National Centre for Radio Astrophysics

Internal Technical Report

GMRT/ R-265

Technical Report on Online_v2 core software & Testing done

Raju Uprade

Email: rajsingh@gmrt.ncra.tifr.res.in

Revision	Date: 17 June 2015	Modification/ Change
Ver. 1		Initial Version R-265

INDEX

1. Abstract :

To control and coordinate the newly upgraded GMRT systems for performing astronomical observations, the new GMRT Monitor and Control (M&C) system is being developed. The new M&C system has a modern hardware and software architectural features as compare to the existing GMRT control system. The high end server class machines in the Central Electronics Building (CEB) are used to run the central supervisory M&C system, which communicates to all thirty antennas spread over a radial distance about ~ 15 km using a dedicated Ethernet/Optical-Fiber link at 1 Gbps.

Central supervisory M&C system host Online-V2 software which follows client-server software architecture and design is based on the TCP/IP communication. The multi-threaded M&C Application server program developed in C runs under Linux operating system. The number of clients running on various sub-systems at the antenna base like Front-end, servo and Feed Positioning System connects to the M&C application server via multiple communication channels using the TCP/IP communication. The clients for each sub-system self-discovers the connection automatically and send it's ID to the server. The M&C application server is capable of handling the communication channels for all thirty antennas in real-time at turn-around time of 0.5 to 1 sec. The M&C application software is integrated with Qt and QML (Qt Meta Language) based Graphical User Interface which provides the user interface to astronomer, telescope-operator and engineers in the CEB. The M&C system provide multiple type of interfaces like command-line for debugging, Qt based client and browser interface to view the data over Internet. M&C system log all event and monitoring data using MySQL database, alert the user on occurrence of alarms.

2. Background :

The Giant Meter wave Radio Telescope(GMRT)is being upgraded to allow wide band frequency observation in a seamless frequency coverage from 150 to 1500 MHz along with the modern servo systems for antennas and feeds. The control and monitor upgrade is required to speed up antenna communication by exploiting the power of the upgraded Ethernet over OF. This alongwith the upgraded servo PC104 would result in reduced overhead times for several antenna control operations.

Work related to Online_V2 started in October 2012. First steps were to study the existing ONLINE software (Software which is being in GMRT for last 15 years) and the Linux version referred to as TELESET-ABCCOM as well as we had extensive discussions in the team. As the second steps studied various control & monitoring software-tools used in other astronomy institute like SCADA, PVSS, moniCA. For exploring ready made software, Installed PVSS software on Laptop & explored various functionalities of it. After exploring various software, it was agreed that we will develop our own Online_V2 with open source technologies. Development work for Online_V2 started from January 2013.Initially we developed a stand alone readline server which can send commands to Rabbit MCM card using our own defined structure protocol. Online_v2 builds on our experience from TELESET-ABCCOM, C&M 15 M NCRA telescope in Pune & GMRT ONLINE SOFTWARE. This report explains how this software has been implemented and tested:

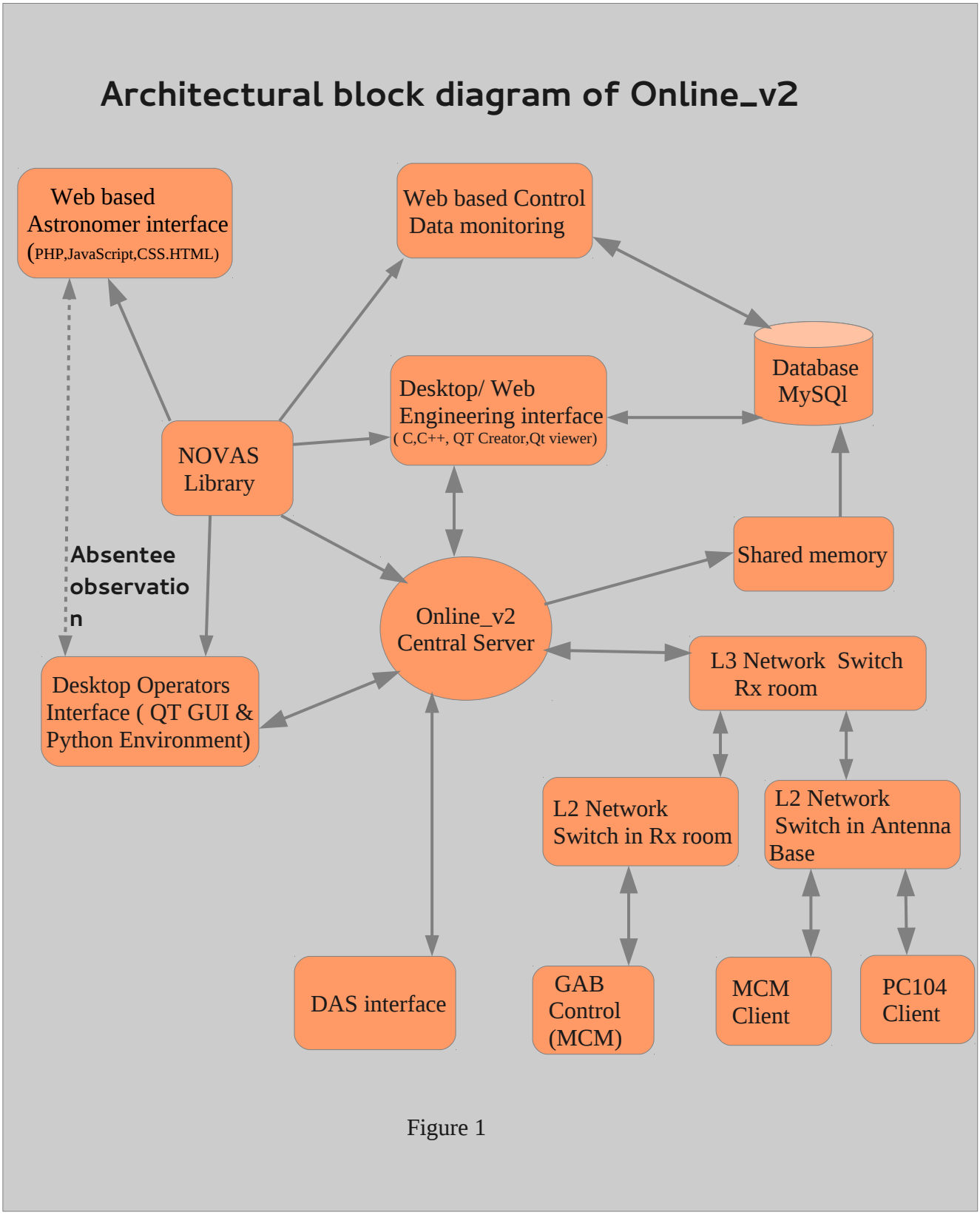

Architectural block diagram of Online_v2
Web based Astronomer interface (PHP,JavaScript,CSS.HTML)
Web based Control Data monitoring
Database MySQl
Desktop/ Web Engineering interface (C,C++, QT Creator,Qt viewer)
NOVAS Library
Absentee observatio n
Shared memory
Online_v2 Central Server
Desktop Operators Interface (QT GUI & Python Environment)
L3 Network Switch Rx room
L2 Network Switch in Rx room
L2 Network Switch in Antenna Base
DAS interface
GAB Control (MCM)
MCM Client
PC104 Client

Figure 1

2.1 Online_v2 core : Why it is required :

Online_v2 core software is the central software which is interfaced with all other antenna sub-system as well as DAS system. Online_v2 core is the master control program with a command terminal from where user can issues commands to any antenna sub-system & DAS system, write the command & response to shared memory, create user & sub array. It is the nerve center of Online_v2 control and monitoring system.

2.2 Online_v2 core : Interfacing with other system :

Online_v2 has been interfaced with antenna sub-system, desktop GUI (QML/QT) and Python Environment using TCP/IP based socket, using message queue to interact with DAS server program and shared memory to write all command-response into the shared memory as well as Online_v2 core software write XML files which can be read by any web based program to display data.

3. Structure : why we followed particular structure :

In Online_v2 there is no need of ANTCOM since Rabbit MCM and Ethernet has ensured that the load can be shared by the central Online_v2 computer and Rabbit MCM. Online_v2 core directly communicates with antenna sub-system without the need of ABCCOM (ANTCOM) because we have shifted most of the intelligence and work in the low level device called as rabbit processor MCM card, which is much more powerful than currently used ANTCOM. During the designing of Online_v2 control and monitor system we have optimized the work to be done by rabbit processor based MCM card as well as all other component of the whole system.

4. Implementation :

Online_v2 core software has been implemented as a multi-threaded work queue based server with each thread assigned specific jobs to perform. Creating a multi-threaded TCP/IP protocol based server requires the capabilities to handle network communication, multi-threading and transferring data between threads.

This server opens multiple thread which runs in parallel to accomplish the task. Out of them, one thread runs Readline library enabled high user interface which accepts commands from user through terminal, one thread to accept command from Python environment and one thread fro accepting commands from desktop GUI (QML/QT) based. All other threads run in background performing 1 seconds background monitoring of every sub-system as well as take command out of the work queue whenever command is there for them to send to particular sub-system.

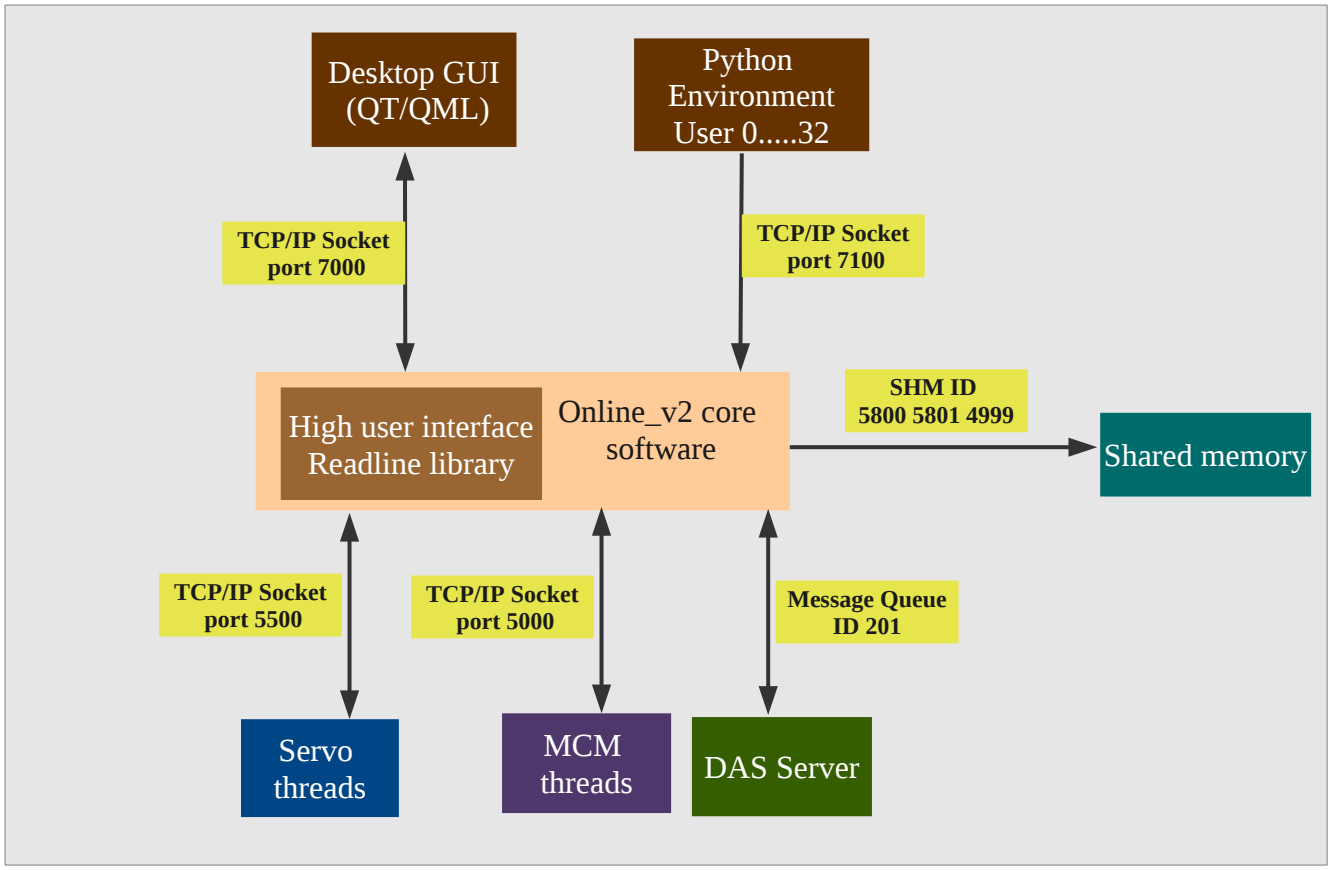
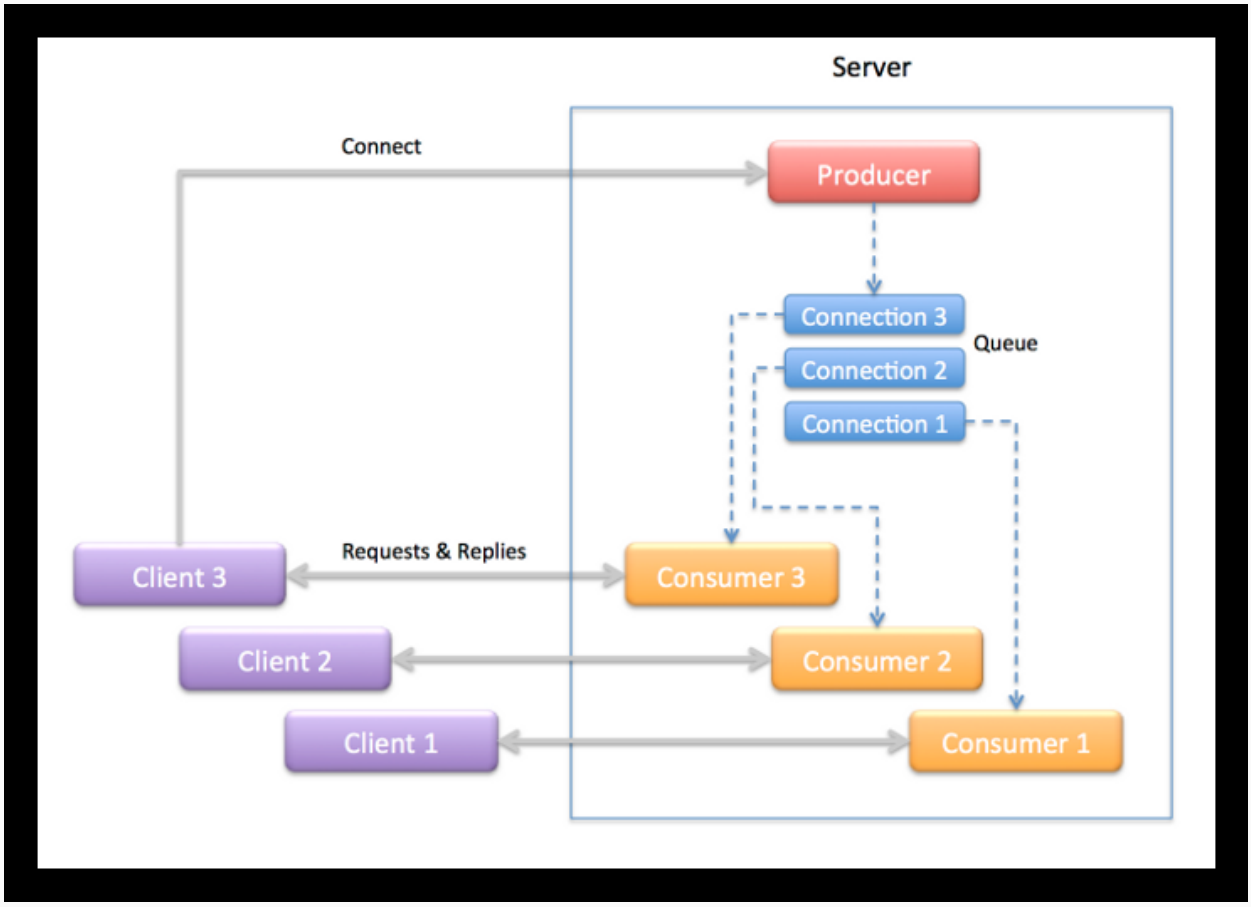


Fig : Block Diagram of Online_v2 core Software

4.1 Server Structure

Producer–Consumer Model

The server is based on the producer-consumer multi threaded model where a single producer thread passes work items to 1 or more consumer threads via a work queue. In the case of the TCP/IP server, the producer thread accepts connections then queues the connections for the consumer threads which in turn handle the connection processing as shown in this diagram.



4.2 Producer Thread

The producer thread in the server is implemented in the main() function. It's job is to create the work queue and consumer threads then accept connections from clients and pass the connections off to the consumer threads to handle. Specifically, the producer thread takes the following actions:

1.Create a work queue object.

2.Create the consumer threads.

3.Start listening for connections from clients.

4.Wait to accept a connections from a client .

5.For each connection create a work item that transfers the connected socket to a consumer thread to handle the connection.

6.Return to step 4.

