (ID 31)
>A32	CANTalonSRX (ID 32)
>O32	CTRE_future2 (ID 32)
>A33	CANTalonSRX (ID 33)
>O33	CTRE_future2 (ID 33)
>A34	CANTalonSRX (ID 34)
>O34	CTRE_future2 (ID 34)
>A35	CANTalonSRX (ID 35)
>O35	CTRE_future2 (ID 35)
>A36	CANTalonSRX (ID 36)
>O36	CTRE_future2 (ID 36)
f2:1	Robot Drive (2 motors, ArcadeStandard)
S0:2	Encoder (FPGA Index 0, 4X)

CHAPTER 2

FPGA

The RoboRIO has an onboard FPGA which handles much of the hardware, including digital IO, relays, analog inputs, PWM, etc. It's what ultimately handles the safety of the robot, by preventing hardware from being used when disabled.

2.1 Registers

The FPGA has a variety of registers that can be interacted with using the standard NI FPGA library function calls.

The following was generated from the `roboRIO_FPGA_2019_19.0.2.lvbitx` file.

	Register	Offset	Dir	Type/Size	Comments
0	LocalTime	18000	Indicator	U32	
1	Revision	18004	Indicator	U32	
2	Version	1800a	Indicator	U16	
3	LocalTimeUpper	1800c	Indicator	U32	
4	LEDs	18010	Control	Cluster (17)	
4.0	> Comm			U8	
4.1	> Mode			U8	
4.2	> RSL			Boolean	
5	UserButton	18016	Indicator	Boolean	
6	SysWatchdog.Status	18018	Indicator	Cluster (32)	
6.0	> SystemActive			Boolean	
6.1	> PowerAlive			Boolean	
6.2	> SysDisableCount			uint (15)	
6.3	> PowerDisableCount			uint (15)	
7	SysWatchdog.Command	1801c	Control	U32	
8	SysWatchdog.Challenge	18020	Indicator	U32	
9	SysWatchdog.Timer	18024	Indicator	U32	
10	SysWatchdog.Active	1802a	Control	Boolean	
11	SysWatchdog.ForcedKills	1802e	Indicator	uint (15)	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
12	AI.ReadSelect	18032	Control	Cluster (4)	
12.0	> Channel			uint (3)	
12.1	> Averaged			Boolean	
13	AI.LatchOutput	18036	Control	Boolean	
14	AI.Output	18038	Indicator	I32	
15	AI.Config	1803c	Control	Cluster (29)	
15.0	> ScanSize			uint (3)	
15.1	> ConvertRate			uint (26)	
16	AI.ScanList	18040	Control	uint (3) x 8 (24)	
17	AI.OversampleBits	18044	Control	uint (4) x 8 (32)	
18	AI.AverageBits	18048	Control	uint (4) x 8 (32)	
19	AI.LoopTiming	1804c	Indicator	U32	
20	Accumulator0.Center	18050	Control	I32	
21	Accumulator0.Reset	18056	Control	Boolean	
22	Accumulator0.Output	18058	Indicator	Cluster (96)	
22.0	> Value			I64	
22.1	> Count			U32	
23	Accumulator0.Deadband	1805c	Control	I32	
24	Accumulator1.Center	18060	Control	I32	
25	Accumulator1.Reset	18066	Control	Boolean	
26	Accumulator1.Output	18068	Indicator	Cluster (96)	
26.0	> Value			I64	
26.1	> Count			U32	
27	Accumulator1.Deadband	1806c	Control	I32	
28	AnalogTrigger.Output	18070	Indicator	Cluster (4) x 8 (32)	
28	> AnalogTrigger.Output			Cluster (4)	
28.0	> > InHysteresis			Boolean	
28.1	> > OverLimit			Boolean	
28.2	> > Rising			Boolean	
28.3	> > Falling			Boolean	
29	AnalogTrigger0.SourceSelect	18076	Control	Cluster (14)	
29.0	> Channel			uint (3)	
29.1	> Averaged			Boolean	
29.2	> Filter			Boolean	
29.3	> FloatingRollover			Boolean	
29.4	> RolloverLimit			int (8)	
30	AnalogTrigger0.UpperLimit	18078	Control	I32	
31	AnalogTrigger0.LowerLimit	1807c	Control	I32	
32	AnalogTrigger1.SourceSelect	18082	Control	Cluster (14)	
32.0	> Channel			uint (3)	
32.1	> Averaged			Boolean	
32.2	> Filter			Boolean	
32.3	> FloatingRollover			Boolean	
32.4	> RolloverLimit			int (8)	
33	AnalogTrigger1.UpperLimit	18084	Control	I32	
34	AnalogTrigger1.LowerLimit	18088	Control	I32	
35	AnalogTrigger2.SourceSelect	1808e	Control	Cluster (14)	
35.0	> Channel			uint (3)	
35.1	> Averaged			Boolean	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
35.2	> Filter			Boolean	
35.3	> FloatingRollover			Boolean	
35.4	> RolloverLimit			int (8)	
36	AnalogTrigger2.UpperLimit	18090	Control	I32	
37	AnalogTrigger2.LowerLimit	18094	Control	I32	
38	AnalogTrigger3.SourceSelect	1809a	Control	Cluster (14)	
38.0	> Channel			uint (3)	
38.1	> Averaged			Boolean	
38.2	> Filter			Boolean	
38.3	> FloatingRollover			Boolean	
38.4	> RolloverLimit			int (8)	
39	AnalogTrigger3.UpperLimit	1809c	Control	I32	
40	AnalogTrigger3.LowerLimit	180a0	Control	I32	
41	AnalogTrigger4.SourceSelect	180a6	Control	Cluster (14)	
41.0	> Channel			uint (3)	
41.1	> Averaged			Boolean	
41.2	> Filter			Boolean	
41.3	> FloatingRollover			Boolean	
41.4	> RolloverLimit			int (8)	
42	AnalogTrigger4.UpperLimit	180a8	Control	I32	
43	AnalogTrigger4.LowerLimit	180ac	Control	I32	
44	AnalogTrigger5.SourceSelect	180b2	Control	Cluster (14)	
44.0	> Channel			uint (3)	
44.1	> Averaged			Boolean	
44.2	> Filter			Boolean	
44.3	> FloatingRollover			Boolean	
44.4	> RolloverLimit			int (8)	
45	AnalogTrigger6.UpperLimit	180b4	Control	I32	
46	AnalogTrigger6.LowerLimit	180b8	Control	I32	
47	AnalogTrigger6.SourceSelect	180be	Control	Cluster (14)	
47.0	> Channel			uint (3)	
47.1	> Averaged			Boolean	
