PRO4 / Cockpit application level communication protocols

# Summary:

While the PRO4 system conforms to the published vrcsr communication protocol document, there are extrinsic interactions which Cockpit performs. This essentially forms an application layer protocol.

This document only covers the primary use cases of the PRO4 ROV. It does not cover accessory communications or the protocols involved in updating firmware. It also does not document seldom used special case operations such as tareing the IMU, resetting the ROV, etc.

# Cockpit Communications loop:

The cockpit communications loop can be thought of as state machine.

The various states are:

1. Autodetect mode
2. Init mode
   1. Get Config data
   2. Populate CSR
3. Primary operation mode
4. Engine room mode

Note that Cockpit can also enter other non primary modes such as updating the firmware and requesting various reset operations.

# Cockpit Communication modes:

## Autodetect mode

Cockpit sends a detection packet out the PC comm ports.

The communication mode will switch on reception of a reply ACK packet.

The detection packet is a 0 length request with a FLAG byte set

Cockpit Sent Request Packet:

|  |  |
| --- | --- |
| FA AF 01 01 00 00 55 00 |  |

This is a standard zero length request packet with an ID of 0x1 and Flag of 0x1.

Note the appended 0x00 for a zero length packet.

**Any** well-formed packet response is considered a valid reply.

Example ROV response:

FD DF 01 01 00 09 2B 01 EE 01 25 00 E2 FF 00 00 D6

(Note here the ROV is responding with a 9 byte date payload. (in this case this is due to historical reasons and the payload data can be ignored.)

Upon reception of a valid response packet, the cockpit communication loop will transition to the *Init mode state*

## Init mode

### Read Configuration data

Once an ROV has been detected on a comm link, Cockpit will read the Config data.

This action is done by performing a standard CSR read on the special register

CONFIG\_DATA == 0xF7

Cockpit Sent Request Packet:

FA AF 01 80 F7 00 23 00

The ROV will respond with a packet containing the configuration data/

FD DF 01 80 F7 FF AB 25 01 24 01 01 00 01 00 4A

31 34 34 36 31 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 60 0D F0 0D 01 00 00 00 50

72 6F 34 43 50 55 32 30 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 50

72 6F 34 50 6F 77 65 72 31 36 43 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 50

72 6F 34 50 72 65 73 73 75 72 65 30 34 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 50

72 6F 34 43 61 6D 49 4F 30 32 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 50

72 6F 34 43 61 6D 30 31 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 50

72 6F 34 4C 45 44 4D 61 69 6E 30 33 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 54

68 72 75 73 74 42 4C 30 31 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 54

68 72 75 73 74 42 4C 47 30 31 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 B5

This is a standard response packet for CSR read from address 0xF7. It is a 293 byte packet.

The payload data (offset 0x7) can be mapped to two concatenated packed data structures.

The 32 reserved byte bootloader data structure, followed by the 260 byte Hardware config structure. (Note that the device type byte is the 1st byte in the payload data, for a total of 1 + 32 + 260 data bytes)

The data structures used:

/\*\* bootloader data struct in EEPROM \*/

#define BOOTLOADER\_SN\_SIZE 16

struct BOOTLOADER\_DATA {

uint16\_t bootloader\_version;

uint16\_t device\_type;

char serial\_number[BOOTLOADER\_SN\_SIZE];

char RESERVED[8]

uint32\_t valid\_application\_flag;

};

/\*\* Size of individual data fields in standard generic hardware config struct

\*/

#define PRO4\_HW\_CONFIG\_FIELD\_LEN 32

/\*\* Hardware config structure for the PRO4

\*/

typedef struct HARDWARE\_CONFIG\_PRO4\_tag {

uint32\_t DataStructVersionNumber;

char BOARD\_REV\_CPU[PRO4\_HW\_CONFIG\_FIELD\_LEN];

char BOARD\_REV\_POWER[PRO4\_HW\_CONFIG\_FIELD\_LEN];

char BOARD\_REV\_DEPTH[PRO4\_HW\_CONFIG\_FIELD\_LEN];

char BOARD\_REV\_CAMERA\_IO[PRO4\_HW\_CONFIG\_FIELD\_LEN];

char DEVICE\_REV\_CAMERA[PRO4\_HW\_CONFIG\_FIELD\_LEN];

char DEVICE\_REV\_LIGHTS[PRO4\_HW\_CONFIG\_FIELD\_LEN];

char DEVICE\_REV\_THRUSTER\_HORIZ[PRO4\_HW\_CONFIG\_FIELD\_LEN];

char DEVICE\_REV\_THRUSTER\_VERT[PRO4\_HW\_CONFIG\_FIELD\_LEN];

} HARDWARE\_CONFIG\_PRO4;

### Populate Full CSR data

Once the configuration data has been successfully retrieve the entire ROV csr is read.

This is done with a 0 length request packet.

The ROV will respond with a full read of the Device Specific Memory Map region of the CSR (0x0- - 0xF0)

(Note that the device type byte is the 1st byte in the payload data, for a total of 1 + 240 data bytes)

This populates the data representation of the ROV state in Cockpit.