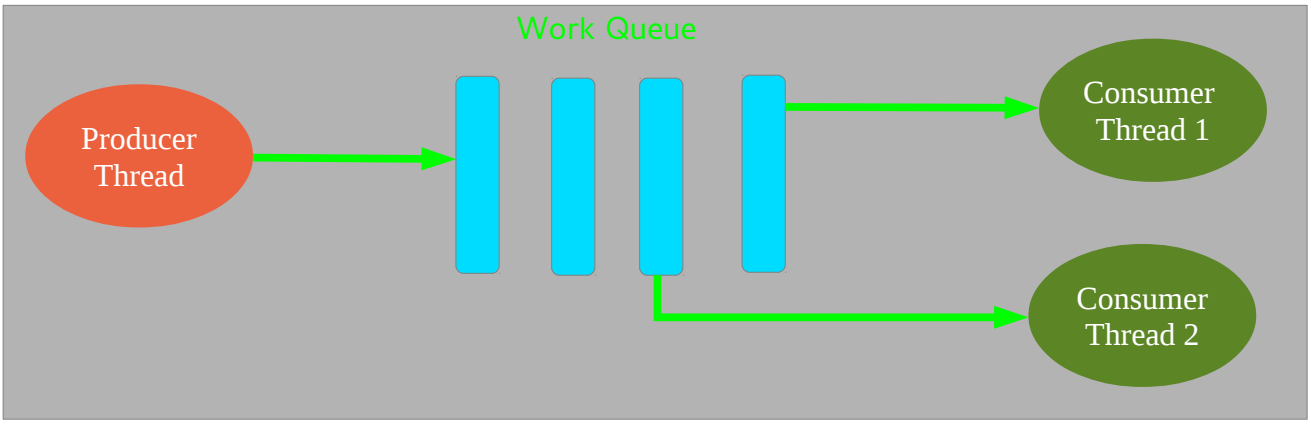
4.3 Consumer Thread

The consumer threads are the workers that do the protocol session handling for the server. Each consumer thread handles a connection in the following manner:

1. Wait for a work item to be added to the queue.
2. Remove a work item from the queue.
3. Extract the object from the work item.
4. Wait to receive a request from the client.
5. Process the request when it is received.
6. Send the reply back to the client.
7. Repeat steps 4 – 6 until the client closes the connection.
8. Close the server end of the connection when the client closes the connection.
9. Delete the work item.
10. Return to step 1.

4.4 Multi-threaded Work Queue :

Building a multi-threaded application is to devise a means to distribute tasks between threads so they can be processed concurrently. Queues are good devices for transferring work items from one thread to another. One approach to multi threading is the producer-consumer model where one thread – the producer –places work items in a queue and one or more consumer threads waits for and removes the items to process.



When a consumer thread runs it checks the number of items on the queue. If there are one ore more work items on the queue the consumer removes one and processes it. If none are available the consumer waits for the producer to add items to the queue. These steps are repeated continually for the lifetime of the application.

4.5 Online_v2–MCM–Servo Communication Protocol :

In Online_v2 development we are using generic structure based communication protocol to communicate command & response between Online_v2 program and MCM rabbit card. We had developed communication protocol for the 15 Meter NCRA CMS system in the Pune campus. Amongst the important changes were trimming the command structure from 5000 bytes to 1600 bytes and response structure from 17000 to 5000 bytes as number of variable required were reduced. We note that communication protocol developed for 15m telescope has been working without any problem for last three year.
Connection is made using TCP/IP networking socket programming between Online_v2 and MCM program. Connection remains persistent means connections are not broken after exchanging information. Connections are broken only if MCM device is power off.

4.6 Command Structure for Online_v2–MCM communication :

```
typedef struct
{
  int seq;                              // Unique Sequence
  char timestamp[64];                   //  Timestamp of command
  char system_name[16];                 // System Name for which command is
  char op_name[16];                     // Operation to perform ( Init/Set/Mon/Reset )
  short int number_param;               // Number of parameter
  char parameter_name[32][16];          // Parameter Name
  char Argument_Ch1[32][16];            // Channel One argument
  char Argument_Ch2[32][16];            // Channel Two argument
 } cmd;
```

Response structure for MCM–Online_v2 communication :

```
typedef struct
{
 int response_type;                     // Response type
 int seq;                               // Sequence number
 char timestamp[64];                    // Time stamp
 char system_name[16];                  // System name
 char Mon_raw[64][8];                   // 64 channel raw data
 char Mon_sum[32][64];                  // Monitoring summary prepared from 64 channel raw data
 short int num_resp_msg;                // Number of Response Message
 char response_message[32][64];         // Response message from MCM
} resp;
```

Structure used to send command for a particular antenna :

```
typedef struct
{
  char antenna_name[10];
  cmd  CMD;                    // command structure is embedded in the antenna structure
} ANT_CMD;
```

Structure used for creating user & sub-array :

```
typedef struct
{
   int sub_num;
   int num_in_sub;
   char ant_name[32][8];
 }sub;
typedef struct
{
   int usernb;
   sub s_a[4];
   char timestamp[26];
} user;

typedef struct
{
   user Tx[4];
} setdata;
```

Command structure for Online_v2-Servo Communication :

```
typedef struct
{
  int seq;
  char timestamp[64];
  char system_name[16];
  char op_name[16];
  short int number_param;
  char para_name[32][16];
  char para_value[32][16];
 } servocmd;

typedef struct
{
  char antenna_name[10];
  servocmd  SRVCMD;
} SRVANT_CMD;
```

Response structure for Servo –Online_v2 Communication :

```
typedef struct
{
 int seq;
 char timestamp[64];
 char system_name[16];
 int response_code;          // immediate =1 ( ACK or NAK ) event=3 & final =2  data mon = 4
 int response_type;          // success =1 or failure = 2
 char response_msg[50];  // Accepted ,not accepted, syntax error, irrelevant  command,incomplete,unknown + Event
 short int num_resp_msg;
 char para_name[32][32];
 char para_value[32][32];
} servoresp;
```

Size of Online_V2-MCM command structure is 1638 bytes and MCM-Online_v2 size of response structure is 4698 bytes.
Size of Online_V2-Servo command structure is 1132 bytes and Servo-Online_v2 size of response structure is 2192 bytes.

We have been using this generic communication protocol structure in C03 antenna testing & in two antenna 4 sub-system each Lab test set-up. We have not found any major problem related to communication. This generic structure based communication protocol is working very well for our Online_v2 need.

In term of reducing time interval, we are continuously working on it. With the help of powerful machines & optimization we will be able to reduce the time interval as well.

4.7 Comparison between Present Online, Teleset–Abccom & Online_v2 protocol :

Present Online do all the hex code generation and decoding in Online machine and pass it on to communication handler through PC router on a serial link @9.6 kbps. All the intelligence is in Online code.

Teleset-Abccom protocol differ in a sense that most of the intelligence has been shifted to Abccom software. Teleset only gives higher level ASCII command like English text for ex: if setnew set1.dat . Abccom will decode what it means by if setnew set1.dat, generate the corresponding hex code for that and send that to MCM 10 (IF system) on serial link & get the response and form the response packet from the raw data sent by MCM 10 (IF system) which can be sent to Teleset.
In Online_v2, all the bit pattern generation and decoding of response from raw data is done by MCM card so we just have to send higher level command from Online_v2 by an agreed upon communication protocol. We are following a structure based communication protocol where we are sending a command structure to a particular MCM card and we are getting a response structure. In Online_v2 most of the intelligence is shifted in the MCM card itself.

4.8 DAS interfacing with Online_v2 :

DAS server starts reading corresel.hdr file generated by GSB configuration GUI. Online_v2 send commands and read response to & from DAS server via message queue id 201. Command received from Online_v2 is sent on sockets to the correlator system by DAS server. Response from correlator system is received on socket by DAS server, which is then written to message queue by DAS server to be read by Online_v2.

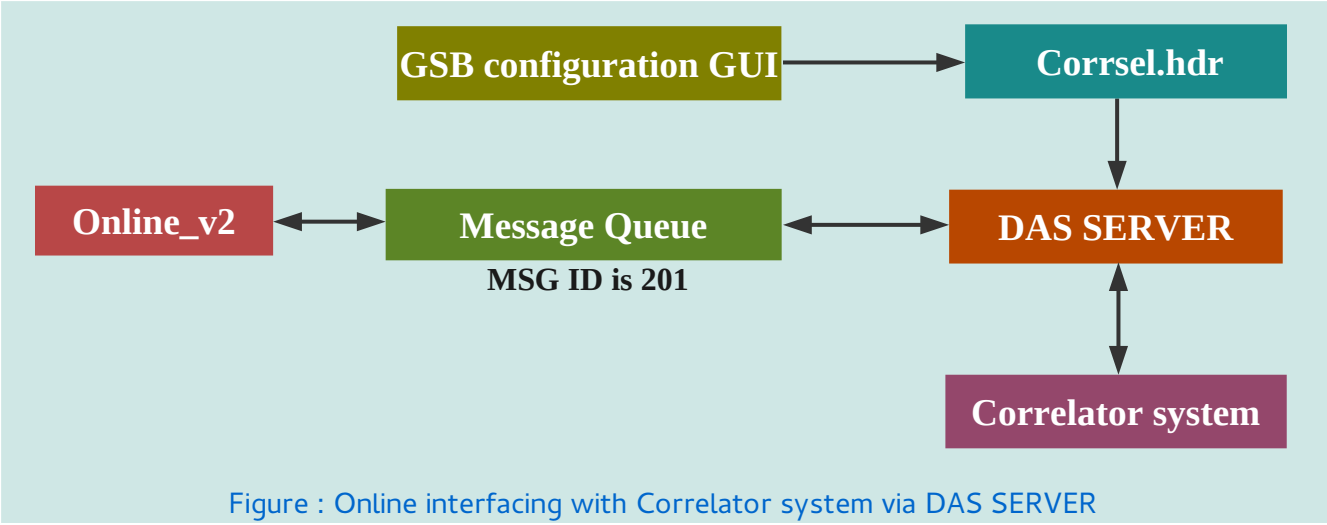


Figure : Online interfacing with Correlator system via DAS SERVER

There are mainly six commands which are send by Online to the DAS server :

1. initndas => DAS init
2. initproj => Add project
3. strtndas => Start project
4. stpndas => Stop
5. delproj => Delete Project
6. hltndas => Finish

initndas : [antmax] [corrsetup] [bandmask] [sub array no] [.....]

initprj : [antmax] [sub array] [proj title] [proj object] [band mask selection] [prjcode]

strtndas : [source param] [freq param] [sub array number]

Antenna name mentioned during initndas & initproj should match.

DAS Command structure :

	sub array Num	cmd num	Cmd type	CMD	CMODE	Antenna mask		File location
1.	0	0	das	init	8	7FFFFFFE	F	/home/teleset/gsb.hdr // Init Command
2.	0	1	das	addp	8			/home/teleset/proj.hdr.user0 // ADD project command
3.	4	2	das	start				/home/teleset/scan.hdr.user4 // DAS Start command
4.	4	3	das	stop				// DAS Stop command
5.	0	4	das	delp	8			// Delete project command
6.	0	5	das	fini	8			// Finish command

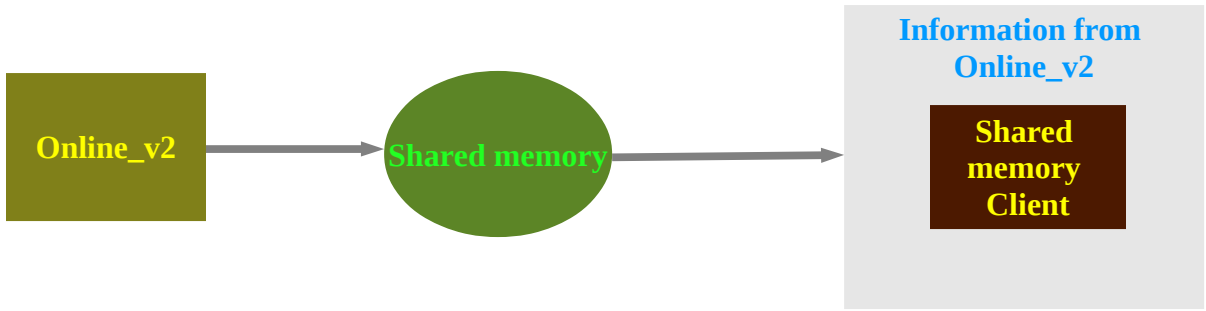
gsb.hdr => GSB config header file.

proj.hdr => generated by Online_v2

scan.hdr => generated by Online_v2.

4.9 Shared memory Client Program :

Online_v2 writes command-response from every thread of every antenna, user & sub-array information and DAS command-response messages from DAS server into the shared memory. Shared memory client program gets attached to the shared memory ID created by Online_v2 and reads all the information which can be displayed as per user's wish. Any number of clients can get attached to this shared memory segments.



4.10 Command implementation :

4.10.1. Command for User & sub-array creation issued from Online_v2 terminal :

1.	create	This command will create user & subarray with antenna in that particular group user input 0 => user & sub-array will created through file input user input 1 => user & sub-array will created through terminal input
2.	add	This command will add user,sub-array & antenna in the particular group
3.	cmd2sub	This command will send send command to the particular group of antenna
4.	delusr	This command will delete particular group user along with all information
5.	showuser	This command will display information about particular user

4.10.2. DAS command issued from Online_v2 terminal :

1.	dasinit	This command will Initialize DAS system
2.	Addp	This command will add project file which will be send to DAS system
3.	delprj	This command will delete the project file
4.	startscan	This command will start scanning
5.	stopscan	This command will stop scanning
6.	finish	This command will halt DAS system

4.10.3. Antenna command issued from Online_v2 terminal :

To give command to a particular antenna we have followed a particular format :
For example :

1.	C00 Sentinel init 1st string will be Antenna name 2nd string will be sub-system name 3rd string will be sub-system command

4.10.4. Sub-system commands :

4.1 Sentinel system commands

No.	Command Name	Description	Argument
1.	Init	Initialize the system	-
2.	Mon	Monitor the health of the system	-
3.	set	Set the system as per the Argument	dmask Ch1 Ch2
4.	reset	Reset the system	-

4.2 Front End system commands

No.	Command Name	Description	Argument
1.	Init	Initialize the system	-
2.	Mon	Monitor the health of the system	-
3.	set	Set the system as per the Argument	band_sel Ch1 Ch2 slr_attn Ch1 Ch2 channel Ch1 Ch2 sub_band_sel Ch1 Ch2 rf Ch1 Ch2 cal_ns Ch1 Ch2
4.	reset	Reset the system	-

4.3 Fiber Optics system commands

No.	Command Name	Description	Argument
1.	Init	Initialize the system	-
2.	Mon	Monitor the health of the system	-
3.	set	Set the system as per the Argument	rf_attn Ch1 Ch2
4.	reset	Reset the system	-

4.4 GMRT Analog Backend system commands

No.	Command Name	Description	Argument
1.	Init	Initialize the system	-
2.	Mon	Monitor the health of the system	-
3.	set	Set the system as per the Argument	reflo Ch1 Ch2 lo Ch1 Ch2 attn Ch1 Ch2 filter Ch1 Ch2 lpf Ch1 Ch2 source Ch1 Ch2 signal Ch1 Ch2 path Ch1 Ch2 channel Ch1 Ch2
4.	reset	Reset the system	-

4.5 Self Test system commands

No.	Command Name	Description	Argument
1.	Init	Initialize the system	-
2.	Mon	Monitor the health of the system	-
3.	set	Set the system as per the Argument	ss Ch1 //set spectrum spreader fdb Ch1 // set frequency doubler fdv Ch1 //set frequency divider
4.	reset	Reset the system	-

4.6 GMRT servo system commands

No.	Command Name	Description	Argument
1.	sendsacsrc	Send source information RA DEC to track routine	Source RA DEC
2.	goin	Inner Track	-
3.	goout	Outer Track	-
4.	trkon	Start Tracking	-
5.	trkoff	Stop Tracking	-
6.	coldstart		-
7.	hold	Hold Axis	Ax
8.	track	Track	Time,Ax,Ang1,Ang2
9.	position	Position Axis	Ax,Ang1,Ang2
10.	stow	Stow the antenna	Ax
11.	stow_release	Stow release	Ax
12.	stop	Stop	Ax
13.	abort	Abort tracking	
14.	close	Close	
15.	resethw	Reset Hardware	
16.	readangles	Read angles	
17.	readanavar	Read analog variables	
18.	readdigvar	Read digital variables	
19.	readantstatus	Read antenna status	
20.	readversion	Read version	
21.	readsetpara	Read set paramameters	

4.7GMRT servo system commands (Yet to be Implemented)

No.	Command Name	Description	Argument
1.	TRKELOFF	track elevation offset	New track parameters
2.	TRKAZOFF	track azimuth offset	New track parameters
3.	TRKANTOFF	track azimuth, elevation offset	New track parameters
4.	TRKRAOFF	track right ascension offset	New track parameters
5.	TRKDECOFF	track declination offset	New track parameters
6.	SCANELSRC	scan src in el with derv=ipa,ptime=jpa	New track parameters
7.	SCANAZSRC	scan src in az with derv=ipa,ptime=jpa	New track parameters
8.	SCANRASRC	scan src in ra with derv=ipa,ptime=jpa	New track parameters
9.	SCANDEC	scan src in dec with derv=ipa,ptime=jpa	New track parameters

4.8 GMRT Feed positioning system commands

No.	Command Name	Description	Argument
1.	mvpos	move to the feed position	feed 610/150/1420/325
2.	loadparam	load parameters from fps.dat file	File name
3.	fpsinit	Initialize FPS system	-
4.	fpsnull	FPS NULL command	-
5.	set_tpoint	turning point position	Encoder count
6.	set_rampdcnt	Set ramp down count	Nb timer ticks
7.	set_low_rpm	Set lower RPM limit	Nb timer ticks
8.	set_brake_dd	Set break down diff	Nb enc. pulses
9.	set_rampupcnt	Set ramp up count	Nb timer ticks
10.	set_stoptimecnt	Set stop time count	Nb timer ticks
11.	set_max_pwm_cnt	Set Max PWM count	PWM count
12.	set_max_angle	Set MAX angle	Angle
13.	set_min_angle	Set MIN angle	Angle
14.	read_tpoint	Read turning point	
15.	read_rampdcnt	Read ramp down count	
16.	read_low_rpm	Read lower RPM limit	
17.	read_brake_dd	Read break down diff	
18.	read_rampupcnt	Read ramp up count	

19.	read_stoptimecnt	Read stop time count	
20.	read_max_pwm_cnt	Read stop time count	
21.	read_max_angle	Read MAX angle	
22.	read_min_angle	Read MIN angle	
23.	read_version	Read version	
24.	run_to_cal	Run to calibrate	
25.	free_run_tow	Free run towards	270/-10 deg
26.	run_to_preset	Run to preset	Target pos
27.	run_fine_tune	Run to fine tune	Target pos, PWM count
28.	password_run	Password run	
29.	reboot	reboot	
30.	fpsstop	stop	

4.11 Online_v2 Source files details :

No.	File Name	Description
1.	main.c	Opens multiple threads for high user,mcm,python env,servo & GUI
2.	highuser.c , highuer.h	High user thread which accept user command through readline
3.	common.c	Common file for mostly used functions like timestamp etc.
4.	das.c , das.h	Source file to read,write to the das server using Message queue
5.	pyenv.c ,pyenv.h	Python environment thread to accept command from Python env
6.	server.h	Header file contain mostly used functions & communication structure.
7.	servo.h	Header file containing servo communication structure & functions declarations.
8.	servo.c	Source file for filling servo command structure
9.	shm.h	Header file for shared memory function declarations.
10.	shmed_mem.c	server which writes command & response to shared memory.
11.	shm_client.c	Client which read command & response from shared memory.
12.	system.c, system.h	Source file for filling MCM command structure (FE,Sentinel,GAB,OF & self test).
13.	fps.h, fps.c	Feed positioning system header & Source file for filling FPS command structure
14.	gui.c , gui.h	GUI thread to accept command from Desktop GUI (QT/QML)
15.	mcm.c , mcm.h	File for accepting client connection from various sub-system
16.	mcm_thread.c, mcm_thread.h	File for various MCM sub-system threads
17.	queue.c, queue.h	File for queue related function declarations & definition. .
18.	servo_queue.h, servo_queue.c	Servo queue related function declarations & definition.
19.	socket.c, socket.h	Socket related function declarations & definition.
20.	ssclink.h, ssclink.c	Servo communication function declarations & definition.
21.	trackservo.c, trackservo.h,	Source file containing algorithm for source tracking (Taken from ABCCOM servo tracking algorithm)
22.	user1.h	Header file for user input
23.	user_info.c	Source file for taking user input related to user & sub array creation.
24.	xml.c, xml.h	File for writing XML file of command & response.