47.2	> Filter			Boolean	
47.3	> FloatingRollover			Boolean	
47.4	> RolloverLimit			int (8)	
48	AnalogTrigger5.UpperLimit	180c0	Control	I32	
49	AnalogTrigger5.LowerLimit	180c4	Control	I32	
50	AnalogTrigger7.SourceSelect	180ca	Control	Cluster (14)	
50.0	> Channel			uint (3)	
50.1	> Averaged			Boolean	
50.2	> Filter			Boolean	
50.3	> FloatingRollover			Boolean	
50.4	> RolloverLimit			int (8)	
51	AnalogTrigger7.UpperLimit	180cc	Control	I32	
52	AnalogTrigger7.LowerLimit	180d0	Control	I32	
53	PWM.LoopTiming	180d6	Indicator	U16	
54	PWM.CycleStartTimeUpper	180d8	Indicator	U32	
55	PWM.CycleStartTime	180dc	Indicator	U32	
56	PWM.Config	180e0	Control	Cluster (32)	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
56.0	> Period			U16	
56.1	> MinHigh			U16	
57	PWM.PeriodScaleHdr	180e4	Control	uint (2) x 10 (20)	
58	PWM.PeriodScaleMXP	180e8	Control	uint (2) x 10 (20)	
59	PWM.ZeroLatch	180ec	Control	Boolean x 20 (20)	
60	PWM.Hdr0	180f2	Control	uint (12)	
61	PWM.Hdr1	180f6	Control	uint (12)	
62	PWM.Hdr2	180fa	Control	uint (12)	
63	PWM.Hdr3	180fe	Control	uint (12)	
64	PWM.Hdr4	18102	Control	uint (12)	
65	PWM.Hdr5	18106	Control	uint (12)	
66	PWM.Hdr6	1810a	Control	uint (12)	
67	PWM.Hdr7	1810e	Control	uint (12)	
68	PWM.Hdr8	18112	Control	uint (12)	
69	PWM.Hdr9	18116	Control	uint (12)	
70	PWM.MXP0	1811a	Control	uint (12)	
71	PWM.MXP1	1811e	Control	uint (12)	
72	PWM.MXP2	18122	Control	uint (12)	
73	PWM.MXP3	18126	Control	uint (12)	
74	PWM.MXP4	1812a	Control	uint (12)	
75	PWM.MXP5	1812e	Control	uint (12)	
76	PWM.MXP6	18132	Control	uint (12)	
77	PWM.MXP7	18136	Control	uint (12)	
78	PWM.MXP8	1813a	Control	uint (12)	
79	PWM.MXP9	1813e	Control	uint (12)	
80	DIO.OutputEnable	18140	Control	Cluster (32)	
80.0	> Headers			uint (10)	
80.1	> SPIPort			uint (5)	
80.2	> Reserved			uint (1)	
80.3	> MXP			U16	
81	DIO.DO	18144	Control	Cluster (32)	
81.0	> Headers			uint (10)	
81.1	> SPIPort			uint (5)	
81.2	> Reserved			uint (1)	
81.3	> MXP			U16	
82	DIO.DI	18148	Indicator	Cluster (32)	
82.0	> Headers			uint (10)	
82.1	> SPIPort			uint (5)	
82.2	> Reserved			uint (1)	
82.3	> MXP			U16	
83	DIO.FilterSelectHdr	1814c	Control	uint (2) x 16 (32)	
84	DIO.FilterPeriodHdr0	18150	Control	uint (24)	
85	DIO.FilterPeriodHdr1	18154	Control	uint (24)	
86	DIO.FilterPeriodHdr2	18158	Control	uint (24)	
87	DIO.FilterSelectMXP	1815c	Control	uint (2) x 16 (32)	
88	DIO.FilterPeriodMXP0	18160	Control	uint (24)	
89	DIO.FilterPeriodMXP1	18164	Control	uint (24)	
90	DIO.FilterPeriodMXP2	18168	Control	uint (24)	
91	DIO.EnableMXPSpecialFunction	1816e	Control	U16	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
92	DIO.PulseLength	18172	Control	U8	
93	DIO.Pulse	18174	Control	Cluster (32)	
93.0	> Headers			uint (10)	
93.1	> SPIPort			uint (5)	
93.2	> Reserved			uint (1)	
93.3	> MXP			U16	
94	DIO.PWMDutyCycleA	18178	Control	U8 x 4 (32)	
95	DIO.PWMDutyCycleB	1817e	Control	U8 x 2 (16)	
96	DIO.PWMOutputSelect	18180	Control	uint (5) x 6 (30)	
97	DIO.PWMPeriodPower	18186	Control	U16	
98	Counter0.Config	18188	Control	Cluster (32)	
98.0	> UpSource			Cluster (6)	
98.0.0	> > Channel			uint (4)	
98.0.1	> > Module			uint (1)	
98.0.2	> > AnalogTrigger			Boolean	
98.1	> DownSource			Cluster (6)	
98.1.0	> > Channel			uint (4)	
98.1.1	> > Module			uint (1)	
98.1.2	> > AnalogTrigger			Boolean	
98.2	> IndexSource			Cluster (6)	
98.2.0	> > Channel			uint (4)	
98.2.1	> > Module			uint (1)	
98.2.2	> > AnalogTrigger			Boolean	
98.3	> IndexActiveHigh			Boolean	
98.4	> IndexEdgeSensitive			Boolean	
98.5	> UpRisingEdge			Boolean	
98.6	> UpFallingEdge			Boolean	
98.7	> DownRisingEdge			Boolean	
98.8	> DownFallingEdge			Boolean	
98.9	> Mode			uint (2)	
98.10	> PulseLengthThreshold			uint (6)	
99	Counter0.Reset	1818e	Control	Boolean	
100	Counter0.Output	18190	Indicator	Cluster (32)	
100.0	> Direction			Boolean	
100.1	> Value			int (31)	
101	Counter0.TimerConfig	18194	Control	Cluster (32)	
101.0	> StallPeriod			uint (24)	
101.1	> AverageSize			uint (7)	
101.2	> UpdateWhenEmpty			Boolean	
102	Counter0.TimerOutput	18198	Indicator	Cluster (32)	
102.0	> Period			uint (23)	
102.1	> Count			int (8)	
102.2	> Stalled			Boolean	
103	Counter1.Config	1819c	Control	Cluster (32)	
103.0	> UpSource			Cluster (6)	
103.0.0	> > Channel			uint (4)	
103.0.1	> > Module			uint (1)	
103.0.2	> > AnalogTrigger			Boolean	
103.1	> DownSource			Cluster (6)	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
103.1.0	> > Channel			uint (4)	
103.1.1	> > Module			uint (1)	
103.1.2	> > AnalogTrigger			Boolean	
103.2	> IndexSource			Cluster (6)	
103.2.0	> > Channel			uint (4)	
103.2.1	> > Module			uint (1)	
103.2.2	> > AnalogTrigger			Boolean	
103.3	> IndexActiveHigh			Boolean	