Please see the PRO4 ROV CSR map for information on how to map this packet to semantic data values.

Cockpit Sent Request Packet

FA AF 01 80 00 00 D4 00

Example CSR response Packet:

FD DF 01 80 00 F1 52 01 00 00 00 00 00 00 00 00

00 00 00 FF FF FF FF 00 00 00 00 64 7C 01 7C 01

90 01 98 03 98 03 84 03 32 00 32 00 32 00 38 FF

FC FF 00 00 10 27 58 02 28 00 40 1F E8 03 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 E4 0C

D8 0E 01 00 01 00 01 00 01 00 0A 00 F4 01 0B 00

32 64 00 00 00 00 00 64 FF 5C 00 00 01 00 00 00

EE 01 25 00 E2 FF 00 00 00 00 00 00 63 02 EE 01

AE DC 56 04 80 1D C3 04 81 01 00 00 31 00 00 00

00 00 00 00 F3 C6 07 00 F8 01 F4 03 06 02 DD 01

3A 01 37 01 30 01 2B 01 CC 0A CC 0A CC 0A CC 0A

CC 0A 62 0B CC 0A CC 0A CC 0A 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 76 01 45 FD

91 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 02 06 06 00 00 7C

## Primary operation mode

The primary mode of operation is a simple 15 byte request packet, writing the first 10 datum in the CSR.

This mode of operation accounts for over 90% of the traffic between Cockpit and the ROV.

An application specific response packet is sent as a reply. This is mainly done for bandwidth and latency concerns.

Cockpit Sent Request Packet

FA AF 01 05 00 0F 5E 00 00 00 00 63 00 00 00 00

00 00 FF FF FF FF 63

Example CSR response Packet:

FD DF 01 05 00 22 04 01 ED 01 25 00 E2 FF 00 00

FF FF 00 00 00 00 67 02 FE 01 B5 DC 76 01 45 FD

A1 07 00 00 30 00 56 04 64 49

/\*\* Attitude depth plus water temp \*/

typedef struct Response\_FullNav\_Health\_tag {

/\*\* full attitude \*/

Response\_Attitude attitude;

/\* full rotatianl accels \*/

int16\_t accel\_rotation\_yaw;

int16\_t accel\_rotation\_pitch;

int16\_t accel\_rotation\_roll;

/\* full linear accels \*/

int16\_t accel\_rotation\_surge;

int16\_t accel\_rotation\_sway;

int16\_t accel\_rotation\_heave;

/\* full linear accels \*/

int16\_t raw\_mag\_x;

int16\_t raw\_mag\_y;

int16\_t raw\_mag\_z;

/\* attitude sensor status \*/

int16\_t attitude\_sensor\_status;

/\*\* internal temp \*/

int16\_t temp;

/\*\* internal humidity \*/

int16\_t rh;

uint8\_t total\_power\_available;

} Response\_FullNav\_Health;

## Engine Room Mode

When the cockpit engine room screen is open the Cockpit switches from the primary operation mode to sending significantly larger packets.

When in engine room mode Cockpit will execute a full command CSR write request. This is a write to the complete set of writeable registers in the CSR, excluding special function registers. For the PRO4 ROV This is equivalent to a write to the first 90(0x5A) bytes of the CSR memory space.

A full CSR read is the standard response requested.

Please see the PRO4 ROV CSR map for information on how to map this packet to semantic data values.

Cockpit Sent Request Packet

FA AF 01 80 00 5A 8E 00 00 00 00 63 00 00 00 00

00 00 FF FF FF FF FF FF 00 00 64 7C 01 7C 01 90

01 98 03 98 03 84 03 32 00 32 00 32 00 38 FF FC

FF 00 00 10 27 58 02 28 00 40 1F E8 03 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 E4 0C D8

0E 01 00 01 00 01 00 01 00 0A 00 F4 01 0B 00 32

64 8A

Example CSR response Packet:

FD DF 01 80 00 F1 52 01 00 00 00 00 63 00 00 00

00 00 00 FF FF FF FF FF FF 00 00 64 7C 01 7C 01

90 01 98 03 98 03 84 03 32 00 32 00 32 00 38 FF

FC FF 00 00 10 27 58 02 28 00 40 1F E8 03 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 E4 0C

D8 0E 01 00 01 00 01 00 01 00 0A 00 F4 01 0B 00

32 64 00 00 00 00 00 64 00 5C 00 00 01 00 00 00

7C FF F3 FF AB FF 00 00 01 00 00 00 46 FF 33 05

F5 DC 56 04 59 1B D2 04 DE 04 00 00 15 00 00 00

00 00 00 00 52 CC 07 00 1C 02 FF 03 0B 02 2F 01

27 01 28 01 24 01 2A 01 CC 0A CC 0A 93 07 CC 0A

CC 0A 62 0B CC 0A CC 0A CC 0A 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 48 03 85 FF

CF 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 02 06 06 00 00 5A