5 . Testing of Online_v2 Core software :

5.1 Lab testing :

- ✔ Eight MCM cards along with L3 network switch have been put in Lab set up for Online_V2. Online_V2 has been tested successfully by configuring eight MCM cards as two antenna four sub-system each & eight antenna one sub-system each. (Front End, Fiber optics, GAB, Sentinel)

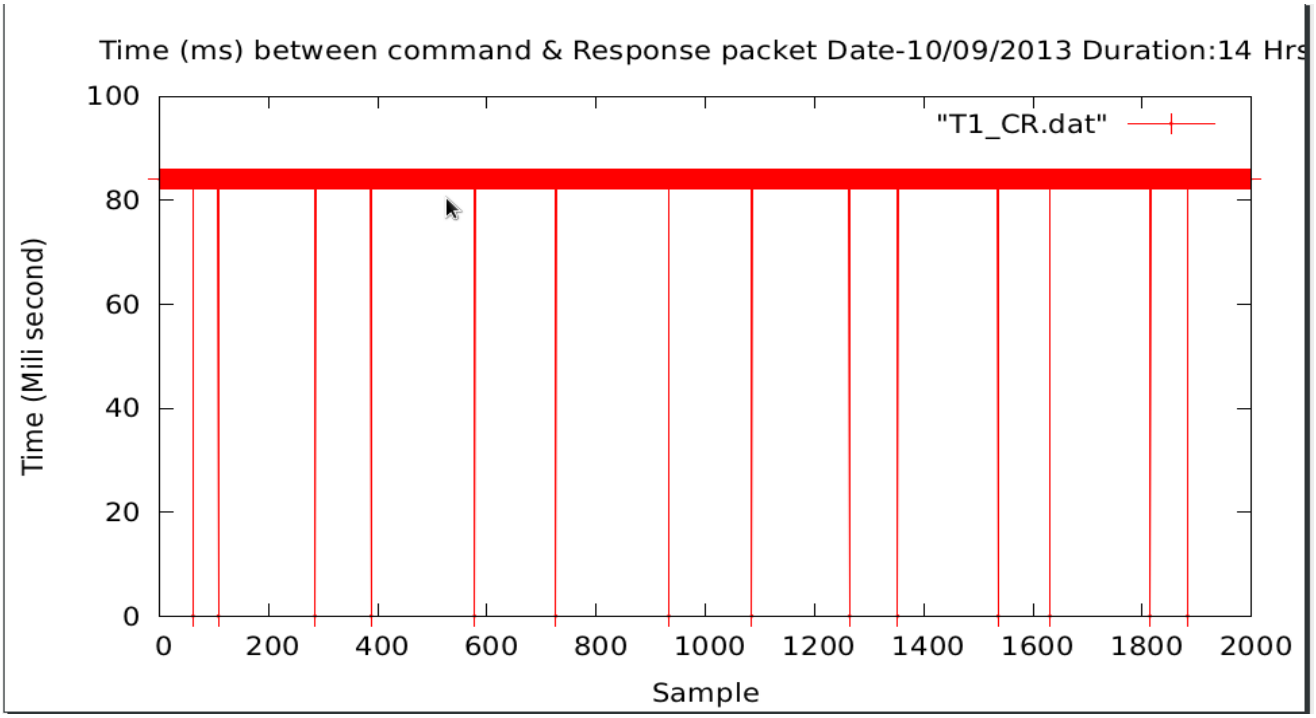
- ✔ All basic servo commands have been implemented in Online_v2 which has been tested with actual hardware set up in servo lab. As well as one antenna tracking routine has been implemented in Online_V2, which has been tested with a prototype servo client program running on PC104 card.

- ✔ We have done several test to measure the round trip time taken by command & response packet. Initially we get into the problem with two successive monitoring packet response time. It was working properly for 2 seconds and we were trying to achieve 1 second monitoring time. We tried increasing TCP MTU as well as tried various options in socket programming but it didn't solved our problem. After various trail & error we changed the sub-net i.e. we isolated our lab setup by putting a new L3 CISCO switch & IP of server machine from rest of the network. It improved our command-response round trip time significantly and we achieved 1 seconds as well 0.5 second round trip monitoring time. Refer to figures in "Plots of round trip time".

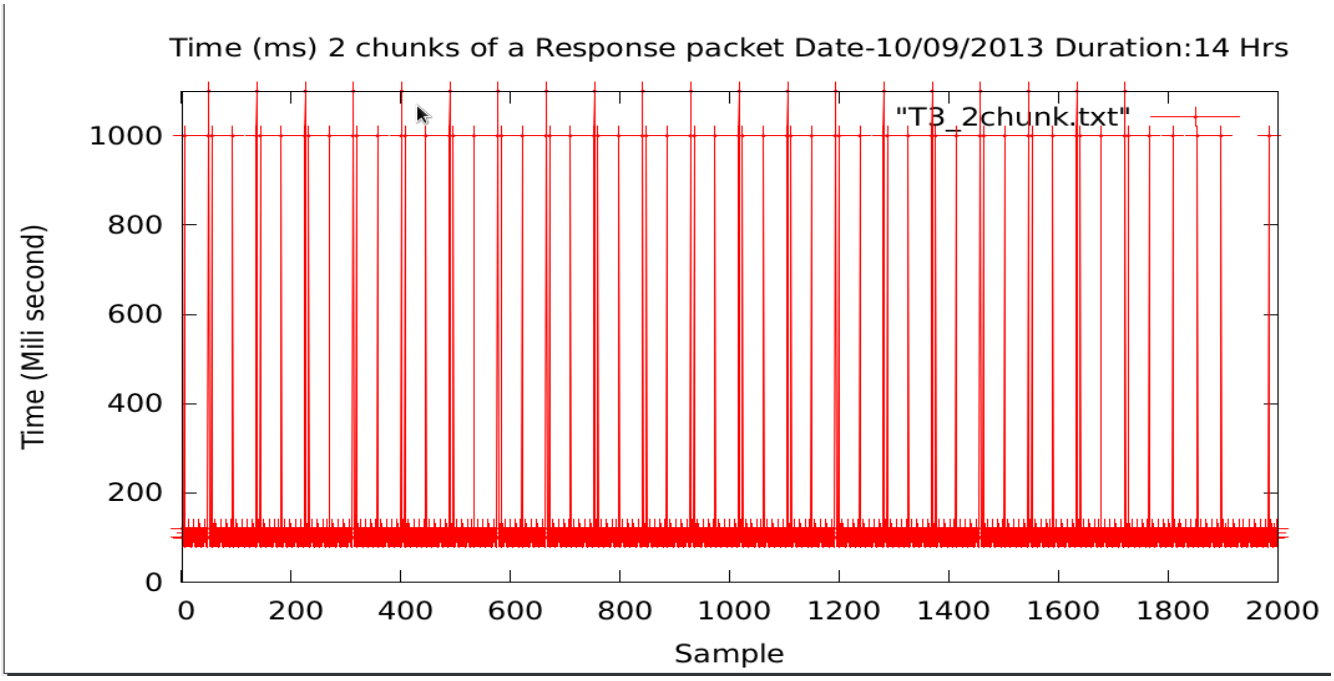
- ✔ Online_v2 software has been tested with Feed positioning system MCM card (NULL command) in telemetry Lab where Rabbit MCM card takes command from Online_v2 software over Ethernet & send it to the FPS MCM card over serial link and get back the response, compose Ethernet response packet and send it back to online_v2 software.

5.2 Plots of round trip time:

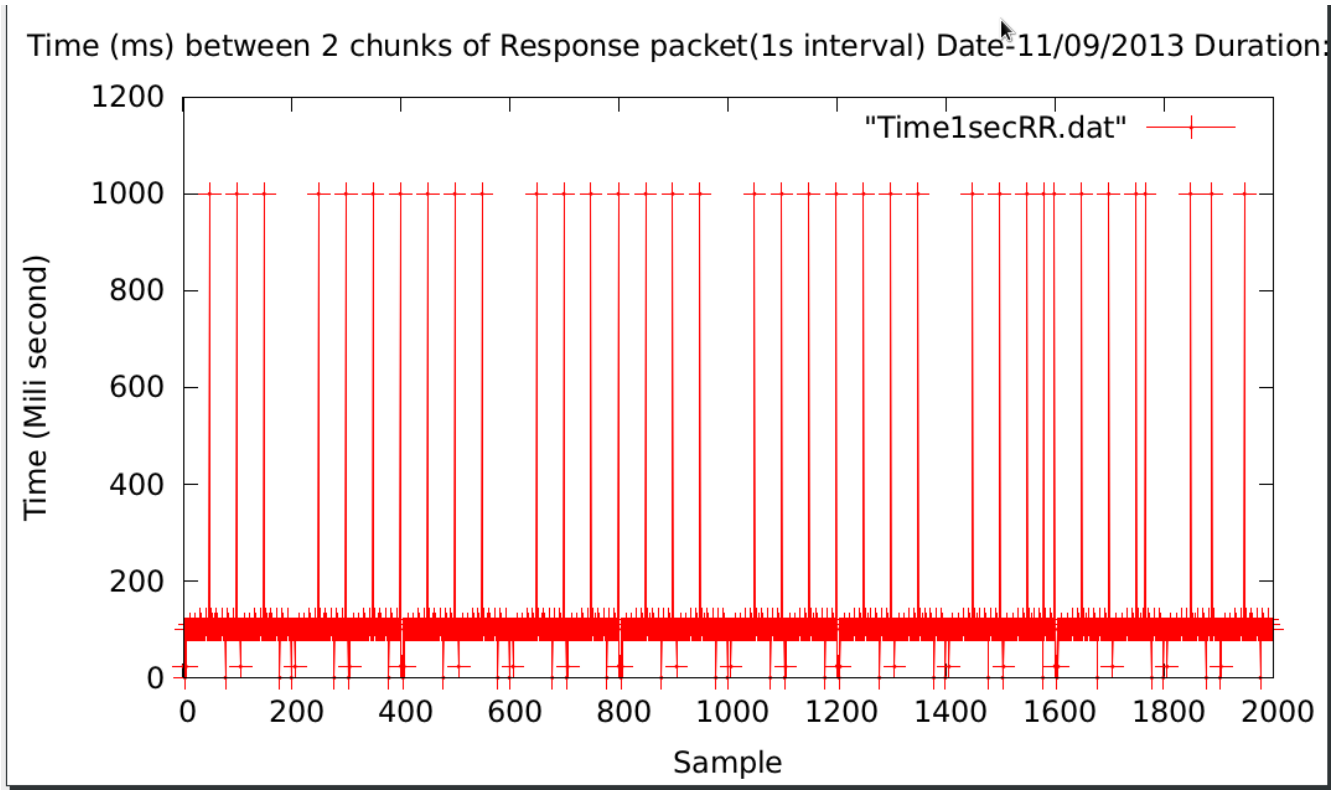
Plots below shows the round trip time when Online_v2 was connected with eight antenna set up in lab using a small unmanageable switch which is not configurable as well as which dose not has routing capability. Plots shows undesirable high dropout of packets.



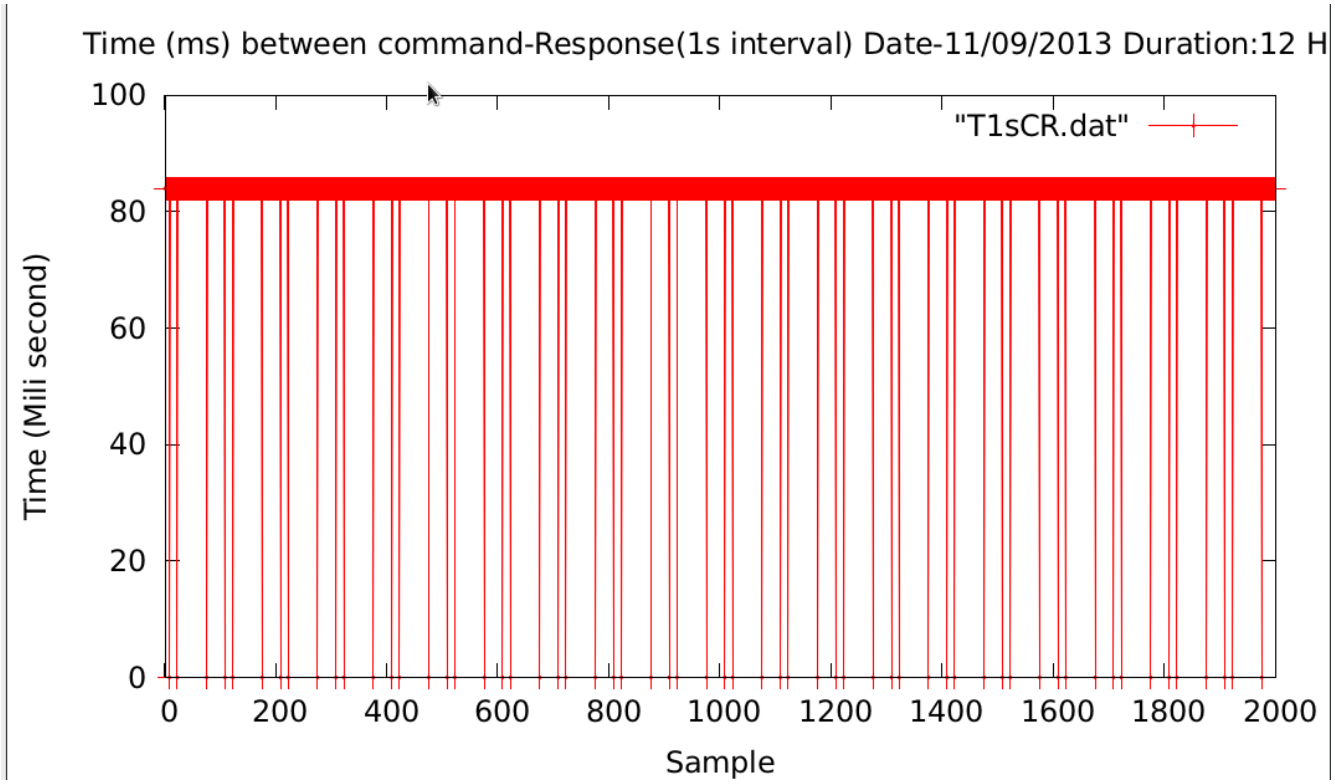
Fig(1): Time between command & response when interval is 2 seconds