103.4	> IndexEdgeSensitive			Boolean	
103.5	> UpRisingEdge			Boolean	
103.6	> UpFallingEdge			Boolean	
103.7	> DownRisingEdge			Boolean	
103.8	> DownFallingEdge			Boolean	
103.9	> Mode			uint (2)	
103.10	> PulseLengthThreshold			uint (6)	
104	Counter1.Reset	181a2	Control	Boolean	
105	Counter1.Output	181a4	Indicator	Cluster (32)	
105.0	> Direction			Boolean	
105.1	> Value			int (31)	
106	Counter1.TimerConfig	181a8	Control	Cluster (32)	
106.0	> StallPeriod			uint (24)	
106.1	> AverageSize			uint (7)	
106.2	> UpdateWhenEmpty			Boolean	
107	Counter1.TimerOutput	181ac	Indicator	Cluster (32)	
107.0	> Period			uint (23)	
107.1	> Count			int (8)	
107.2	> Stalled			Boolean	
108	Counter2.Config	181b0	Control	Cluster (32)	
108.0	> UpSource			Cluster (6)	
108.0.0	> > Channel			uint (4)	
108.0.1	> > Module			uint (1)	
108.0.2	> > AnalogTrigger			Boolean	
108.1	> DownSource			Cluster (6)	
108.1.0	> > Channel			uint (4)	
108.1.1	> > Module			uint (1)	
108.1.2	> > AnalogTrigger			Boolean	
108.2	> IndexSource			Cluster (6)	
108.2.0	> > Channel			uint (4)	
108.2.1	> > Module			uint (1)	
108.2.2	> > AnalogTrigger			Boolean	
108.3	> IndexActiveHigh			Boolean	
108.4	> IndexEdgeSensitive			Boolean	
108.5	> UpRisingEdge			Boolean	
108.6	> UpFallingEdge			Boolean	
108.7	> DownRisingEdge			Boolean	
108.8	> DownFallingEdge			Boolean	
108.9	> Mode			uint (2)	
108.10	> PulseLengthThreshold			uint (6)	
109	Counter2.Reset	181b6	Control	Boolean	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
110	Counter2.Output	181b8	Indicator	Cluster (32)	
110.0	> Direction			Boolean	
110.1	> Value			int (31)	
111	Counter2.TimerConfig	181bc	Control	Cluster (32)	
111.0	> StallPeriod			uint (24)	
111.1	> AverageSize			uint (7)	
111.2	> UpdateWhenEmpty			Boolean	
112	Counter2.TimerOutput	181c0	Indicator	Cluster (32)	
112.0	> Period			uint (23)	
112.1	> Count			int (8)	
112.2	> Stalled			Boolean	
113	Counter3.Config	181c4	Control	Cluster (32)	
113.0	> UpSource			Cluster (6)	
113.0.0	> > Channel			uint (4)	
113.0.1	> > Module			uint (1)	
113.0.2	> > AnalogTrigger			Boolean	
113.1	> DownSource			Cluster (6)	
113.1.0	> > Channel			uint (4)	
113.1.1	> > Module			uint (1)	
113.1.2	> > AnalogTrigger			Boolean	
113.2	> IndexSource			Cluster (6)	
113.2.0	> > Channel			uint (4)	
113.2.1	> > Module			uint (1)	
113.2.2	> > AnalogTrigger			Boolean	
113.3	> IndexActiveHigh			Boolean	
113.4	> IndexEdgeSensitive			Boolean	
113.5	> UpRisingEdge			Boolean	
113.6	> UpFallingEdge			Boolean	
113.7	> DownRisingEdge			Boolean	
113.8	> DownFallingEdge			Boolean	
113.9	> Mode			uint (2)	
113.10	> PulseLengthThreshold			uint (6)	
114	Counter3.Reset	181ca	Control	Boolean	
115	Counter3.Output	181cc	Indicator	Cluster (32)	
115.0	> Direction			Boolean	
115.1	> Value			int (31)	
116	Counter3.TimerConfig	181d0	Control	Cluster (32)	
116.0	> StallPeriod			uint (24)	
116.1	> AverageSize			uint (7)	
116.2	> UpdateWhenEmpty			Boolean	
117	Counter3.TimerOutput	181d4	Indicator	Cluster (32)	
117.0	> Period			uint (23)	
117.1	> Count			int (8)	
117.2	> Stalled			Boolean	
118	Counter4.Config	181d8	Control	Cluster (32)	
118.0	> UpSource			Cluster (6)	
118.0.0	> > Channel			uint (4)	
118.0.1	> > Module			uint (1)	
118.0.2	> > AnalogTrigger			Boolean	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
118.1	> DownSource			Cluster (6)	
118.1.0	> > Channel			uint (4)	
118.1.1	> > Module			uint (1)	
118.1.2	> > AnalogTrigger			Boolean	
118.2	> IndexSource			Cluster (6)	
118.2.0	> > Channel			uint (4)	
118.2.1	> > Module			uint (1)	
118.2.2	> > AnalogTrigger			Boolean	
118.3	> IndexActiveHigh			Boolean	
118.4	> IndexEdgeSensitive			Boolean	
118.5	> UpRisingEdge			Boolean	
118.6	> UpFallingEdge			Boolean	
118.7	> DownRisingEdge			Boolean	
118.8	> DownFallingEdge			Boolean	
118.9	> Mode			uint (2)	
118.10	> PulseLengthThreshold			uint (6)	
119	Counter4.Reset	181de	Control	Boolean	
120	Counter4.Output	181e0	Indicator	Cluster (32)	
120.0	> Direction			Boolean	
120.1	> Value			int (31)	
121	Counter4.TimerConfig	181e4	Control	Cluster (32)	
121.0	> StallPeriod			uint (24)	
121.1	> AverageSize			uint (7)	
121.2	> UpdateWhenEmpty			Boolean	
122	Counter4.TimerOutput	181e8	Indicator	Cluster (32)	
122.0	> Period			uint (23)	
122.1	> Count			int (8)	
122.2	> Stalled			Boolean	
123	Counter5.Config	181ec	Control	Cluster (32)	
123.0	> UpSource			Cluster (6)	
123.0.0	> > Channel			uint (4)	
123.0.1	> > Module			uint (1)	
123.0.2	> > AnalogTrigger			Boolean	
123.1	> DownSource			Cluster (6)	
123.1.0	> > Channel			uint (4)	
123.1.1	> > Module			uint (1)	
123.1.2	> > AnalogTrigger			Boolean	
123.2	> IndexSource			Cluster (6)	
123.2.0	> > Channel			uint (4)	
123.2.1	> > Module			uint (1)	
123.2.2	> > AnalogTrigger			Boolean	
123.3	> IndexActiveHigh			Boolean	
123.4	> IndexEdgeSensitive			Boolean	
123.5	> UpRisingEdge			Boolean	
123.6	> UpFallingEdge			Boolean	
123.7	> DownRisingEdge			Boolean	
123.8	> DownFallingEdge			Boolean	
123.9	> Mode			uint (2)	
123.10	> PulseLengthThreshold			uint (6)	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
124	Counter5.Reset	181f2	Control	Boolean	
125	Counter5.Output	181f4	Indicator	Cluster (32)	
125.0	> Direction			Boolean	
125.1	> Value			int (31)	
126	Counter5.TimerConfig	181f8	Control	Cluster (32)	
126.0	> StallPeriod			uint (24)	
126.1	> AverageSize			uint (7)	
126.2	> UpdateWhenEmpty			Boolean	
127	Counter5.TimerOutput	181fc	Indicator	Cluster (32)	
127.0	> Period			uint (23)	
127.1	> Count			int (8)	
127.2	> Stalled			Boolean	
128	Counter6.Config	18200	Control	Cluster (32)	
128.0	> UpSource			Cluster (6)	
128.0.0	> > Channel			uint (4)	
128.0.1	> > Module			uint (1)	
128.0.2	> > AnalogTrigger			Boolean	
128.1	> DownSource			Cluster (6)	
128.1.0	> > Channel			uint (4)	
128.1.1	> > Module			uint (1)	
128.1.2	> > AnalogTrigger			Boolean	
128.2	> IndexSource			Cluster (6)	
128.2.0	> > Channel			uint (4)	
128.2.1	> > Module			uint (1)	
128.2.2	> > AnalogTrigger			Boolean	
128.3	> IndexActiveHigh			Boolean	
128.4	> IndexEdgeSensitive			Boolean	
128.5	> UpRisingEdge			Boolean	
128.6	> UpFallingEdge			Boolean	
128.7	> DownRisingEdge			Boolean	
128.8	> DownFallingEdge			Boolean	
128.9	> Mode			uint (2)	
128.10	> PulseLengthThreshold			uint (6)	
129	Counter6.Reset	18206	Control	Boolean	
130	Counter6.Output	18208	Indicator	Cluster (32)	
130.0	> Direction			Boolean	
130.1	> Value			int (31)	
131	Counter6.TimerConfig	1820c	Control	Cluster (32)	
131.0	> StallPeriod			uint (24)	
131.1	> AverageSize			uint (7)	
131.2	> UpdateWhenEmpty			Boolean	
132	Counter6.TimerOutput	18210	Indicator	Cluster (32)	
132.0	> Period			uint (23)	
132.1	> Count			int (8)	
132.2	> Stalled			Boolean	
133	Counter7.Config	18214	Control	Cluster (32)	
133.0	> UpSource			Cluster (6)	
133.0.0	> > Channel			uint (4)	
133.0.1	> > Module			uint (1)	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
133.0.2	> > AnalogTrigger			Boolean	
133.1	> DownSource			Cluster (6)	
133.1.0	> > Channel			uint (4)	
133.1.1	> > Module			uint (1)	
133.1.2	> > AnalogTrigger			Boolean	
133.2	> IndexSource			Cluster (6)	
133.2.0	> > Channel			uint (4)	
133.2.1	> > Module			uint (1)	
133.2.2	> > AnalogTrigger			Boolean	
133.3	> IndexActiveHigh			Boolean	
133.4	> IndexEdgeSensitive			Boolean	
133.5	> UpRisingEdge			Boolean	
133.6	> UpFallingEdge			Boolean	
133.7	> DownRisingEdge			Boolean	
133.8	> DownFallingEdge			Boolean	
133.9	> Mode			uint (2)	
133.10	> PulseLengthThreshold			uint (6)	
134	Counter7.Reset	1821a	Control	Boolean	
135	Counter7.Output	1821c	Indicator	Cluster (32)	
135.0	> Direction			Boolean	
135.1	> Value			int (31)	
136	Counter7.TimerConfig	18220	Control	Cluster (32)	
136.0	> StallPeriod			uint (24)	
136.1	> AverageSize			uint (7)	
136.2	> UpdateWhenEmpty			Boolean	
137	Counter7.TimerOutput	18224	Indicator	Cluster (32)	
137.0	> Period			uint (23)	
137.1	> Count			int (8)	
137.2	> Stalled			Boolean	
138	Encoder0.Config	18228	Control	Cluster (21)	
138.0	> ASource			Cluster (6)	
138.0.0	> > Channel			uint (4)	
138.0.1	> > Module			uint (1)	
138.0.2	> > AnalogTrigger			Boolean	
138.1	> BSource			Cluster (6)	
138.1.0	> > Channel			uint (4)	
138.1.1	> > Module			uint (1)	
138.1.2	> > AnalogTrigger			Boolean	
138.2	> IndexSource			Cluster (6)	
138.2.0	> > Channel			uint (4)	
138.2.1	> > Module			uint (1)	
138.2.2	> > AnalogTrigger			Boolean	
138.3	> IndexActiveHigh			Boolean	
138.4	> IndexEdgeSensitive			Boolean	
138.5	> Reverse			Boolean	
139	Encoder0.Reset	1822e	Control	Boolean	
140	Encoder0.Output	18230	Indicator	Cluster (32)	
140.0	> Direction			Boolean	
140.1	> Value			int (31)	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
141	Encoder0.TimerConfig	18234	Control	Cluster (32)	
141.0	> StallPeriod			uint (24)	
141.1	> AverageSize			uint (7)	
141.2	> UpdateWhenEmpty			Boolean	
142	Encoder0.TimerOutput	18238	Indicator	Cluster (32)	
142.0	> Period			uint (23)	
142.1	> Count			int (8)	
142.2	> Stalled			Boolean	
143	Encoder1.Config	1823c	Control	Cluster (21)	
143.0	> ASource			Cluster (6)	
143.0.0	> > Channel			uint (4)	
143.0.1	> > Module			uint (1)	
143.0.2	> > AnalogTrigger			Boolean	
143.1	> BSource			Cluster (6)	
143.1.0	> > Channel			uint (4)	
143.1.1	> > Module			uint (1)	
143.1.2	> > AnalogTrigger			Boolean	
143.2	> IndexSource			Cluster (6)	
143.2.0	> > Channel			uint (4)	
143.2.1	> > Module			uint (1)	
143.2.2	> > AnalogTrigger			Boolean	