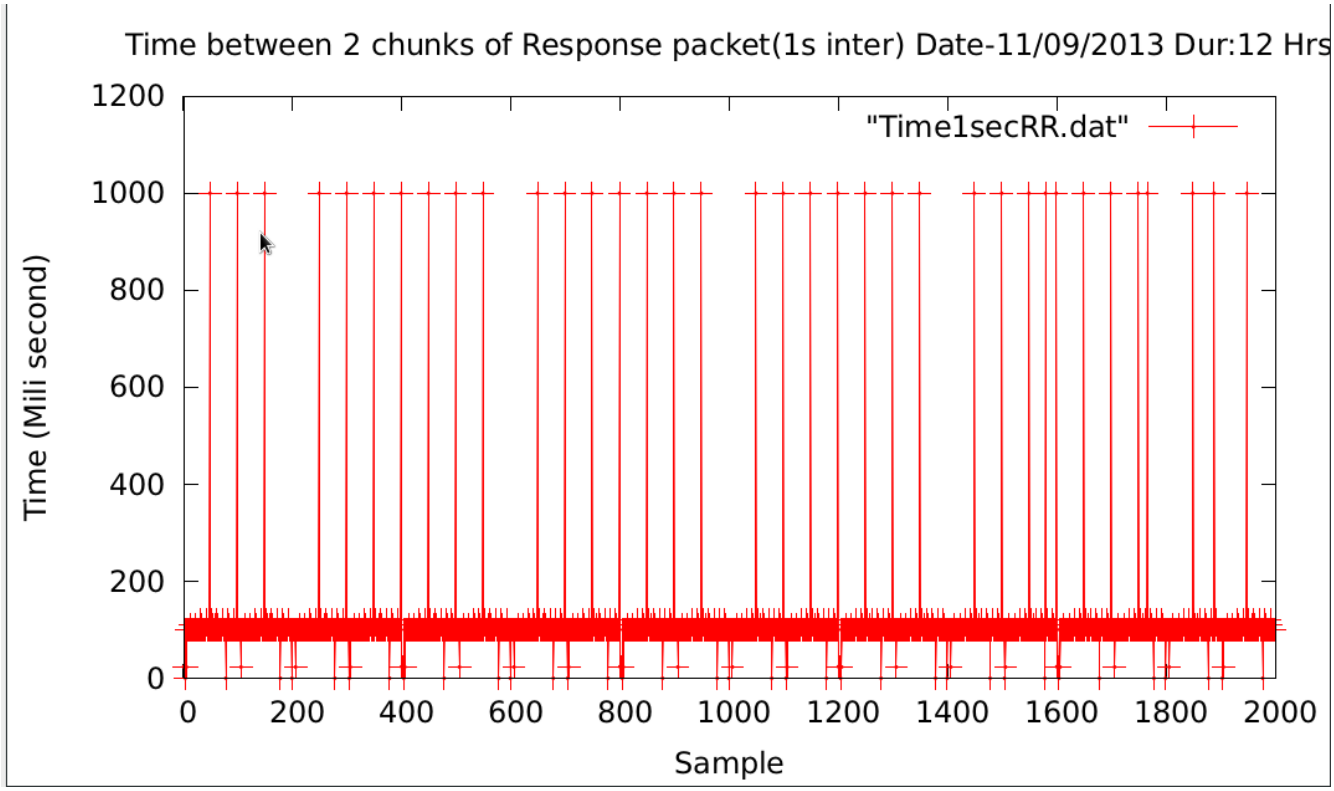
Fig(2) : Time between two chunks of a response packets in 2 second interval



Fig(3) : Time between two chunks of a response packets in 1 second interval

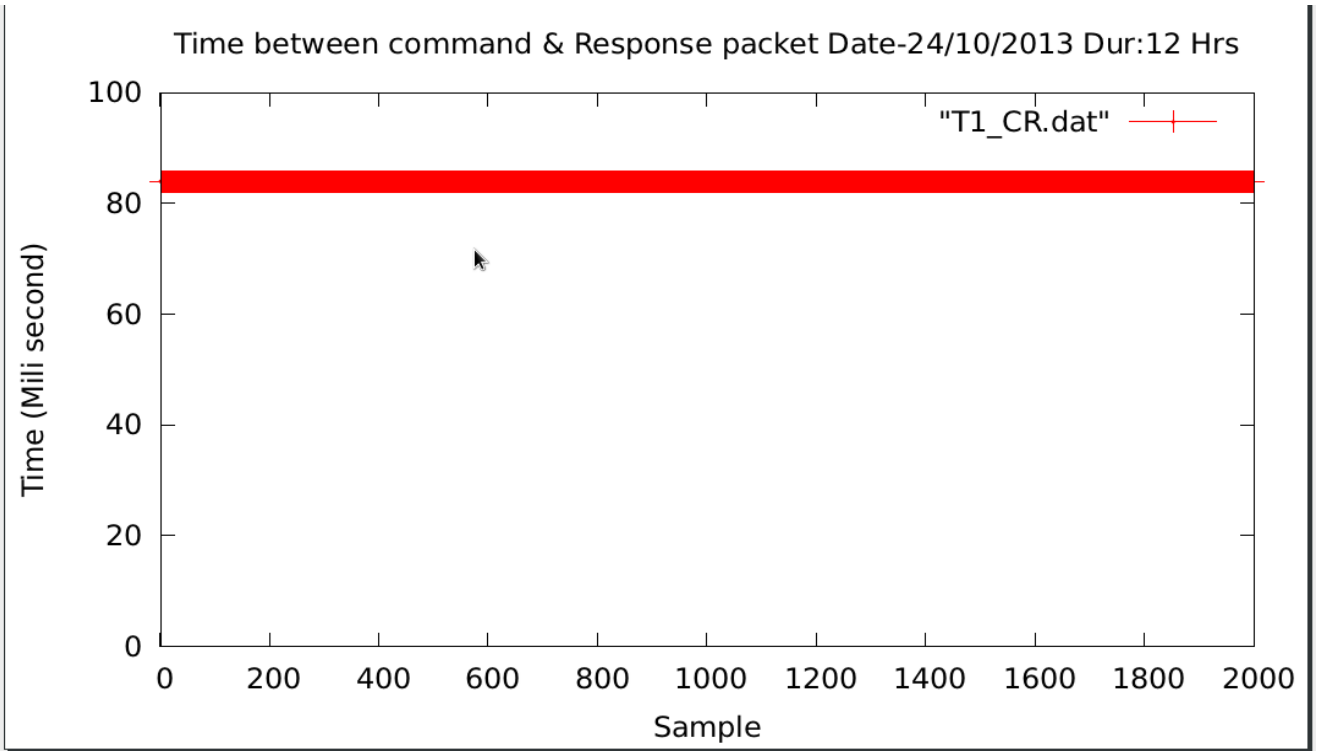


Fig(4) : Time between command & response in 1 second interval

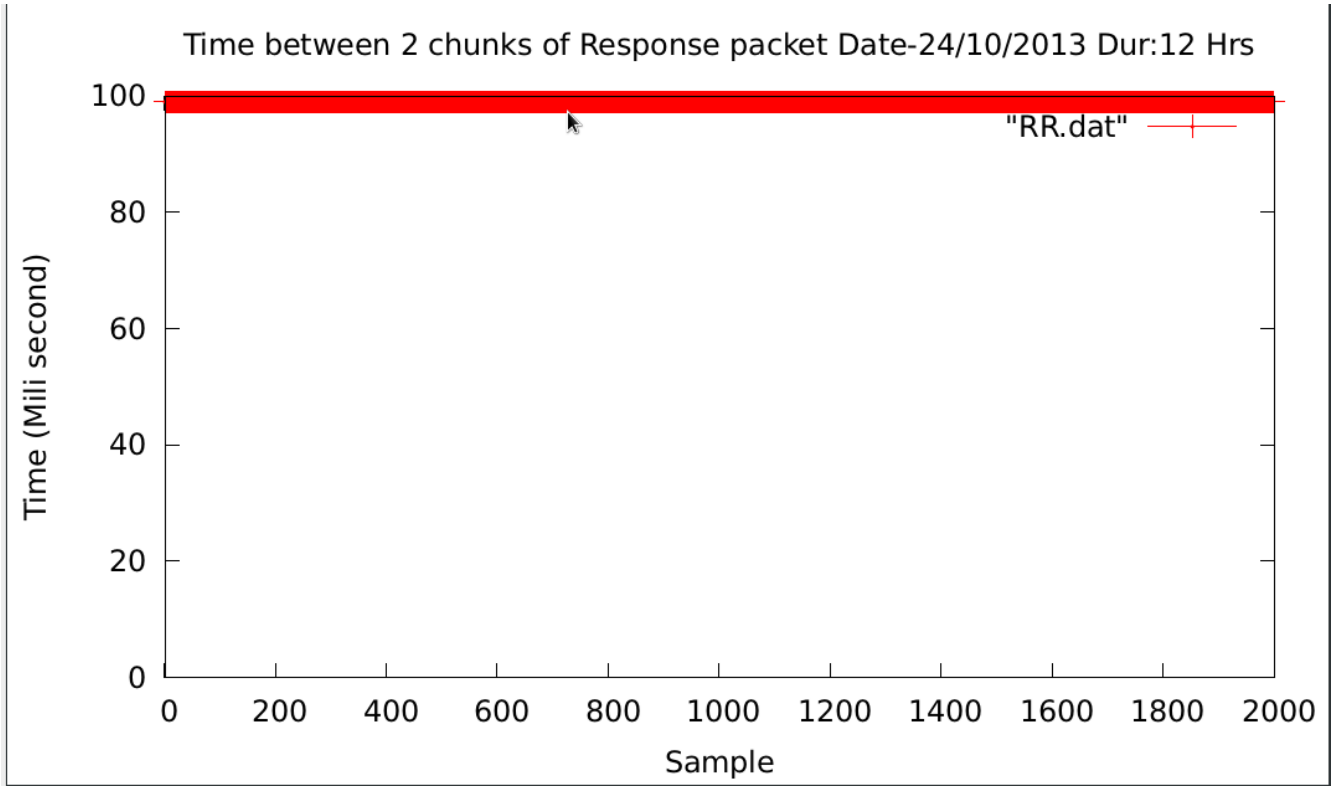


Fig(5) : Time between command & response in 1 second interval

Plots after changing the IP of server machine & sub-net :

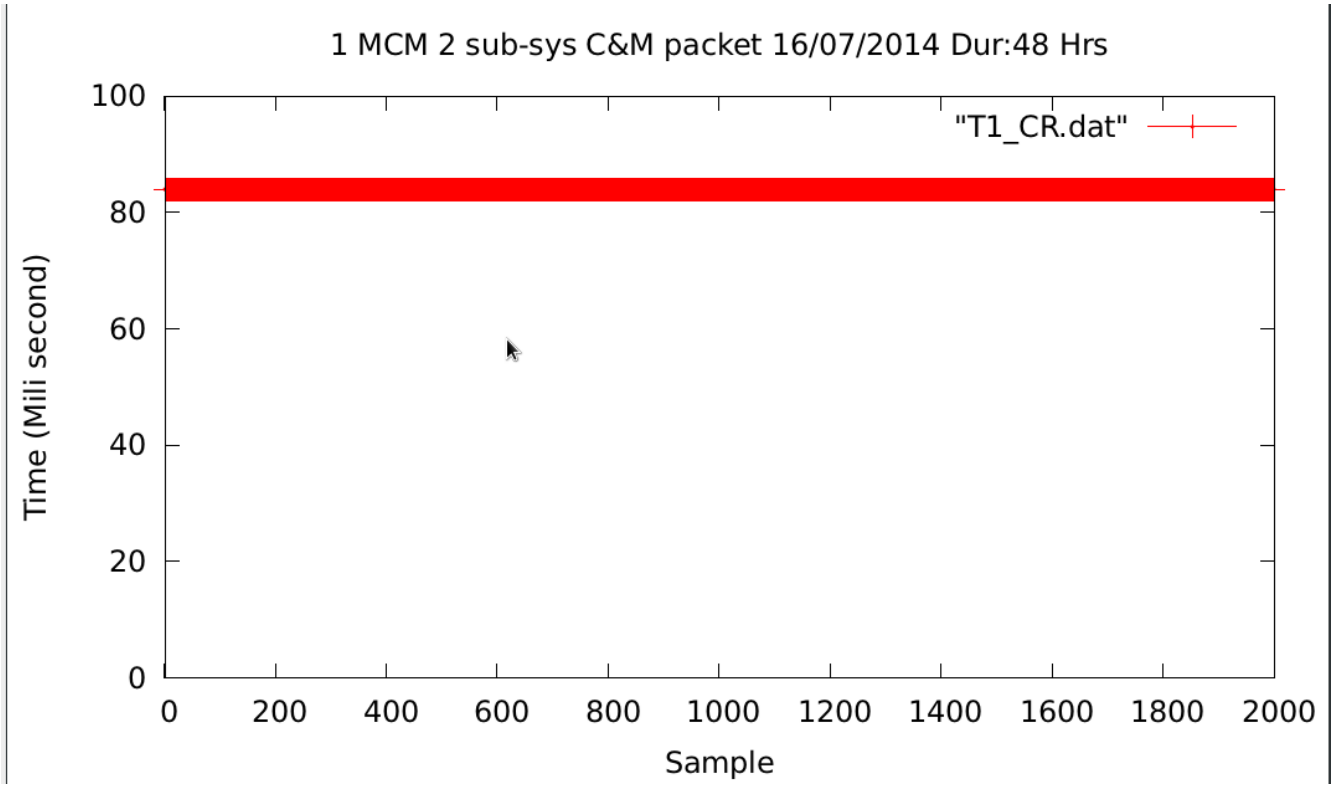


Fig(6) : Time between command-Response after changing IP & sub-net.

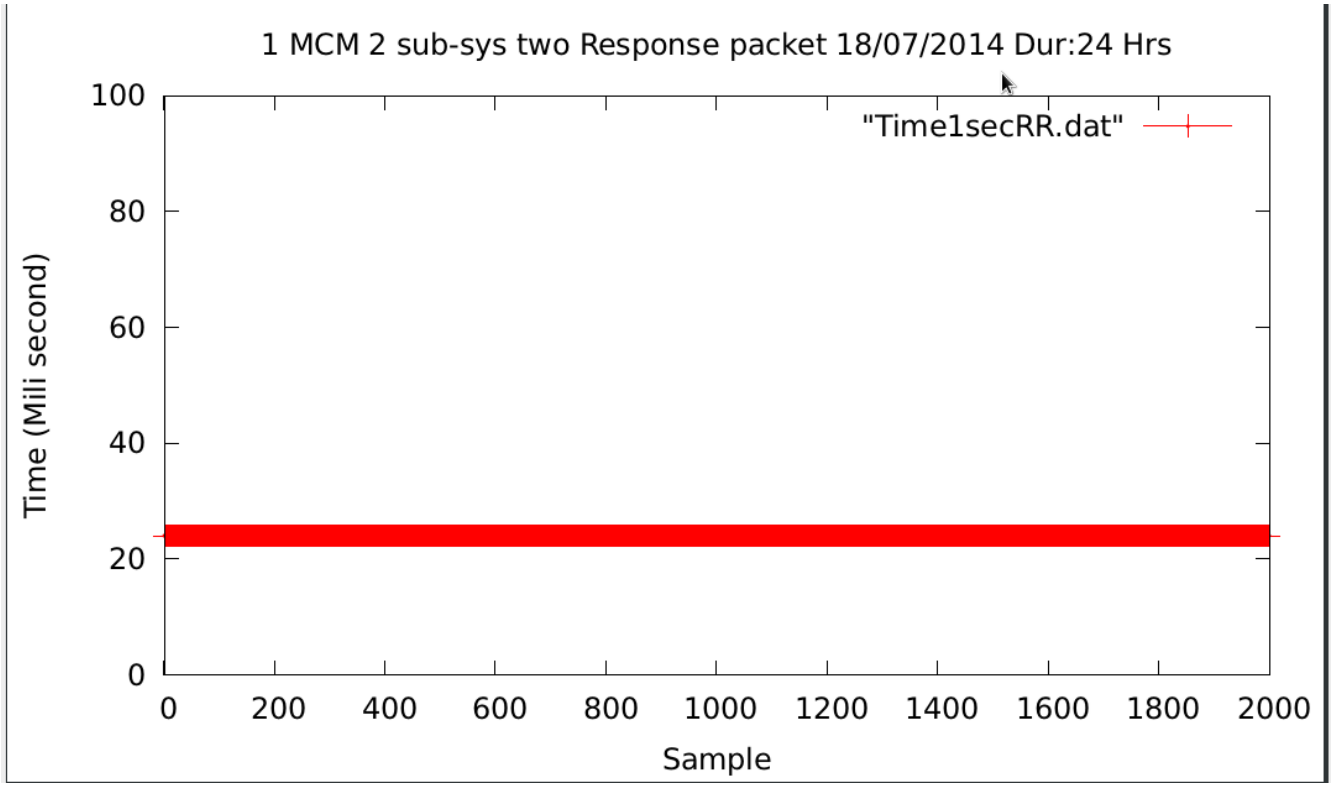


Fig(7) : Time between Two response packets after changing IP & sub-net.

Plots of round trip time when two sub-system configured in one MCM card :



Fig(8) : Time between C & R when two sub-system configured in one MCM card



Fig(9): Time between 2 Response when two sub-system configured in one MCM card

Plots above shows that when we were connecting Online_v2 with 8 antenna MCM cards using small 8 port unmanageable switch, Online_v2 had seen major drop out of response packets. Once we shifted our lab set up to a L3 network switch isolating MCM system sub-net with rest of GMRT sub-net, we observed zero drop out of response packets.

5.3 Antenna Testing of Online_v2 software :

- Four MCM cards were installed at *C03,C06,S04 & S02*. They were communicating very well with Online_v2, sending monitoring response every 1 second time interval.

- C03 rabbit MCM card was connected with a temperature sensor which was sending Antenna shell temperature every 1 second.

- We then moved all MCM cards to central square antenna for out testing. We installed MCM cards at *C01,C03,C04,C06 and C09.* All MCM were working fine, communicating with Online_v2 without any problem.

- After testing Online_v2 software with individual MCM cards, we tested Online_v2 core software during April-2014 & October-2014 maintenance period.

- Successfully tested two antenna sub-system at C06 antenna during April-2014 :

  Date of Testing : 4th April 2014
  Date of closed loop testing : 7th April 2014

During April-2014 MTAC period, we have successfully tested two antenna sub-systems at C06 antenna. We have installed HP make L2 Ethernet switch and two Rabbit MCM cards at C06 for control and monitor of Broadband OF system and Sentinel system.
1. Broadband OF system testing:
The first level control and monitoring of Broadband OF system has been tested by interfacing Rabbit MCM with OF system hardware. The command for RF attenuation was send from all three paths,
1.1. Python environment - ONLINE-V2 - Rabbit MCM - OF hardware.
1.2. GUI environment - ONLINE-V2 - Rabbit MCM - OF hardware.
1.3. Terminal environment - ONLINE-V2 - Rabbit MCM - OF hardware.