143.3	> IndexActiveHigh			Boolean	
143.4	> IndexEdgeSensitive			Boolean	
143.5	> Reverse			Boolean	
144	Encoder1.Reset	18242	Control	Boolean	
145	Encoder1.Output	18244	Indicator	Cluster (32)	
145.0	> Direction			Boolean	
145.1	> Value			int (31)	
146	Encoder1.TimerConfig	18248	Control	Cluster (32)	
146.0	> StallPeriod			uint (24)	
146.1	> AverageSize			uint (7)	
146.2	> UpdateWhenEmpty			Boolean	
147	Encoder1.TimerOutput	1824c	Indicator	Cluster (32)	
147.0	> Period			uint (23)	
147.1	> Count			int (8)	
147.2	> Stalled			Boolean	
148	Encoder2.Config	18250	Control	Cluster (21)	
148.0	> ASource			Cluster (6)	
148.0.0	> > Channel			uint (4)	
148.0.1	> > Module			uint (1)	
148.0.2	> > AnalogTrigger			Boolean	
148.1	> BSource			Cluster (6)	
148.1.0	> > Channel			uint (4)	
148.1.1	> > Module			uint (1)	
148.1.2	> > AnalogTrigger			Boolean	
148.2	> IndexSource			Cluster (6)	
148.2.0	> > Channel			uint (4)	
148.2.1	> > Module			uint (1)	
148.2.2	> > AnalogTrigger			Boolean	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
148.3	> IndexActiveHigh			Boolean	
148.4	> IndexEdgeSensitive			Boolean	
148.5	> Reverse			Boolean	
149	Encoder2.Reset	18256	Control	Boolean	
150	Encoder2.Output	18258	Indicator	Cluster (32)	
150.0	> Direction			Boolean	
150.1	> Value			int (31)	
151	Encoder2.TimerConfig	1825c	Control	Cluster (32)	
151.0	> StallPeriod			uint (24)	
151.1	> AverageSize			uint (7)	
151.2	> UpdateWhenEmpty			Boolean	
152	Encoder2.TimerOutput	18260	Indicator	Cluster (32)	
152.0	> Period			uint (23)	
152.1	> Count			int (8)	
152.2	> Stalled			Boolean	
153	Encoder3.Config	18264	Control	Cluster (21)	
153.0	> ASource			Cluster (6)	
153.0.0	> > Channel			uint (4)	
153.0.1	> > Module			uint (1)	
153.0.2	> > AnalogTrigger			Boolean	
153.1	> BSource			Cluster (6)	
153.1.0	> > Channel			uint (4)	
153.1.1	> > Module			uint (1)	
153.1.2	> > AnalogTrigger			Boolean	
153.2	> IndexSource			Cluster (6)	
153.2.0	> > Channel			uint (4)	
153.2.1	> > Module			uint (1)	
153.2.2	> > AnalogTrigger			Boolean	
153.3	> IndexActiveHigh			Boolean	
153.4	> IndexEdgeSensitive			Boolean	
153.5	> Reverse			Boolean	
154	Encoder3.Reset	1826a	Control	Boolean	
155	Encoder3.Output	1826c	Indicator	Cluster (32)	
155.0	> Direction			Boolean	
155.1	> Value			int (31)	
156	Encoder3.TimerConfig	18270	Control	Cluster (32)	
156.0	> StallPeriod			uint (24)	
156.1	> AverageSize			uint (7)	
156.2	> UpdateWhenEmpty			Boolean	
157	Encoder3.TimerOutput	18274	Indicator	Cluster (32)	
157.0	> Period			uint (23)	
157.1	> Count			int (8)	
157.2	> Stalled			Boolean	
158	Encoder4.Config	18278	Control	Cluster (21)	
158.0	> ASource			Cluster (6)	
158.0.0	> > Channel			uint (4)	
158.0.1	> > Module			uint (1)	
158.0.2	> > AnalogTrigger			Boolean	
158.1	> BSource			Cluster (6)	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
158.1.0	> > Channel			uint (4)	
158.1.1	> > Module			uint (1)	
158.1.2	> > AnalogTrigger			Boolean	
158.2	> IndexSource			Cluster (6)	
158.2.0	> > Channel			uint (4)	
158.2.1	> > Module			uint (1)	
158.2.2	> > AnalogTrigger			Boolean	
158.3	> IndexActiveHigh			Boolean	
158.4	> IndexEdgeSensitive			Boolean	
158.5	> Reverse			Boolean	
159	Encoder4.Reset	1827e	Control	Boolean	
160	Encoder4.Output	18280	Indicator	Cluster (32)	
160.0	> Direction			Boolean	
160.1	> Value			int (31)	
161	Encoder4.TimerConfig	18284	Control	Cluster (32)	
161.0	> StallPeriod			uint (24)	
161.1	> AverageSize			uint (7)	
161.2	> UpdateWhenEmpty			Boolean	
162	Encoder4.TimerOutput	18288	Indicator	Cluster (32)	
162.0	> Period			uint (23)	
162.1	> Count			int (8)	
162.2	> Stalled			Boolean	
163	Encoder5.Config	1828c	Control	Cluster (21)	
163.0	> ASource			Cluster (6)	
163.0.0	> > Channel			uint (4)	
163.0.1	> > Module			uint (1)	
163.0.2	> > AnalogTrigger			Boolean	
163.1	> BSource			Cluster (6)	
163.1.0	> > Channel			uint (4)	
163.1.1	> > Module			uint (1)	
163.1.2	> > AnalogTrigger			Boolean	
163.2	> IndexSource			Cluster (6)	
163.2.0	> > Channel			uint (4)	
163.2.1	> > Module			uint (1)	
163.2.2	> > AnalogTrigger			Boolean	
163.3	> IndexActiveHigh			Boolean	
163.4	> IndexEdgeSensitive			Boolean	
163.5	> Reverse			Boolean	
164	Encoder5.Reset	18292	Control	Boolean	
165	Encoder5.Output	18294	Indicator	Cluster (32)	
165.0	> Direction			Boolean	
165.1	> Value			int (31)	
166	Encoder5.TimerConfig	18298	Control	Cluster (32)	
166.0	> StallPeriod			uint (24)	
166.1	> AverageSize			uint (7)	
166.2	> UpdateWhenEmpty			Boolean	
167	Encoder5.TimerOutput	1829c	Indicator	Cluster (32)	
167.0	> Period			uint (23)	
167.1	> Count			int (8)	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
167.2	> Stalled			Boolean	
168	Encoder6.Config	182a0	Control	Cluster (21)	