The attenuation value was set by sending command to Rabbit MCM card in range of 0 to 31 dB, in step size of 1 dB. The RF power was going down by 2dB for entire range of attenuation values (whatever values we set through Online_v2 must goes down in the OF system by 2 db). This test was done in telemetry lab as well as C06 antenna shell. The changes in attenuation were reflecting in RF power at antenna base (OF Tx output port) and receiver room OF system (OF Rx Mon port) on spectrum analyzer.

2. Sentinel System Testing:
The control port of Rabbit MCM for sentinel system was tested by connecting 32 bit LED test jig. The digital mask was send to Rabbit MCM by three paths mentioned above. The shell

temperature was monitored by connecting temperature sensor to channel 1 of MCM monitoring port. The temperature reading was displayed on ONLINE-V2 shared memory.

The OF system group was involved in setting up OF system and successful completion of testing.

- Successfully tested 3 antenna 2 sub-system at C01, C04 & C06 antenna

Date of Testing : 8th October 2014 – 11th October 2014- **Online_v2 Team members**

During October-2014 MTAC period, we have successfully tested three antenna two sub-systems at C01,C04 and C06 antenna. We have installed CISCO & HP make L2 Ethernet switch and two Rabbit MCM cards at C01,C04 & C06 for control and monitor of Broadband OF system and Sentinel system.

1. Broadband OF system testing:
The first level control and monitoring of Broadband OF system has been tested by interfacing Rabbit MCM with OF system hardware. The command for RF attenuation was send from all three paths,
1.1. Python environment - ONLINE-V2 - Rabbit MCM - OF hardware.
1.2. GUI environment - ONLINE-V2 - Rabbit MCM - OF hardware.
1.3. Terminal environment - ONLINE-V2 - Rabbit MCM - OF hardware.

The attenuation value was set by sending command to Rabbit MCM card in range of 0 to 31 dB, in step size of 1 dB. The RF power was going down by 2dB for entire range of attenuation values. This test was done in telemetry lab as well as C06 antenna shell. The changes in attenuation were reflecting in RF power at antenna base (OF Tx output port) and receiver room OF system (OF Rx Mon port) on spectrum analyzer.

2. Sentinel System Testing:
The control port of Rabbit MCM for sentinel system was tested by connecting 32 bit LED test jig. The digital mask was send to Rabbit MCM by three paths mentioned above. The shell temperature was monitored by connecting temperature sensor to channel 1 of MCM monitoring port. The temperature reading was displayed on ONLINE-V2 shared memory.

The OF system group was involved in setting up OF system and successful completion of testing.

- ✗ Successfully tested 16 antenna 2 sub-system

Date of Testing : 7th April 2015 – 18th April 2015- **Online_v2 Team members**

During April-2015 MTAC period, we have successfully tested 16 antenna two sub-systems . We have installed CISCO & HP make L2 Ethernet switch and two Rabbit MCM cards for control and monitor of Broadband OF system and Sentinel system.

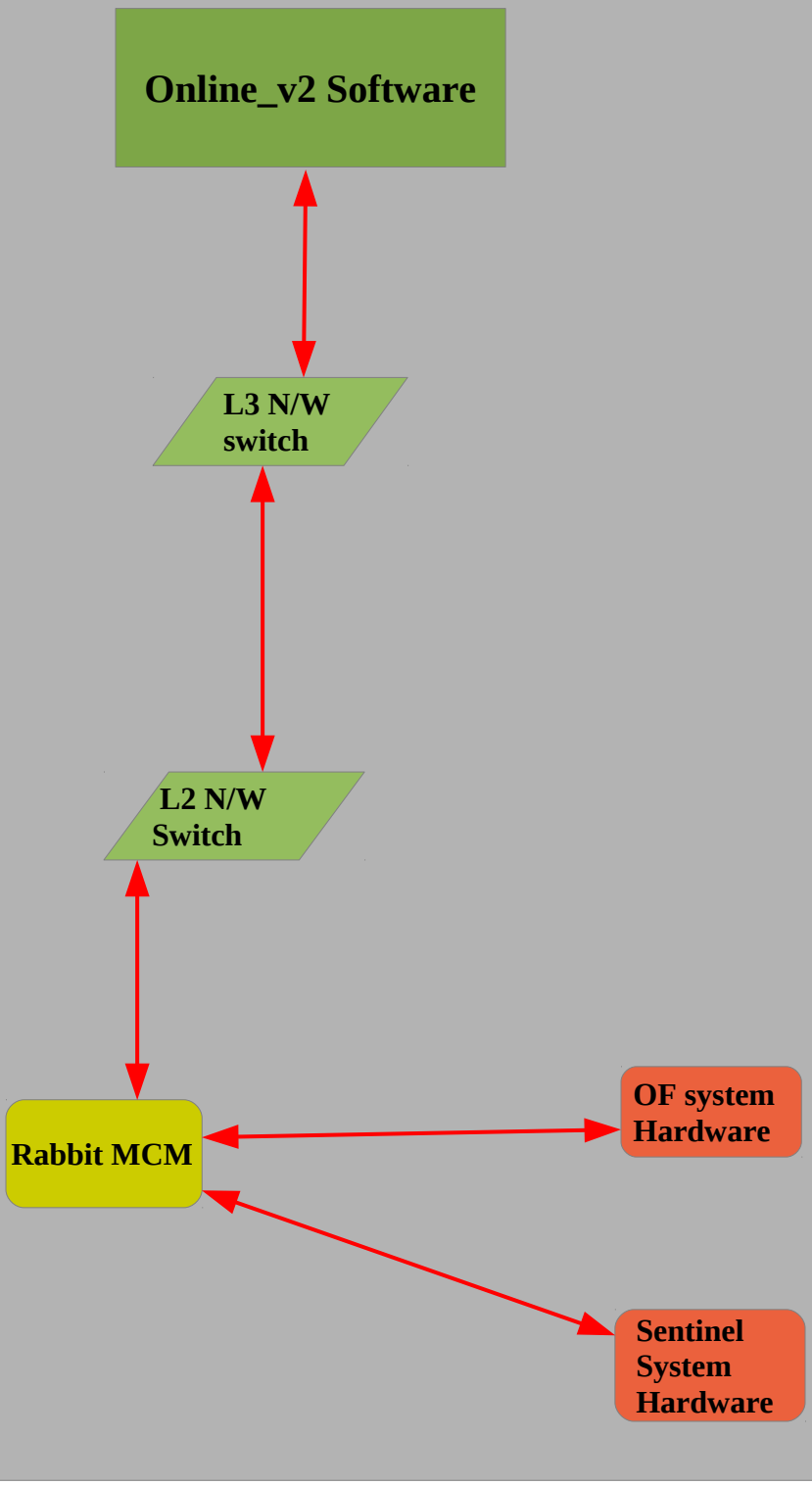
1. Broadband OF system testing:
The first level control and monitoring of Broadband OF system has been tested by interfacing Rabbit MCM with OF system hardware. The command for RF attenuation was send from all three paths,
1.1. Python environment - ONLINE-V2 - Rabbit MCM - OF hardware.
1.2. GUI environment - ONLINE-V2 - Rabbit MCM - OF hardware.
1.3. Terminal environment - ONLINE-V2 - Rabbit MCM - OF hardware.

The attenuation value was set by sending command to Rabbit MCM card in range of 0 to 31 dB, in step size of 1 dB. The RF power was going down by 2dB for entire range of attenuation values. This test was done in telemetry lab as well as C06 antenna shell. The changes in attenuation were reflecting in RF power at antenna base (OF Tx output port) and receiver room OF system (OF Rx Mon port) on spectrum analyzer.

2. Sentinel System Testing:
The control port of Rabbit MCM for sentinel system was tested by connecting 32 bit LED test jig. The digital mask was send to Rabbit MCM by three paths mentioned above. The shell temperature was monitored by connecting temperature sensor to channel 1 of MCM monitoring port. The temperature reading was displayed on ONLINE-V2 shared memory.

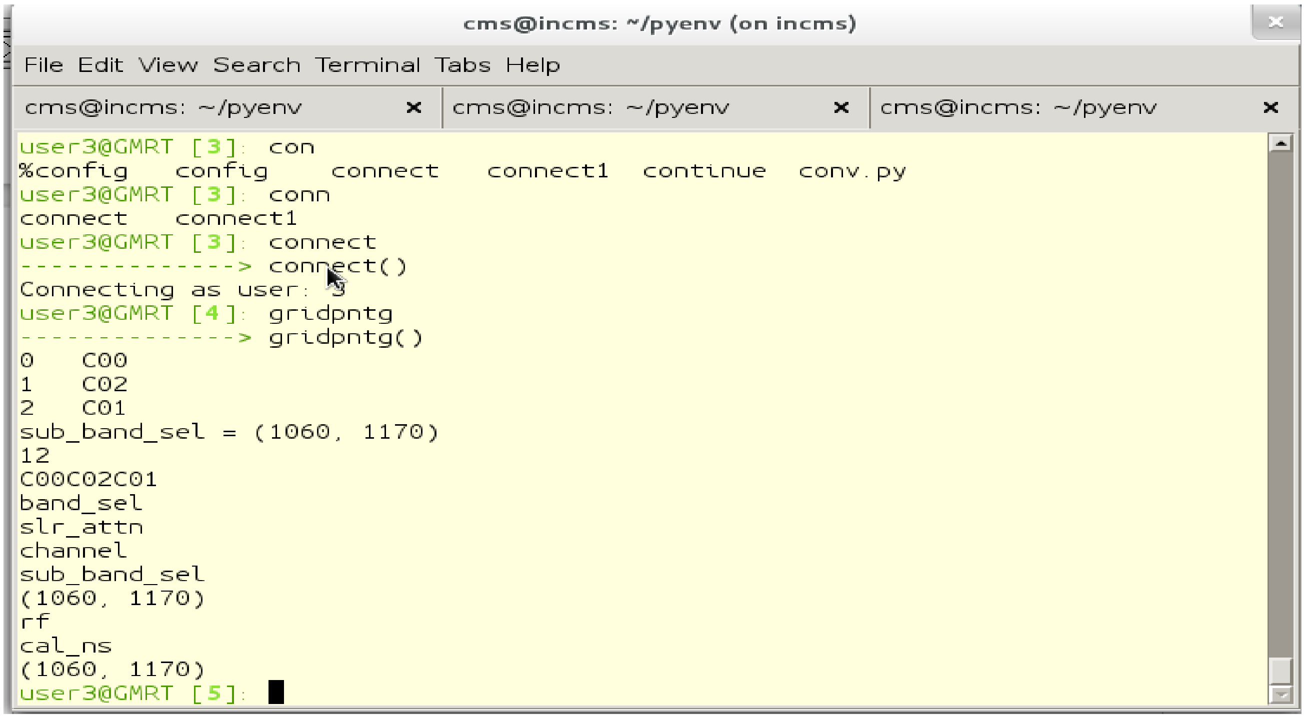
The OF system group was involved in setting up OF system and successful completion of testing. Kindly refer to the below figure for testing setup:

Testing Setup :



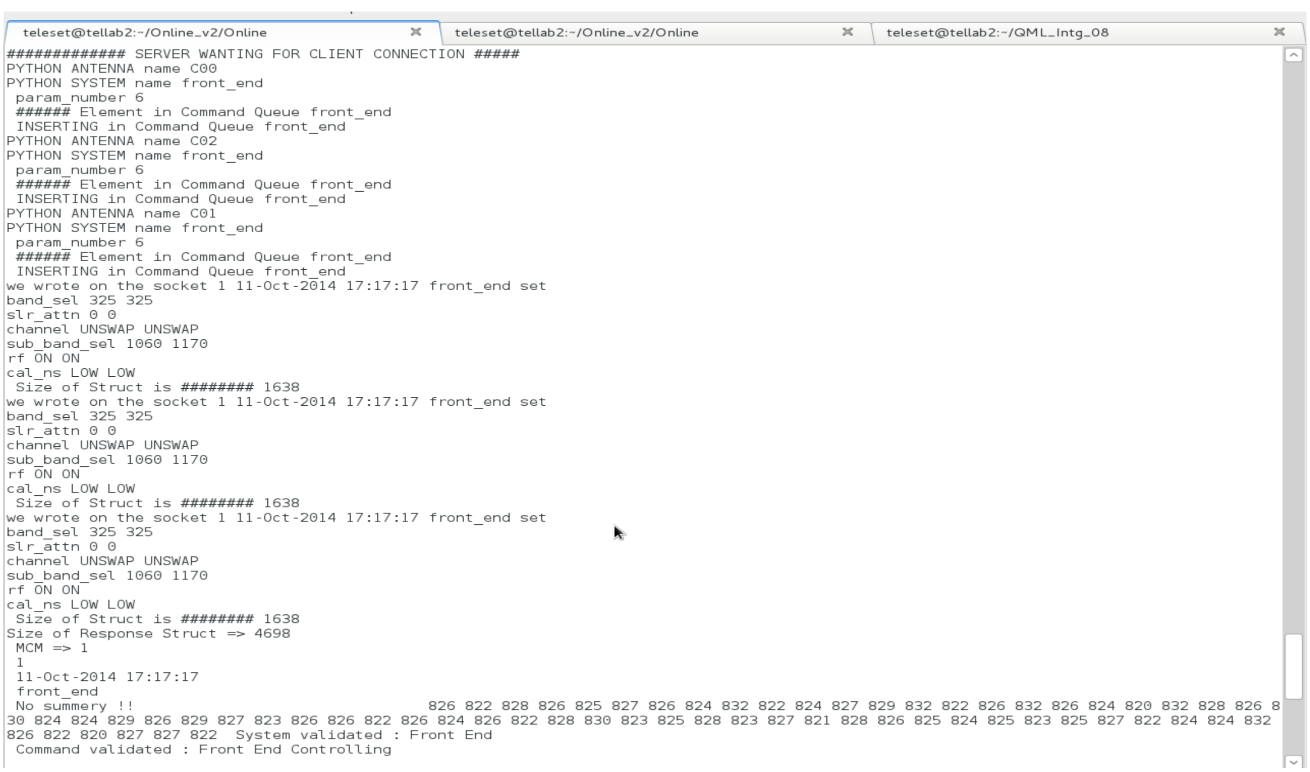
5.4 Snapshot of the Online_v2-Python_Environment-MCM testing :

Python Environment user 2 :



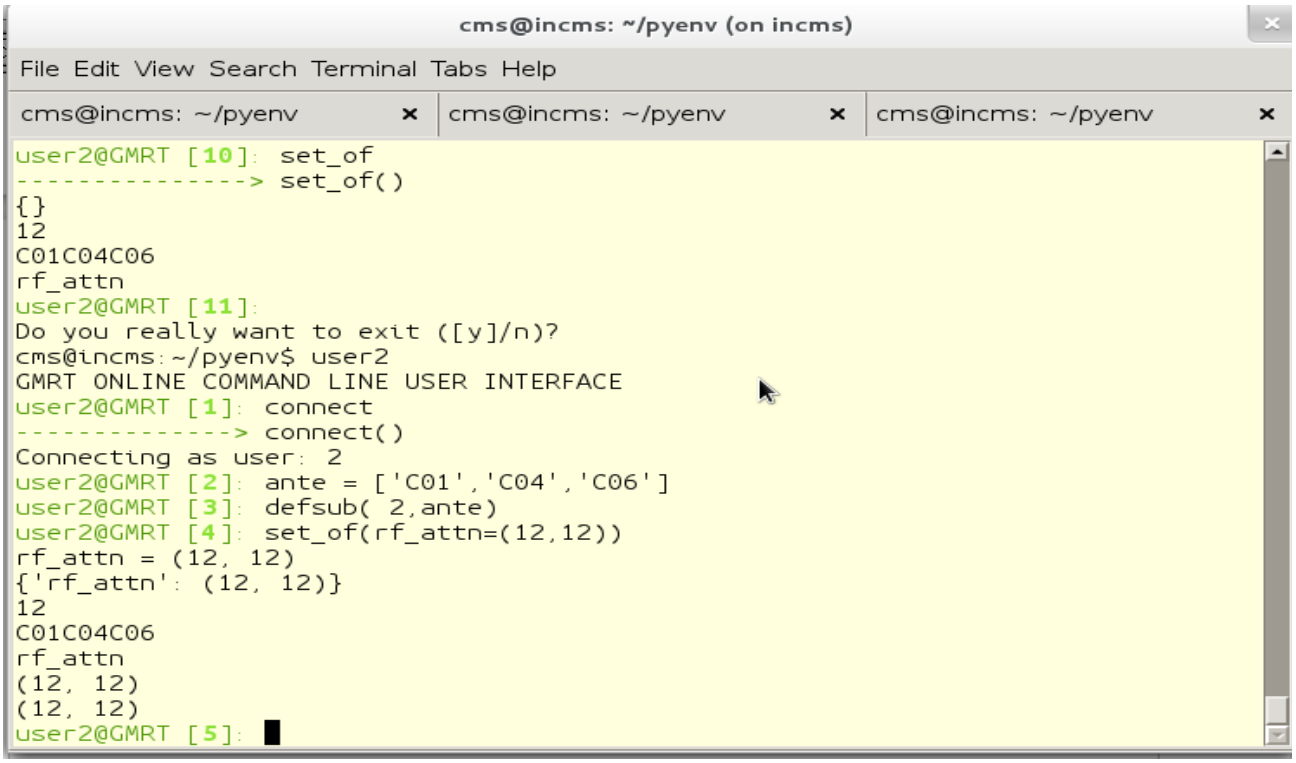
```
cms@incms: ~/pyenv (on incms)
File Edit View Search Terminal Tabs Help
cms@incms: ~/pyenv    cms@incms: ~/pyenv    cms@incms: ~/pyenv
user3@GMRT [3]: con
%config   config    connect    connect1   continue   conv.py
user3@GMRT [3]: conn
connect   connect1
user3@GMRT [3]: connect
---------------> connect()
Connecting as user: 3
user3@GMRT [4]: gridpntg
---------------> gridpntg()
0   C00
1   C02
2   C01
sub_band_sel = (1060, 1170)
12
C00C02C01
band_sel
slr_attn
channel
sub_band_sel
(1060, 1170)
rf
cal_ns
(1060, 1170)
user3@GMRT [5]:
```

Output of Online_V2 screen :

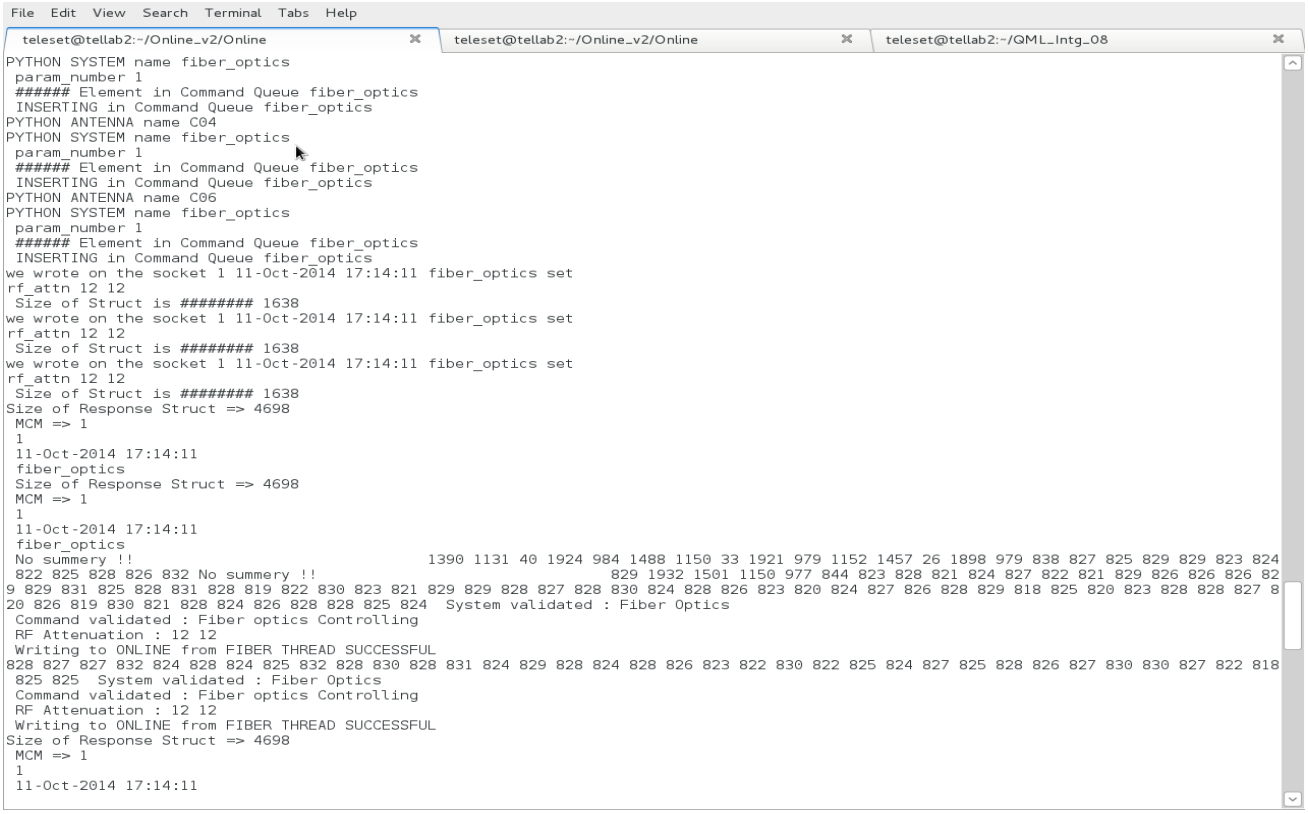


```
teleset@tellab2:~/Online_v2/Online    teleset@tellab2:~/Online_v2/Online    teleset@tellab2:~/QML_Intg_08
############# SERVER WANTING FOR CLIENT CONNECTION #####
PYTHON ANTENNA name C00
PYTHON SYSTEM name front_end
 param_number 6
 ###### Element in Command Queue front_end
 INSERTING in Command Queue front_end
PYTHON ANTENNA name C02
PYTHON SYSTEM name front_end
 param_number 6
 ###### Element in Command Queue front_end
 INSERTING in Command Queue front_end
PYTHON ANTENNA name C01
PYTHON SYSTEM name front_end
 param_number 6
 ###### Element in Command Queue front_end
 INSERTING in Command Queue front_end
we wrote on the socket 1 11-Oct-2014 17:17:17 front_end set
band_sel 325 325
slr_attn 0 0
channel UNSWAP UNSWAP
sub_band_sel 1060 1170
rf ON ON
cal_ns LOW LOW
 Size of Struct is ######## 1638
we wrote on the socket 1 11-Oct-2014 17:17:17 front_end set
band_sel 325 325
slr_attn 0 0
channel UNSWAP UNSWAP
sub_band_sel 1060 1170
rf ON ON
cal_ns LOW LOW
 Size of Struct is ######## 1638
we wrote on the socket 1 11-Oct-2014 17:17:17 front_end set
band_sel 325 325
slr_attn 0 0
channel UNSWAP UNSWAP
sub_band_sel 1060 1170
rf ON ON
cal_ns LOW LOW
 Size of Struct is ######## 1638
Size of Response Struct => 4698
 MCM => 1
 1
 11-Oct-2014 17:17:17
 front_end
 No summery !!                     826 822 828 826 825 827 826 824 832 822 824 827 829 832 822 826 832 826 824 820 832 828 826 8
30 824 824 829 826 829 827 823 826 826 822 826 824 826 822 828 830 823 825 828 823 827 821 828 826 825 824 825 823 825 827 822 824 824 832
826 822 820 827 827 822  System validated : Front End
 Command validated : Front End Controlling
```

Python Environment user 3 : Procedure from User3

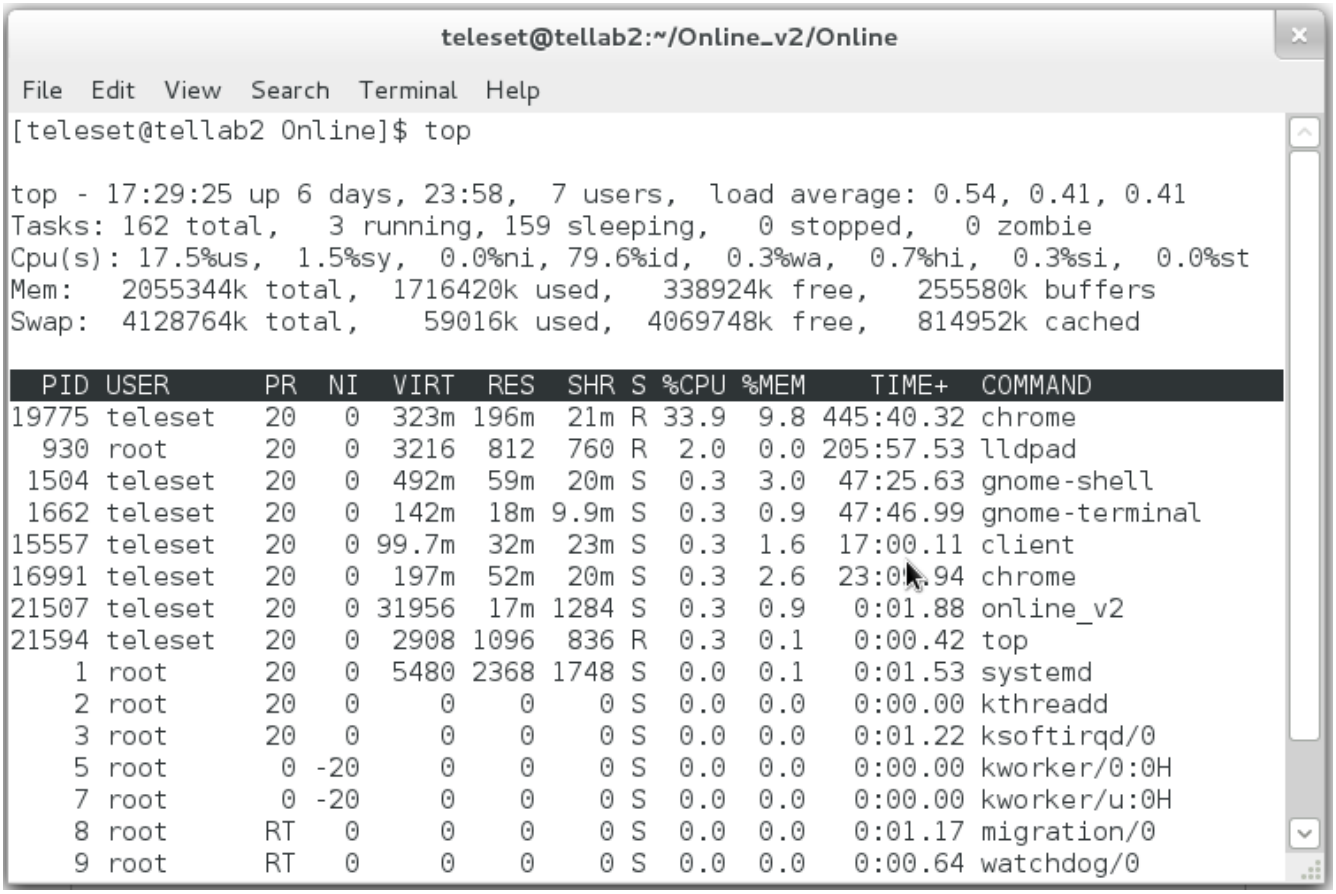


Online_V2 output Screen :



5.5 TOP Output :

With around 15 antenna sub-system connected to Online_v2, TOP output shows that Online_V2 program uses only 0.3% of CPU and 0.9% of Memory.



=> We also tested the stability and robustness of Online_v2 and Python environment by giving commands from user2 and user3 at a interval of 0.1 and 0.5 seconds. Both programs were able to executed commands without any problem.

5.6 Successfully tested GAB and Front end system in Lab :

Date of Testing : 5th September 2014 to 23rd September 2014

During September month, we have successfully tested GAB & FE sub-systems in respective Labs. The command for FE & GAB setting was sent from :

1. GUI environment – ONLINE_V2 – Rabbit MCM – FE / GAB hardware.
2. Terminal environment – ONLINE_V2 – Rabbit MCM – FE / GAB hardware.

- Front End system testing : The first level control of Front end system has been tested by interfacing Rabbit MCM with front end system hardware in FE lab. The control setting values were set by sending command to Rabbit MCM card. All changes were reflected on spectrum analyzer.

Commands tested for Front end system :

1. Frequency band	50-1420 MHZ	50-1420 MHZ
2. Solar attenuation	0 db- terminate	0 db-terminate
3. Channel	swap/unswap	swap/unswap
4. Sub band selection	subband1-subband4	subband1-subband4
5. RF	On/Off	On/Off
6. calibrator noise	Low/High/Med/ExHigh	Low/High/Med/ExHigh

- GAB System Testing : The first level control GAB system has been tested by interfacing Rabbit MCM with GAB system hardware in ABR lab. The control setting values were set by sending command to Rabbit MCM card. All changes were reflected on spectrum analyzer. The FE and GAB system group was involved in setting up GAB & FE system and successful completion of testing.

Commands tested for GAB system :

1. Reference LO	10-105 MHZ	10-105 MHZ
2. LO	600-17000 KHZ	600-17000 KHZ
3. Attenuation	10 db	10 db
4. Filter	8	8
5. LPF	0	0
6. Source	Siggen	Synthesizer
7. Signal	Antenna	Noise
8. Path	Direct	Mixer
9. Channel	1	2

5.7 Online_V2 interfacing with GWB correlator via GPU DAS server :

Test done on 23/04/2014 Time : 10 AM to 11 AM

Online_V2 & GPU Dasserver executed on oper2 (IP 192.168.1.14 machine).

********** Online_V2 Terminal display **********

```
[observer@oper2 Online]$ ./online_v2

msgget: Calling msgget(0xc9,01600)
msgget: msgget succeeded: msqid = 0
Successfully Created MESSAGE QUEUE ID=0
############### SERVER WANTING FOR CLIENT CONNECTION ######
==> SERVO SERVER WANTING FOR CLIENT CONNECTION ===>### SERVER WANTING
FOR GUI CLIENT TO CONNECT ######
$$$ SERVER WANTING FOR PYTHON ENVIRONMENT CLIENT TO CONNECT $$$$

 >> dasinit

 CMD[0] => dasinit
DAS INIT
Message Sent=>0 0 das init 1 7FFFFFFE F /home/observer/Online_v2/gpu.hdr
waiting for ack id 2
ACK Got from DASSERVER :  ack
SUCCEDED
```

Online_v2–Servo test with actual hardware in SSL lab :

Date : 22/12/2014 Time : 11.40 AM to 12.30 PM

Test done by : Mr. Thiyagu & Mr. Raju

PC104 Card with servo program kept in SSL lab with actual hardware connected.

PC104 card IP : 192.168.8.42

Online_v2 server IP : 192.168.8.45

############### SERVER WANTING FOR CLIENT CONNECTION ######
ACCEPTED CONNECTION C00 Servo System 192.168.8.42

SERVO thread opened succesfully=> 0

C00 servo hold // commands issued from Online_v2 terminal

CMD[0] => C00
CMD[1] => servo
CMD[2] => hold
Command for C00 ANTENNA
ANTENNA C00 C00
System servo
OP NAME hold
********** hold
####### Element in Command Queue servo
we wrote on the socket 12 servo hold
ax B
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 12 22-Dec-2014 12:03:03 servo hold
ax B
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 12
Timestamp is 22-Dec-2014 12:03:03
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 12
Timestamp is 22-Dec-2014 12:03:03
System name is servo
Response code is 2
Response type is 1
Servo Final Resp: Command SUCCESS
Servo Final Resp: Command SUCCESS
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

5.8 Servo-Online_V2 successful interface testing at C01 antenna

Date : 29/12/2014
Time : 14.00 PM to 15.00 PM
Persons : Mr. Thiyagu & Mr.Raju

[raju@localhost Online]$ ./online_v2
HIGHUSER thread CREATED=> 0
SERVO thread CREATED=> 0
GUI INTERFACE thread CREATED=> 0
PYTHON INTERFACE thread CREATED=> 0
SERVER WANTING FOR GUI CLIENT TO CONNECT
==> SERVO SERVER WANTING FOR CLIENT CONNECTION ===>$$$ SERVER WANTING FOR PYTHON ENVIRONMENT CLIENT TO CONNECT $$$$
MCM SYSTEM thread CREATED=> 0

msgget: Calling msgget(0xc9,01600)
msgget: msgget succeeded: msqid = 0
Sucessfully Created MESSAGE QUEUE ID=0
############### SERVER WANTING FOR CLIENT CONNECTION #####

>> **ACCEPTED CONNECTION C01 Servo System 192.168.4.3**
SERVO thread opened succesfully=>
C01 servo track // commands issued from Online_v2 terminal

CMD[0] => C01
CMD[1] => servo
CMD[2] => track
Command for C01 ANTENNA
ANTENNA C01 C01
System servo
OP NAME track
********** track
####### Element in Command Queue servo

>> we wrote on the socket 11 29-Dec-2014 14:35:20 servo track
ax B
time 16:00:00
ang1 115:00:00
ang2 90:00:00
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 11

Timestamp is 29-Dec-2014 14:35:20
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 11
Timestamp is 29-Dec-2014 14:35:20
System name is servo
Response code is 2
Response type is 1
Servo Final Resp: Command SUCCESS
Servo Final Resp: Command SUCCESS
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

5.9 Online_v2– FPS system testing over rabbit serial link in C06 antenna

Date : 20/03/2015 Time : 11.45 Am to 13.10 PM

Test Done by: Charu Kanade, Abhay Bhumkar, Mahadev Misal & Raju Uprade

Rabbit card with Device IP 192.168.21.107 connected to FPS system over serial link

Online_v2 machine IP : 192.168.8.45

```
[teleset@tellab2 Online]$ ./online_v2
HIGHUSER thread CREATED=> 0
SERVO thread CREATED=> 0
GUI INTERFACE thread CREATED=> 0
PYTHON INTERFACE thread CREATED=> 0
MCM SYSTEM thread CREATED=> 0

msgget: Calling msgget(0xc9,01600)
msgget: msgget succeeded: msqid = 0
Sucessfully Created MESSAGE QUEUE ID=0
$$$ SERVER WANTING FOR PYTHON ENVIRONMENT CLIENT TO CONNECT $$$$

 >>  ACCEPTED CONNECTION FROM FPS MCM DEVICE 192.168.21.107
FPS thread opened succesfully=> 0
############### SERVER WANTING FOR CLIENT CONNECTION ######

C06 fps reboot                    // Command from Online_V2 terminal

 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => reboot
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  reboot
we wrote on the socket 35  fps reboot
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 35 20-Mar-2015 12:16:35 fps reboot
 Size of Struct is ######### 1638
```

Size of Response Struct => 4698
 MCM => 1
 35
 20-Mar-2015 12:16:35
 fps
  ####### NUmber of RESPONSE MSG is 1
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Reboot
 Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps run_to_cal // Command from Online_V2 terminal

 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => run_to_cal
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME run_to_cal
we wrote on the socket 30 fps run_to_cal
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 30 20-Mar-2015 12:16:51 fps run_to_cal
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 30
 20-Mar-2015 12:16:51
 fps
  ####### NUmber of RESPONSE MSG is 1
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Run to calibrate

5.10 . Online_v2- 16 antennae (OF+Sentinel system) testing during April 2015 MTAC

Description : Two sub-system (OF + Sentinel were configured in one MCM card)

Date : Full MTAC (2/4/2015 to 22/4/2015)

Test Done by: Online_v2 team + OF team.