168.0	> ASource			Cluster (6)	
168.0.0	> > Channel			uint (4)	
168.0.1	> > Module			uint (1)	
168.0.2	> > AnalogTrigger			Boolean	
168.1	> BSource			Cluster (6)	
168.1.0	> > Channel			uint (4)	
168.1.1	> > Module			uint (1)	
168.1.2	> > AnalogTrigger			Boolean	
168.2	> IndexSource			Cluster (6)	
168.2.0	> > Channel			uint (4)	
168.2.1	> > Module			uint (1)	
168.2.2	> > AnalogTrigger			Boolean	
168.3	> IndexActiveHigh			Boolean	
168.4	> IndexEdgeSensitive			Boolean	
168.5	> Reverse			Boolean	
169	Encoder6.Reset	182a6	Control	Boolean	
170	Encoder6.Output	182a8	Indicator	Cluster (32)	
170.0	> Direction			Boolean	
170.1	> Value			int (31)	
171	Encoder6.TimerConfig	182ac	Control	Cluster (32)	
171.0	> StallPeriod			uint (24)	
171.1	> AverageSize			uint (7)	
171.2	> UpdateWhenEmpty			Boolean	
172	Encoder6.TimerOutput	182b0	Indicator	Cluster (32)	
172.0	> Period			uint (23)	
172.1	> Count			int (8)	
172.2	> Stalled			Boolean	
173	Encoder7.Config	182b4	Control	Cluster (21)	
173.0	> ASource			Cluster (6)	
173.0.0	> > Channel			uint (4)	
173.0.1	> > Module			uint (1)	
173.0.2	> > AnalogTrigger			Boolean	
173.1	> BSource			Cluster (6)	
173.1.0	> > Channel			uint (4)	
173.1.1	> > Module			uint (1)	
173.1.2	> > AnalogTrigger			Boolean	
173.2	> IndexSource			Cluster (6)	
173.2.0	> > Channel			uint (4)	
173.2.1	> > Module			uint (1)	
173.2.2	> > AnalogTrigger			Boolean	
173.3	> IndexActiveHigh			Boolean	
173.4	> IndexEdgeSensitive			Boolean	
173.5	> Reverse			Boolean	
174	Encoder7.Reset	182ba	Control	Boolean	
175	Encoder7.Output	182bc	Indicator	Cluster (32)	
175.0	> Direction			Boolean	
175.1	> Value			int (31)	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
176	Encoder7.TimerConfig	182c0	Control	Cluster (32)	
176.0	> StallPeriod			uint (24)	
176.1	> AverageSize			uint (7)	
176.2	> UpdateWhenEmpty			Boolean	
177	Encoder7.TimerOutput	182c4	Indicator	Cluster (32)	
177.0	> Period			uint (23)	
177.1	> Count			int (8)	
177.2	> Stalled			Boolean	
178	Interrupt0.Config	182ca	Control	Cluster (9)	
178.0	> Source			Cluster (6)	
178.0.0	> > Channel			uint (4)	
178.0.1	> > Module			uint (1)	
178.0.2	> > AnalogTrigger			Boolean	
178.1	> RisingEdge			Boolean	
178.2	> FallingEdge			Boolean	
178.3	> WaitForAck			Boolean	
179	Interrupt0.RisingTimeStamp	182cc	Indicator	U32	
180	Interrupt0.FallingTimeStamp	182d0	Indicator	U32	
181	Interrupt1.Config	182d6	Control	Cluster (9)	
181.0	> Source			Cluster (6)	
181.0.0	> > Channel			uint (4)	
181.0.1	> > Module			uint (1)	
181.0.2	> > AnalogTrigger			Boolean	
181.1	> RisingEdge			Boolean	
181.2	> FallingEdge			Boolean	
181.3	> WaitForAck			Boolean	
182	Interrupt1.RisingTimeStamp	182d8	Indicator	U32	
183	Interrupt1.FallingTimeStamp	182dc	Indicator	U32	
184	Interrupt2.Config	182e2	Control	Cluster (9)	
184.0	> Source			Cluster (6)	
184.0.0	> > Channel			uint (4)	
184.0.1	> > Module			uint (1)	
184.0.2	> > AnalogTrigger			Boolean	
184.1	> RisingEdge			Boolean	
184.2	> FallingEdge			Boolean	
184.3	> WaitForAck			Boolean	
185	Interrupt2.RisingTimeStamp	182e4	Indicator	U32	
186	Interrupt2.FallingTimeStamp	182e8	Indicator	U32	
187	Interrupt3.Config	182ee	Control	Cluster (9)	
187.0	> Source			Cluster (6)	
187.0.0	> > Channel			uint (4)	
187.0.1	> > Module			uint (1)	
187.0.2	> > AnalogTrigger			Boolean	
187.1	> RisingEdge			Boolean	
187.2	> FallingEdge			Boolean	
187.3	> WaitForAck			Boolean	
188	Interrupt3.RisingTimeStamp	182f0	Indicator	U32	
189	Interrupt3.FallingTimeStamp	182f4	Indicator	U32	
190	Interrupt4.Config	182fa	Control	Cluster (9)	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
190.0	> Source			Cluster (6)	
190.0.0	> > Channel			uint (4)	
190.0.1	> > Module			uint (1)	
190.0.2	> > AnalogTrigger			Boolean	
190.1	> RisingEdge			Boolean	
190.2	> FallingEdge			Boolean	
190.3	> WaitForAck			Boolean	
191	Interrupt4.RisingTimeStamp	182fc	Indicator	U32	
192	Interrupt4.FallingTimeStamp	18300	Indicator	U32	
193	Interrupt5.Config	18306	Control	Cluster (9)	
193.0	> Source			Cluster (6)	
193.0.0	> > Channel			uint (4)	
193.0.1	> > Module			uint (1)	
193.0.2	> > AnalogTrigger			Boolean	
193.1	> RisingEdge			Boolean	
193.2	> FallingEdge			Boolean	
193.3	> WaitForAck			Boolean	
194	Interrupt5.RisingTimeStamp	18308	Indicator	U32	
195	Interrupt5.FallingTimeStamp	1830c	Indicator	U32	
196	Interrupt6.Config	18312	Control	Cluster (9)	
196.0	> Source			Cluster (6)	
196.0.0	> > Channel			uint (4)	
196.0.1	> > Module			uint (1)	
196.0.2	> > AnalogTrigger			Boolean	
196.1	> RisingEdge			Boolean	
196.2	> FallingEdge			Boolean	
196.3	> WaitForAck			Boolean	
197	Interrupt6.RisingTimeStamp	18314	Indicator	U32	
198	Interrupt6.FallingTimeStamp	18318	Indicator	U32	