ANTENNA DEVICE COMMUNICATION is BROKEN
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.115
E02 thread opened succesfully=> 0
############# SERVER WANTING FOR CLIENT CONNECTION #####
E02 sentinel set

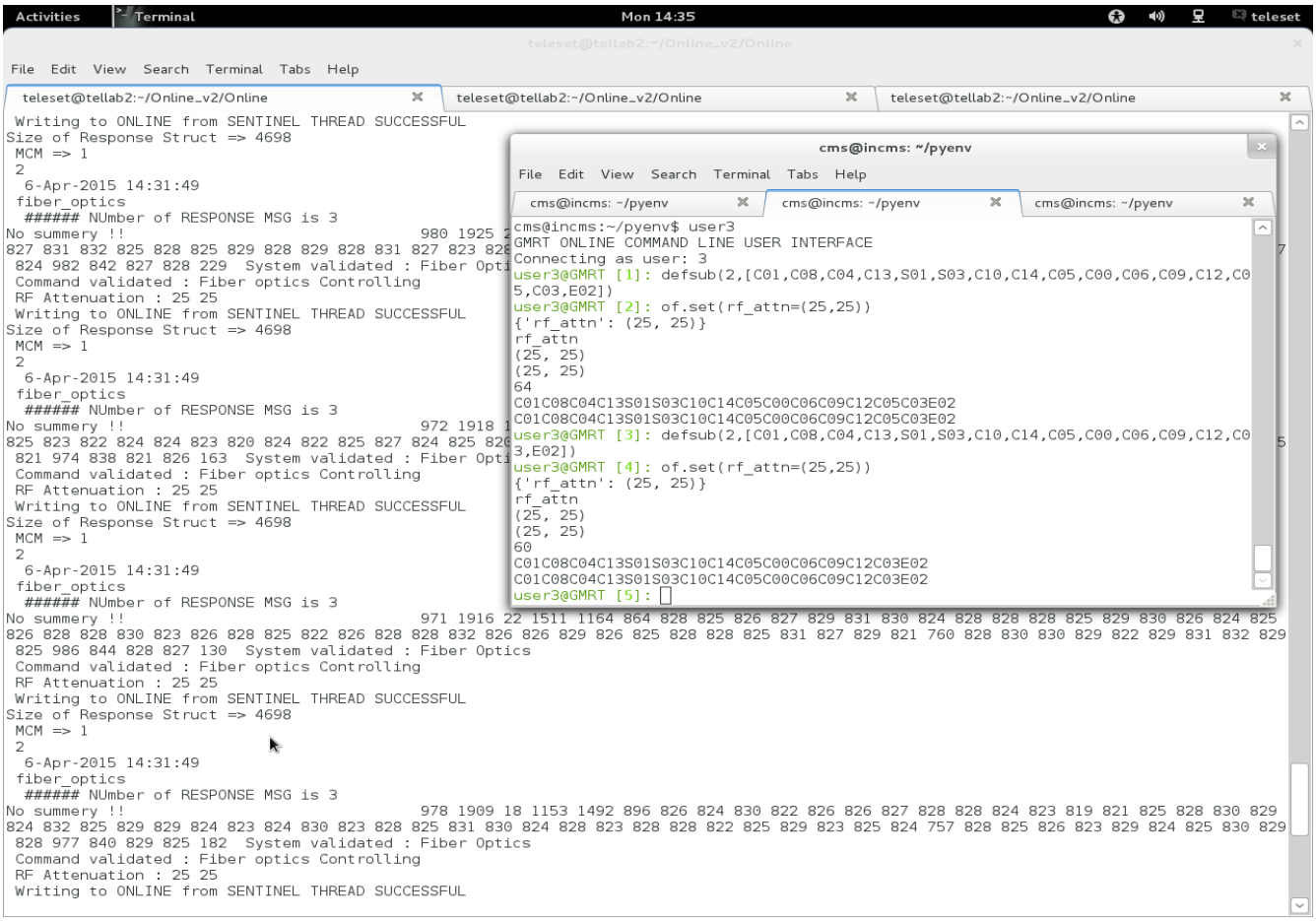
CMD[0] => E02
CMD[1] => sentinel
CMD[2] => set
E02 Antenna
Element in Command Queue sentinel
INSERTING in Command Queue sentinel

>> we wrote on the socket 11 3-Apr-2015 11:17:46 sentinel set
dmask ffff 5555
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
11
3-Apr-2015 11:17:46
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 810 864 856 854 794 812 843 842 832 841 837 829 852 832 840 836 842 832 829 820 819 832 813 835 816 820 818 831 834 836 830 837 844 836 839 832 838 832 829 819 820 818 805 823 823 830 822 832 274 387 381 257 428 769 804 609 289 382 609 675 498 336 781 998 System validated : Sentinel
Command validated : Sentinal Controlling
Digital Mask : ffff 5555
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
E02 fiber_optics mon

CMD[0] => E02
CMD[1] => fiber_optics
CMD[2] => mon
E02 Antenna
Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics

>> we wrote on the socket 10 3-Apr-2015 11:18:00 fiber_optics mon

Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
10
3-Apr-2015 11:18:00
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 823 862 829 839 827 810 833 807 818 807 808 838 811 819 828 828 816 821 830 829 830 820 841 835 835 831 824 830 820 823 829 820 825 837 822 816 823 844 828 837 833 832 834 834 839 820 816 818 322 300 425 595 857 841 358 459 538 352 448 821 518 821 728 864 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 0.55
Voltage(-5.0V) : 1.00
Voltage(-3.1V) : 0.73
Voltage(-1.1V) : 0.77
Voltage(GND) : 0.61
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL



6. Summary :

1. Online_v2 core software has been successfully tested with multiple sub-system, it has the potential to grow as a replacement for current online system.

2. Online_v2 has been interfaced with Python environment, QML/QT GUI and readline based terminal, which makes Online_v2 suitable for automated operation of GMRT telescope.

3. Different modules of Online_v2 are under testing in the Telemetry Lab for an year. The system has been running without any glitches.

4. Online_v2 has been installed in GMRT correlator Lab where *Mr. Harshwardhan Reddy* is using it for his GWB correlator Lab testing.

5. Online_v2 has been successfully tested with Servo system PC104 over Ethernet communication at Servo Lab as well as C01 antenna.

6. Online_v2 has been successfully tested with FPS system over Rabbit MCM card serial link in FPS lab as well as in C06 antenna.

7. Online_v2 has been successfully tested with 16 antennae (OF + Sentinel system) during April 2015 MTAC.

8. Online_v2 has been developed In house, so it will be very easy to change, maintain & upgrade the Online_v2 software.

9. During April MTAC 2015, MCM monitoring data from all 16 MCM/antennas were logged in the database from shared memory.

10. Online_v2 has been successfully interfaced with GWB,GAB,Of,Sentinel & servo system.

7. References :

1. **http://vichargrave.com (Multi Threaded Server design)**

2. Online_V2 - An Upgraded Control-Monitor Software for GMRT MwSky Poster : Authors N.G.Katharia, R.Uprade, S.N.Katore, N.M.Sisodiya, S.Sherkar, D.Bhong, C.Kanade, S.Nayak (link address http://www.ncra.tifr.res.in:8081/~ngk/)

3. TELESET-ABCCOM software Technical report by Laurent Pommier Date 05/01/2006.

Appendix :

Structure used for Online_v2 –DAS server interfacing :

Msgtag structure is declared in display.h in present Online code which is used by Online to pass DAS command DAS server :

display.h

```
typedef struct
{
        long type;
        char buff[256];
} msgtag;
```

DAS Server side :

Message queue structure

```
typedef struct
{
        long msglevel;
        protocolType prot;
        char buf[MsgSz];
} MsgBufType;
typedef unsigned char  UNC;
typedef unsigned short UNS;
typedef short        SHRT;
typedef unsigned long  UNL;
typedef unsigned int   UNL;

typedef struct ProtTag
{
  UNC  origin;              /* MSG Originating station */
  UNC  dest;                /* Destination station */
  UNC  type;              /* The type of req, eg. Sh_com,RPC,FileIO,DAS,etc.*/
  UNC  cmd;               /* Actual service name or shell command. */
  UNC  arg[4];    /*Used as a parameter of any type that can be coded in 4 bytes. */
  UNC  id;        /* Request no, from the origin, wraps at 2^8 */
  UNC  flag;          /* Flags of queries or replies */
  UNS  usrflag;    /* Any short val or set of flags for user */
  UNL  len;       /* Length of the packet that follows */
```

```
} ProtocolType;
```

online.h structure used in writing DAS project file & scan file :

```
typedef struct {
        float i;
        float q;
        float u;
        float v;
}  POL_FLUX;


typedef struct {
  char object[32];
  POL_FLUX flux;
  double mjd0;
  double ra_app, dec_app, ra_date, dec_date, dra,ddec; /* rad, rad/s */
  double freq[2];
  double first_lo[2];
  double bb_lo[2];
  double rest_freq[2];
  double lsrvel[2];  /* Hz, km/s  */
  double ch_width;  /* Hz */
  int id;
  int net_sign[4];
  int mode;
  int dum1;
  unsigned int antmask;
  unsigned short bandmask,dum2;
  short calcode, qual;
} SourceParType;                    // sizeof(SourceParType) to be written in scan.hdr.


typedef struct {
  char code[8];
  char observer[32];
  char title[32];
  unsigned int antmask;
  unsigned short bandmask,seq;
} ProjectType;                      // sizeof(ProjectType) to be written in Proj.hdr.

typedef struct {
```

```
  int status;
  float t;  /* program dependent meaning ! */
  ProjectType proj;
  SourceParType source;
} ScanInfoType;



typedef struct {
        char datafile[80];
        int  das_sub_state;
        int  cmd_no;
        ScanInfoType cntl;
}               DAS_SUB_CNTL;


typedef struct {
        char   config_file[80];
        int    msg_id;
        int    das_state;
        int    das_sub_state[MAX_DAS_SUB];
        DAS_SUB_CNTL sub_cntl[MAX_DAS_SUB];
}           DAS_CNTL;
```

Command – Response between Online_v2 & DAS server :

```
 >> dasinit

 CMD[0] => dasinit
DAS INIT
Message Sent=>0 0 das init 1 7FFFFFFE F /home/observer/Online_v2/gpu.hdr
waiting for ack id 2
ACK Got from DASSERVER :  ack
SUCCEDED

 >> addp

 CMD[0] => addp
ADD Project
Message Sent=>4 1 das addp 1 /home/observer/Online_v2/proj.hdr
waiting for ack id 2
```

```
ACK Got from DASSERVER :  ack
waiting for success is 2
waiting for success is 2
waiting for success is 2
SUCCEDED

 >> startscan

 CMD[0] => startscan
DAS START SCAN
Message Sent=>4 2 das start /home/observer/Online_v2/scan.hdr
waiting for ack id 2
ACK Got from DASSERVER :  ack
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
SUCCEDED

 >> stopscan

 CMD[0] => stopscan
DAS STOP SCAN
Message Sent=>4 3 das stop
waiting for ack id 2
ACK Got from DASSERVER :  ack
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
SUCCEDED

 >> startscan

 CMD[0] => startscan
DAS START SCAN
Message Sent=>4 2 das start /home/observer/Online_v2/scan.hdr
waiting for ack id 2
ACK Got from DASSERVER :  ack
waiting for success is 2
waiting for success is 2
waiting for success is 2
```

```
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
SUCCEDED
 >> stopscan

 CMD[0] => stopscan
DAS STOP SCAN
Message Sent=>4 3 das stop
waiting for ack id 2
ACK Got from DASSERVER :  ack
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
SUCCEDED

 >> startscan

 CMD[0] => startscan
DAS START SCAN
Message Sent=>4 2 das start /home/observer/Online_v2/scan.hdr
waiting for ack id 2
ACK Got from DASSERVER :  ack
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
SUCCEDED

 >> stopscan

 CMD[0] => stopscan
DAS STOP SCAN
Message Sent=>4 3 das stop
waiting for ack id 2
ACK Got from DASSERVER :  ack
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
```

```
waiting for success is 2
SUCCEDED

 >> startscan
CMD[0] => startscan
DAS START SCAN
Message Sent=>4 2 das start /home/observer/Online_v2/scan.hdr
waiting for ack id 2
ACK Got from DASSERVER :  ack
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
waiting for success is 2
SUCCEDED

 **********GPU DASSERVER TERMINAL DISPLAY ********

[observer@oper2 dassrv-gpu]$ ./dassrv
>> Reading configuration from /home/observer/corrsel.hdr
###### INFO CorrBandMask 0[3] 1[3] 2[3] 3[3]
DPCMODE=UsbPolar Corr0Mask=3 Corr1Mask=3 Corr2Mask=3
All the stale messages cleared
Cmd Recvd: 0 0 das init 1 7FFFFFFE F /home/observer/Online_v2/gpu.hdr
 During Assignment PTAB->Corr_id is 0
#### 1 7ffffffe f /home/observer/Online_v2/gpu.hdr
MAP MASK O_ANTMASK =>7ffffffe
MAP MASK D_ANTMASK =>3fffffff
got 1328 bytes from /home/observer/Online_v2/gpu.hdr
got 1328 bytes from /home/observer/Online_v2/gpu.hdr
got 1328 bytes from /home/observer/Online_v2/gpu.hdr
got 1328 bytes from /home/observer/Online_v2/gpu.hdr
To CorrA: AcqNode, CmdDest=0, CmdId=0, CmdType=DAS, Cmd=init, param=1 7FFFFFFE
F /home/observer/Online_v2/gpu.hdr
CorrId=1
CorrId=1
open 1 Corr CorrA Port 6001
Waiting for CorrA to acknowledge.....   OK! Got 16
13 F=16 S=8
'SUCCEDED' sent to ONLINE
Cmd Recvd: 4 1 das addp 1 /home/observer/Online_v2/proj.hdr
 During Assignment PTAB->Corr_id is 0
got 80 bytes from /home/observer/Online_v2/proj.hdr
```

```
MAP MASK O_ANTMASK =>7ffffffe
MAP MASK D_ANTMASK =>3fffffff
CorrA :AddProjReq for TEST 3fffffff 3 3
IN ADDPROJECT PTAB->Corr_id is 0
To CorrA: AcqNode, CmdDest=1, CmdId=0, CmdType=DAS, Cmd=addp, param=1
/home/observer/Online_v2/proj.hdr
CorrId=1
13 F=16 S=8
'SUCCEDED' sent to ONLINE
IN Main ADDPROJECT PTAB->Corr_id is 1
Cmd Recvd: 4 2 das start /home/observer/Online_v2/scan.hdr
 During Assignment PTAB->Corr_id is 1
got 232 bytes from /home/observer/Online_v2/scan.hdr
MAP MASK O_ANTMASK =>7ffffffe
MAP MASK D_ANTMASK =>3fffffff
====> Source Antmask is 7ffffffe ====>d_antmask is 3fffffff
====> Source Antmask is 3fffffff ====>d_antmask is 3fffffff
CorrA : StartReq for Code TEST AntMask 3fffffff BandMask 3
To CorrA: AcqNode, CmdDest=2, CmdId=0, CmdType=DAS, Cmd=start,
param=/home/observer/Online_v2/scan.hdr
CorrId=1
13 F=16 S=8
'SUCCEDED' sent to ONLINE
Cmd Recvd: 4 3 das stop
 During Assignment PTAB->Corr_id is 1
CorrA : StopReq for Project TEST
To CorrA: AcqNode, CmdDest=3, CmdId=0, CmdType=DAS, Cmd=stop, param=
CorrId=1
13 F=16 S=8
'SUCCEDED' sent to ONLINE

Cmd Recvd: 0 4 das delp  1
 During Assignment PTAB->Corr_id is 0
SubArray 0: No Project or Bad CorrId
Error Number -1
'FAILED' sent to ONLINE
Cmd Recvd: 0  5 das fini  1
 During Assignment PTAB->Corr_id is 0
To CorrA: AcqNode, CmdDest=5, CmdId=0, CmdType=DAS, Cmd=fini, param=1
CorrId=1
13 F=16 S=8
'SUCCEDED' sent to ONLINE
```

Command – Response between Online_v2 & PC04 servo system at Jogshed Lab :

C00 servo track // commands issued from Online_v2 terminal

```
 CMD[0] => C00
 CMD[1] => servo
 CMD[2] => track
Command for C00 ANTENNA
ANTENNA C00 C00
System  servo
OP NAME  track
********** track
 ####### Element in Command Queue servo
we wrote on the socket 11  servo track
ax B
time 15:30:00
ang1 180:00:00
ang2 19:00:00
 Size of Struct is ######### 1126
 INSERTING in Command Queue servo

 >> we wrote on the socket 11 22-Dec-2014 12:03:12 servo track
ax B
time 15:30:00
ang1 180:00:00
ang2 19:00:00
 Size of Struct is ######### 1126
 Size of Response Struct is ######### 2192
SEQ number is 11
Timestamp is 22-Dec-2014 12:03:12
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
 Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
 Size of Response Struct is ######### 2192
SEQ number is 11
Timestamp is 22-Dec-2014 12:03:12
System name is servo
Response code is 2
Response type is 2
```

Servo Final Resp: Command IRREVELENT ***// As elevation axis was not there***
Servo Final Resp: Command IRREVELENT
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C00 servo readanavar // commands issued from Online_v2 terminal