199	Interrupt7.Config	1831e	Control	Cluster (9)	
199.0	> Source			Cluster (6)	
199.0.0	> > Channel			uint (4)	
199.0.1	> > Module			uint (1)	
199.0.2	> > AnalogTrigger			Boolean	
199.1	> RisingEdge			Boolean	
199.2	> FallingEdge			Boolean	
199.3	> WaitForAck			Boolean	
200	Interrupt7.RisingTimeStamp	18320	Indicator	U32	
201	Interrupt7.FallingTimeStamp	18324	Indicator	U32	
202	DMA.Rate	18328	Control	U32	
203	DMA.Config	1832c	Control	Cluster (22)	
203.0	> Pause			Boolean	
203.1	> Enable			Cluster (20)	
203.1.0	> > AI0_Low			Boolean	
203.1.1	> > AI0_High			Boolean	
203.1.2	> > AIAveraged0_Low			Boolean	
203.1.3	> > AIAveraged0_High			Boolean	
203.1.4	> > AI1_Low			Boolean	
203.1.5	> > AI1_High			Boolean	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
203.1.6	> > AIAveraged1_Low			Boolean	
203.1.7	> > AIAveraged1_High			Boolean	
203.1.8	> > Accumulator0			Boolean	
203.1.9	> > Accumulator1			Boolean	
203.1.10	> > DI			Boolean	
203.1.11	> > AnalogTriggers			Boolean	
203.1.12	> > Counters_Low			Boolean	
203.1.13	> > Counters_High			Boolean	
203.1.14	> > CounterTimers_Low			Boolean	
203.1.15	> > CounterTimers_High			Boolean	
203.1.16	> > Encoders_Low			Boolean	
203.1.17	> > Encoders_High			Boolean	
203.1.18	> > EncoderTimers_Low			Boolean	
203.1.19	> > EncoderTimers_High			Boolean	
203.2	> ExternalClock			Boolean	
204	DMA.ExternalTriggers0	18330	Control	Cluster (8) x 4 (32)	
204	> Trigger			Cluster (8)	
204.0	> > ExternalClockSource			Cluster (6)	
204.0.0	> > > Channel			uint (4)	
204.0.1	> > > Module			uint (1)	
204.0.2	> > > AnalogTrigger			Boolean	
204.1	> > RisingEdge			Boolean	
204.2	> > FallingEdge			Boolean	
205	DMA.ExternalTriggers1	18334	Control	Cluster (8) x 4 (32)	
205	> Trigger			Cluster (8)	
205.0	> > ExternalClockSource			Cluster (6)	
205.0.0	> > > Channel			uint (4)	
205.0.1	> > > Module			uint (1)	
205.0.2	> > > AnalogTrigger			Boolean	
205.1	> > RisingEdge			Boolean	
205.2	> > FallingEdge			Boolean	
206	Alarm.TriggerTime	18338	Control	U32	
207	Alarm.Enable	1833e	Control	Boolean	
208	Relay.Value	18342	Control	Cluster (8)	
208.0	> Forward			uint (4)	
208.1	> Reverse			uint (4)	
209	Power.Status	18344	Indicator	Cluster (24)	
209.0	> User3V3			U8	
209.1	> User5V			U8	
209.2	> User6V			U8	
210	Power.Disable	1834a	Control	Cluster (3)	
210.0	> User3V3			Boolean	
210.1	> User5V			Boolean	
210.2	> User6V			Boolean	
211	Power.UserVoltage6V	1834e	Indicator	U16	
212	Power.UserCurrent6V	18352	Indicator	U16	
213	Power.UserVoltage5V	18356	Indicator	U16	
214	Power.UserCurrent5V	1835a	Indicator	U16	
215	Power.UserVoltage3V3	1835e	Indicator	U16	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
216	Power.UserCurrent3V3	18362	Indicator	U16	
217	Power.VinVoltage	18366	Indicator	U16	
218	Power.VinCurrent	1836a	Indicator	U16	
219	Power.OnChipTemperature	1836e	Indicator	U16	
220	Power.MXP_DIOVoltage	18372	Indicator	U16	
221	Power.IntegratedIO	18376	Indicator	U16	
222	Power.AOVoltage	1837a	Indicator	U16	
223	Power.FaultCounts	1837c	Indicator	Cluster (32)	
223.0	> OverCurrentFaultCount3V3			U8	
223.1	> OverCurrentFaultCount5V			U8	
223.2	> OverCurrentFaultCount6V			U8	
223.3	> UnderVoltageFaultCount5V			U8	
224	Power.ResetFaultCounts	18382	Control	Boolean	
225	BIST.Enable	18386	Control	Boolean	
226	BIST.DO0SquareEnable	1838a	Control	Boolean	
227	BIST.DO0SquareTicks	1838c	Control	U32	
228	BIST.DO0	18392	Control	Boolean	
229	BIST.DO1SquareEnable	18396	Control	Boolean	
230	BIST.DO1SquareTicks	18398	Control	U32	
231	BIST.DO1	1839e	Control	Boolean	
232	AO.MXP0	183a2	Control	U16	
233	AO.MXP1	183a6	Control	U16	
234	SPI.ChipSelectActiveHigh	183aa	Control	Cluster (5)	
234.0	> Hdr			uint (4)	
234.1	> MXP			uint (1)	
235	SPI.EnabledIO	183ae	Control	uint (5)	
236	SPI.AutoSPI1Select	183b2	Control	Boolean	
237	SPI.AutoByteCount	183b6	Control	Cluster (11)	
237.0	> TxByteCount			uint (4)	
237.1	> ZeroByteCount			uint (7)	
238	SPI.AutoForceOne	183ba	Control	Boolean	
239	SPI.AutoRate	183bc	Control	U32	
240	SPI.AutoTriggerConfig	183c2	Control	Cluster (9)	
240.0	> ExternalClockSource			Cluster (6)	
240.0.0	> > Channel			uint (4)	
240.0.1	> > Module			uint (1)	
240.0.2	> > AnalogTrigger			Boolean	
240.1	> RisingEdge			Boolean	
240.2	> FallingEdge			Boolean	
240.3	> ExternalClock			Boolean	
241	SPI.AutoChipSelect	183c6	Control	U8	
242	SPI.AutoTx0	183c8	Control	U8 x 4 (32)	
243	SPI.AutoTx1	183cc	Control	U8 x 4 (32)	
244	SPI.AutoTx2	183d0	Control	U8 x 4 (32)	
245	SPI.AutoTx3	183d4	Control	U8 x 4 (32)	
246	SPI.TransferSkippedFullCount	183d8	Indicator	U32	
247	SPI.DebugState	183de	Indicator	EnumU16	Idle, Check Window, Check Av
248	SPI.DebugSubstate	183e2	Indicator	U8	
249	SPI.DebugRevision	183e4	Indicator	U32	

Table 1 – continued from previous page

	Register	Offset	Dir	Type/Size	Comments
250	SPI.DebugEnabled	183e8	Indicator	U32	
251	SPI.DebugIntStat	183ec	Indicator	U32	
252	SPI.DebugIntStatReadCount	183f0	Indicator	U32	
253	Accel.ADDR	183f6	Control	U8	
254	Accel.CNTR	183fa	Control	U8	
255	Accel.DATO	183fe	Control	U8	
256	Accel.DATI	18402	Indicator	U8	
257	Accel.CNTL	18406	Control	U8	
258	Accel.STAT	1840a	Indicator	U8	
259	Accel.CNFG	1840e	Control	U8	
260	Accel.GO	18412	Control	Boolean	
261	HMB.Config	18414	Control	Cluster (24)	
261.0	> Enables			Cluster (24)	
261.0.0	> > AI0_Low			Boolean	
261.0.1	> > AI0_High			Boolean	
261.0.2	> > AIAveraged0_Low			Boolean	
261.0.3	> > AIAveraged0_High			Boolean	
261.0.4	> > AI1_Low			Boolean	
261.0.5	> > AI1_High			Boolean	
261.0.6	> > AIAveraged1_Low			Boolean	
261.0.7	> > AIAveraged1_High			Boolean	
261.0.8	> > Accumulator0			Boolean	
261.0.9	> > Accumulator1			Boolean	
261.0.10	> > DI			Boolean	
261.0.11	> > AnalogTriggers			Boolean	
261.0.12	> > Counters_Low			Boolean	
261.0.13	> > Counters_High			Boolean	
261.0.14	> > CounterTimers_Low			Boolean	
261.0.15	> > CounterTimers_High			Boolean	
261.0.16	> > Encoders_Low			Boolean	
261.0.17	> > Encoders_High			Boolean	
261.0.18	> > EncoderTimers_Low			Boolean	
261.0.19	> > EncoderTimers_High			Boolean	
261.0.20	> > Interrupts			Boolean	
261.0.21	> > PWM			Boolean	
261.0.22	> > PWM_MXP			Boolean	
261.0.23	> > Relay_DO_AO			Boolean	
262	HMB.ForceOnce	1841a	Control	Boolean	
263	ViControl	17000	Control	U32	
264	DiagramReset	17008	Control	U32	
265	ViSignature	17004	Indicator	U32 x 4 (128)	
266	InterruptEnable	16000	Control	U32	
267	InterruptMask	16008	Control	U32	
268	InterruptStatus	1600c	Control	U32	

2.2 Watchdog

The FPGA has a watchdog which, when fed properly, enables all the outputs on the robot (e.g. PWM, Relays, CAN (albeit, indirectly), etc).

Danger: Manually feeding the watchdog bypasses all the built in safeties preventing the robot from moving. Do not manually feed the watchdog on a robot that has any chance of injuring someone.

2.2.1 Feeding

The 2019 FPGA watchdog was updated to be slightly obfuscated, but is still based on a single byte challenge-response, so still has the same problem.

1. Write 0xb007 to SysWatchdog.Command
2. Read from SysWatchdog.Challenge (4 bytes as of 2019)
3. Apply the following deobfuscation to the challenge:

```
uint32_t swapped = obfuscated;

swapped = ((swapped >> 1) & (0x55555555)) | ((swapped & (0x55555555)) << 1);
// Swap consecutive pairs
swapped = ((swapped >> 2) & (0x33333333)) | ((swapped & (0x33333333)) << 2);
// Swap nibbles
swapped = ((swapped >> 4) & (0x0f0f0f0f)) | ((swapped & (0x0f0f0f0f)) << 4);

// XOR outer bytes of swapped with inner bytes of original obfuscated challenge
challenge = ( (swapped >> 24) ^ (obfuscated >> 16) ^ (obfuscated >> 8) ^ swapped ) & 0xff;
```

4. The official code combines the response with random data to reobfuscate it. The result is that the lower two bytes XORed together produce the response.

```
response = 0xcc | ((0xcc ^ responses[challenge]) << 8)
```

Note: This can be any byte (0xcc used above) other than 0x00. Using 0x00 will cause the watchdog to die when the response is 0x00. This is what lead to FIRST to release a late build season image, since the original didn't check.

When enabled, every 20ms there's a $\frac{1}{256}$ chance for the (lower byte of the) random obfuscation number to be 0, and a $\frac{1}{256}$ chance for the response to be 0. When both were 0 (on average, every 21.8 minutes), the watchdog would die.

5. Write the response to SysWatchdog.Command
6. Write 0xfeed to SysWatchdog.Command and go to step 2

To stop feeding, one should write 0xdead to SysWatchdog.Command

2.2.2 Challenge-response

The challenge-response is a single byte in 2019, just obfuscated, so simply intercepting, deobfuscating, and recording all the proper responses is trivial.

The current responses are:

```
uint8_t responses[256] = {
    118, 154, 243, 85, 81, 32, 153, 216, 137, 123, 26, 45, 165, 28, 100, 2,
    168, 234, 61, 231, 155, 250, 248, 74, 58, 230, 246, 69, 77, 163, 7, 52,
    106, 239, 240, 64, 188, 147, 254, 5, 126, 138, 121, 113, 229, 179, 49, 161,
```

(continues on next page)

(continued from previous page)

```

109, 57, 27, 18, 86, 148, 195, 55, 222, 211, 117, 210, 104, 3, 226, 67,
209, 217, 177, 12, 219, 101, 73, 145, 99, 164, 215, 214, 94, 78, 17, 0,
169, 225, 245, 119, 53, 157, 224, 107, 212, 51, 218, 63, 8, 174, 252, 96,
184, 223, 105, 38, 129, 82, 92, 134, 194, 80, 102, 48, 146, 36, 204, 242,
238, 253, 130, 40, 152, 182, 70, 178, 247, 84, 187, 235, 208, 42, 202, 71,
23, 54, 141, 175, 180, 20, 89, 29, 97, 108, 221, 156, 206, 236, 189, 143,
9, 30, 125, 35, 87, 144, 159, 90, 114, 191, 139, 6, 201, 171, 228, 213,
15, 65, 232, 50, 79, 150, 220, 133, 116, 181, 185, 16, 190, 91, 200, 47,
197, 203, 68, 21, 43, 167, 199, 241, 142, 255, 183, 158, 122, 251, 59, 176,
39, 1, 192, 83, 124, 37, 115, 132, 128, 22, 244, 249, 93, 46, 227, 11,
149, 172, 196, 136, 193, 207, 76, 13, 162, 19, 170, 10, 66, 131, 44, 110,
75, 31, 103, 33, 56, 198, 140, 120, 166, 72, 98, 95, 127, 112, 160, 25,
34, 186, 173, 135, 88, 14, 4, 237, 62, 24, 233, 60, 111, 205, 151, 41
};

```