CMD[0] => C00
CMD[1] => servo
CMD[2] => readanavar
Command for C00 ANTENNA
ANTENNA C00 C00
System servo
OP NAME readanavar
readanavar
####### Element in Command Queue servo
we wrote on the socket 21 servo readanavar
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 21 22-Dec-2014 12:03:41 servo readanavar
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 21
Timestamp is 22-Dec-2014 12:03:41
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 21
Timestamp is 22-Dec-2014 12:03:41
System name is servo
Response code is 4
Response type is 1
Servo Analog Data
time
14:59:00
az_motor1_current
+0001.2695
az_motor2_current

+0001.2695
az_tacho1
+0001.1484
az_tacho2
+0001.1484
el_motor1_current
+0001.2695
el_motor2_current
+0001.2695
el_tacho1
+0001.1484
el_tacho2
+0001.1484
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C00 servo readdigvar // commands issued from Online_v2 terminal

CMD[0] => C00
CMD[1] => servo
CMD[2] => readdigvar
Command for C00 ANTENNA
ANTENNA C00 C00
System servo
OP NAME readdigvar
readdigvar
####### Element in Command Queue servo
we wrote on the socket 22 servo readdigvar
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 22 22-Dec-2014 12:04:46 servo readdigvar
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 22
Timestamp is 22-Dec-2014 12:04:46
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192

SEQ number is 22
Timestamp is 22-Dec-2014 12:04:46
System name is servo
Response code is 4
Response type is 1
Servo Digital Status
time
15:00:05
az_run
1
az_aol
0
az_brk
0
az_ccp
0
az_ccf
0
az_cpl
0
az_cfl
0
az_cwr
1
az_enc
0
az_ac
1
el_run
1
el_aol
0
el_brk
0
el_dnp
1
el_dnf
1
el_upp
0
el_upf
0

el_std
0
el_str
1
el_stp
0
el_enc
0
el_ac
1
mode
REMOTE
w50
0
w80
0
pos

trk
TRACKING
slw

ssc
1
dc
0
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C00 servo abort // commands issued from Online_v2 terminal

CMD[0] => C00
CMD[1] => servo
CMD[2] => abort
Command for C00 ANTENNA
ANTENNA C00 C00
System servo
OP NAME abort
abort

####### Element in Command Queue servo
we wrote on the socket 18 servo abort
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 18 22-Dec-2014 12:05:40 servo abort
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 18
Timestamp is 22-Dec-2014 12:05:40
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 18
Timestamp is 22-Dec-2014 12:05:40
System name is servo
Response code is 2
Response type is 2
Servo Final Resp: Command IRREVELENT
Servo Final Resp: Command IRREVELENT
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C00 servo position // commands issued from Online_v2 terminal

CMD[0] => C00
CMD[1] => servo
CMD[2] => position
Command for C00 ANTENNA
ANTENNA C00 C00
System servo
OP NAME position
********** position
####### Element in Command Queue servo
we wrote on the socket 13 servo position
ax B
ang1 90:00:00
ang2 45:00:00
Size of Struct is ######### 1126

INSERTING in Command Queue servo

>> we wrote on the socket 13 22-Dec-2014 12:06:06 servo position
ax B
ang1 90:00:00
ang2 45:00:00
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 13
Timestamp is 22-Dec-2014 12:06:06
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 13
Timestamp is 22-Dec-2014 12:06:06
System name is servo
Response code is 2
Response type is 2
Servo Final Resp: Command IRREVELENT
Servo Final Resp: Command IRREVELENT
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C00 servo readangles // commands issued from Online_v2 terminal

CMD[0] => C00
CMD[1] => servo
CMD[2] => readangles
Command for C00 ANTENNA
ANTENNA C00 C00
System servo
OP NAME readangles
readangles
####### Element in Command Queue servo
we wrote on the socket 20 servo readangles
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 20 22-Dec-2014 12:07:11 servo readangles

Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 20
Timestamp is 22-Dec-2014 12:07:11
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 20
Timestamp is 22-Dec-2014 12:07:11
System name is servo
Response code is 4
Response type is 1
Servo Angles Data
time
15:02:30
az_cp
+001:58:40
az_tp
+090:00:00
el_cp
-001:58:40
el_tp
-001:58:40
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C00 servo stop // commands issued from Online_v2 terminal

CMD[0] => C00
CMD[1] => servo
CMD[2] => stop
Command for C00 ANTENNA
ANTENNA C00 C00
System servo
OP NAME stop
********** stop

####### Element in Command Queue servo
we wrote on the socket 14 servo stop
ax B
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 14 22-Dec-2014 12:08:00 servo stop
ax B
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 14
Timestamp is 22-Dec-2014 12:08:00
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 14
Timestamp is 22-Dec-2014 12:08:00
System name is servo
Response code is 2
Response type is 2
Servo Final Resp: Command IRREVELENT
Servo Final Resp: Command IRREVELENT
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C00 servo readantstatus // commands issued from Online_v2 terminal

CMD[0] => C00
CMD[1] => servo
CMD[2] => readantstatus
Command for C00 ANTENNA
ANTENNA C00 C00
System servo
OP NAME readantstatus
readantstatus

####### Element in Command Queue servo
we wrote on the socket 24 servo readantstatus
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 24 22-Dec-2014 12:08:29 servo readantstatus
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 24
Timestamp is 22-Dec-2014 12:08:29
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 24
Timestamp is 22-Dec-2014 12:08:29
System name is servo
Response code is 4
Response type is 1
Servo Anatenna State
time
15:03:48
az_state
StwRlsBkd
el_state
StwRlsBkd
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C00 servo readversion // commands issued from Online_v2 terminal

CMD[0] => C00
CMD[1] => servo
CMD[2] => readversion
Command for C00 ANTENNA
ANTENNA C00 C00
System servo
OP NAME readversion
readversion

####### Element in Command Queue servo
we wrote on the socket 25 servo readversion
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 25 22-Dec-2014 12:09:54 servo readversion
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 25
Timestamp is 22-Dec-2014 12:09:54
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 25
Timestamp is 22-Dec-2014 12:09:54
System name is servo
Response code is 4
Response type is 1
Servo Software Version
time
15:05:13
Version
2.6
Station Number
C10
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

Command – Response between Online_v2 & PC04 servo system at C01 :

C01 servo hold // commands issued from Online_v2 terminal

CMD[0] => C01
CMD[1] => servo
CMD[2] => hold
Command for C01 ANTENNA
ANTENNA C01 C01
System servo
OP NAME hold

********** hold
 ####### Element in Command Queue servo
we wrote on the socket 12 servo hold
ax B
 Size of Struct is ######### 1126
 INSERTING in Command Queue servo

 >> we wrote on the socket 12 29-Dec-2014 14:31:26 servo hold
ax B
 Size of Struct is ######### 1126
 Size of Response Struct is ######### 2192
SEQ number is 12
Timestamp is 29-Dec-2014 14:31:26
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
 Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
 Size of Response Struct is ######### 2192
SEQ number is 12
Timestamp is 29-Dec-2014 14:31:26
System name is servo
Response code is 2
Response type is 1
Servo Final Resp: Command SUCCESS
Servo Final Resp: Command SUCCESS
 Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C01 servo position // commands issued from Online_v2 terminal

 CMD[0] => C01
 CMD[1] => servo
 CMD[2] => position
Command for C01 ANTENNA
ANTENNA C01 C01
System servo
OP NAME position
 ####### Element in Command Queue servo
we wrote on the socket 13 servo position
ax B
ang1 100:00:00
ang2 45:00:00

Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 13 29-Dec-2014 14:32:04 servo position
ax B
ang1 100:00:00
ang2 45:00:00
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 13
Timestamp is 29-Dec-2014 14:32:04
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 13
Timestamp is 29-Dec-2014 14:32:04
System name is servo
Response code is 2
Response type is 1
Servo Final Resp: Command SUCCESS
Servo Final Resp: Command SUCCESS
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

>> C01 servo readangles // commands issued from Online_v2 terminal

CMD[0] => C01
CMD[1] => servo
CMD[2] => readangles
Command for C01 ANTENNA
ANTENNA C01 C01
System servo
OP NAME readangles
readangles
####### Element in Command Queue servo
we wrote on the socket 20 servo readangles
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 20 29-Dec-2014 14:33:58 servo readangles
Size of Struct is ######### 1126

Size of Response Struct is ######### 2192
SEQ number is 20
Timestamp is 29-Dec-2014 14:33:58
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 20
Timestamp is 29-Dec-2014 14:33:58
System name is servo
Response code is 4
Response type is 1
Servo Angles Data
time
14:40:45
az_cp
+114:26:20
az_tp
+100:00:00
el_cp
+044:59:50
el_tp
+045:00:00
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C01 servo readangles // commands issued from Online_v2 terminal

CMD[0] => C01
CMD[1] => servo
CMD[2] => readangles
Command for C01 ANTENNA
ANTENNA C01 C01
System servo
OP NAME readangles
readangles
####### Element in Command Queue servo
we wrote on the socket 20 servo readangles
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 20 29-Dec-2014 14:34:41 servo readangles
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 20
Timestamp is 29-Dec-2014 14:34:41
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 20
Timestamp is 29-Dec-2014 14:34:41
System name is servo
Response code is 4
Response type is 1
Servo Angles Data
time
14:41:29
az_cp
+100:08:10
az_tp
+100:00:00
el_cp
+045:00:00
el_tp
+045:00:00
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C01 servo track // commands issued from Online_v2 terminal

CMD[0] => C01
CMD[1] => servo
CMD[2] => track
Command for C01 ANTENNA
ANTENNA C01 C01
System servo
OP NAME track
********** track
####### Element in Command Queue servo
we wrote on the socket 11 servo track
ax B

time 16:00:00
ang1 115:00:00
ang2 90:00:00
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 11 29-Dec-2014 14:35:20 servo track
ax B
time 16:00:00
ang1 115:00:00
ang2 90:00:00
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 11
Timestamp is 29-Dec-2014 14:35:20
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 11
Timestamp is 29-Dec-2014 14:35:20
System name is servo
Response code is 2
Response type is 1
Servo Final Resp: Command SUCCESS
Servo Final Resp: Command SUCCESS
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C01 servo readanavar // commands issued from Online_v2 terminal

CMD[0] => C01
CMD[1] => servo
CMD[2] => readanavar
Command for C01 ANTENNA
ANTENNA C01 C01
System servo
OP NAME readanavar
readanavar
####### Element in Command Queue servo
we wrote on the socket 21 servo readanavar

Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 21 29-Dec-2014 14:35:41 servo readanavar
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 21
Timestamp is 29-Dec-2014 14:35:41
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 21
Timestamp is 29-Dec-2014 14:35:41
System name is servo
Response code is 4
Response type is 1
Servo Analog Data
time
14:42:29
az_motor1_current
+0001.2695
az_motor2_current
+0002.2460
az_tacho1
+0004.8808
az_tacho2
+0002.5839
el_motor1_current
+0000.2929
el_motor2_current
-0002.9296
el_tacho1
+0012.6328
el_tacho2
-0007.1777
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

CO1 servo readdigvar // commands issued from Online_v2 terminal

CMD[0] => C01
CMD[1] => servo
CMD[2] => readdigvar
Command for C01 ANTENNA
ANTENNA C01 C01
System servo
OP NAME readdigvar
readdigvar
####### Element in Command Queue servo
we wrote on the socket 22 servo readdigvar
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 22 29-Dec-2014 14:36:42 servo readdigvar
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 22
Timestamp is 29-Dec-2014 14:36:42
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 22
Timestamp is 29-Dec-2014 14:36:42
System name is servo
Response code is 4
Response type is 1
Servo Digital Status
time
14:43:29
az_run
1
az_aol
0
az_brk
0
az_ccp
0
az_ccf

0
az_cpl
0
az_cfl
0
az_cwr
1
az_enc
0
az_ac
1
el_run
1
el_aol
0
el_brk
0
el_dnp
0
el_dnf
0
el_upp
0
el_upf
0
el_std
0
el_str
1
el_stp
0
el_enc
0
el_ac
1
mode
REMOTE
w50
0
w80
0
pos

trk
TRACKING
slw

ssc
1
dc
0
 Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

>> C01 servo readantstatus // commands issued from Online_v2 terminal

 CMD[0] => C01
 CMD[1] => servo
 CMD[2] => readantstatus
Command for C01 ANTENNA
ANTENNA C01 C01
System servo
OP NAME readantstatus
readantstatus
 ####### Element in Command Queue servo
we wrote on the socket 24 servo readantstatus
 Size of Struct is ######### 1126
 INSERTING in Command Queue servo

 >> we wrote on the socket 24 29-Dec-2014 14:39:21 servo readantstatus
 Size of Struct is ######### 1126
 Size of Response Struct is ######### 2192
SEQ number is 24
Timestamp is 29-Dec-2014 14:39:21
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
 Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
 Size of Response Struct is ######### 2192
SEQ number is 24
Timestamp is 29-Dec-2014 14:39:21
System name is servo
Response code is 4
Response type is 1

Servo Anatenna State
time
14:46:08
az_state
Tracking
el_state
Tracking
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C01 servo abort // commands issued from Online_v2 terminal

CMD[0] => C01
CMD[1] => servo
CMD[2] => abort
Command for C01 ANTENNA
ANTENNA C01 C01
System servo
OP NAME abort
abort
####### Element in Command Queue servo
we wrote on the socket 18 servo abort
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 18 29-Dec-2014 14:40:17 servo abort
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 18
Timestamp is 29-Dec-2014 14:40:17
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 18
Timestamp is 29-Dec-2014 14:40:17
System name is servo
Response code is 2
Response type is 1
Servo Final Resp: Command SUCCESS

Servo Final Resp: Command SUCCESS
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C01 servo readantstatus // commands issued from Online_v2 terminal

CMD[0] => C01
CMD[1] => servo
CMD[2] => readantstatus
Command for C01 ANTENNA
ANTENNA C01 C01
System servo
OP NAME readantstatus
readantstatus
####### Element in Command Queue servo
we wrote on the socket 24 servo readantstatus
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 24 29-Dec-2014 14:40:35 servo readantstatus
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 24
Timestamp is 29-Dec-2014 14:40:35
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 24
Timestamp is 29-Dec-2014 14:40:35
System name is servo
Response code is 4
Response type is 1
Servo Anatenna State
time
14:47:23
az_state
Positiong
el_state
Positiong
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

C01 servo stop // commands issued from Online_v2 terminal

```
 CMD[0] => C01
 CMD[1] => servo
 CMD[2] => stop
Command for C01 ANTENNA
ANTENNA C01 C01
System  servo
OP NAME  stop
********** stop
 ####### Element in Command Queue servo
we wrote on the socket 14  servo stop
ax B
 Size of Struct is ######### 1126
 INSERTING in Command Queue servo
 >> we wrote on the socket 14 29-Dec-2014 14:40:58 servo stop
ax B
 Size of Struct is ######### 1126
 Size of Response Struct is ######### 2192
SEQ number is 14
Timestamp is 29-Dec-2014 14:40:58
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
 Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
 Size of Response Struct is ######### 2192
SEQ number is 14
Timestamp is 29-Dec-2014 14:40:58
System name is servo
Response code is 2
Response type is 1
Servo Final Resp: Command SUCCESS
Servo Final Resp: Command SUCCESS
 Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL
```

C01 servo readdigvar // commands issued from Online_v2 terminal

```
 CMD[0] => C01
```

CMD[1] => servo
CMD[2] => readdigvar
Command for C01 ANTENNA
ANTENNA C01 C01
System servo
OP NAME readdigvar
readdigvar
####### Element in Command Queue servo
we wrote on the socket 22 servo readdigvar
Size of Struct is ######### 1126
INSERTING in Command Queue servo

>> we wrote on the socket 22 29-Dec-2014 14:41:16 servo readdigvar
Size of Struct is ######### 1126
Size of Response Struct is ######### 2192
SEQ number is 22
Timestamp is 29-Dec-2014 14:41:16
System name is servo
Response code is 1
Response type is 1
Command Received
Command Received
Writing IMMEDIATE to ONLINE from SERVO THREAD SUCCESSFUL
Size of Response Struct is ######### 2192
SEQ number is 22
Timestamp is 29-Dec-2014 14:41:16
System name is servo
Response code is 4
Response type is 1
Servo Digital Status
time
14:48:04
az_run
0
az_aol
0
az_brk
1
az_ccp
0
az_ccf
0

az_cpl
0
az_cfl
0
az_cwr
1
az_enc
0
az_ac
1
el_run
0
el_aol
0
el_brk
1
el_dnp
0
el_dnf
0
el_upp
0
el_upf
0
el_std
0
el_str
1
el_stp
0
el_enc
0
el_ac
1
mode
REMOTE
w50
0
w80
0
pos

trk

slw

ssc
1
dc
0
Writing FINAL RESPONSE to ONLINE from SERVO THREAD SUCCESSFUL

Online_v2- FPS system Command- Response of C06 antenna testing

C06 fps reboot // Command from Online_V2 terminal

CMD[0] => C06
CMD[1] => fps
CMD[2] => reboot
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME reboot
we wrote on the socket 35 fps reboot
Size of Struct is ######### 1638
####### Element in Command Queue fps
INSERTING in Command Queue fps

>> we wrote on the socket 35 20-Mar-2015 12:16:35 fps reboot
Size of Struct is ######### 1638
Size of Response Struct => 4698
MCM => 1
35
20-Mar-2015 12:16:35
fps
####### NUmber of RESPONSE MSG is 1
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Reboot
Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps run_to_cal **// Command from Online_V2 terminal**

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => run_to_cal
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  run_to_cal
we wrote on the socket 30  fps run_to_cal
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 30 20-Mar-2015 12:16:51 fps run_to_cal
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 30
 20-Mar-2015 12:16:51
 fps
  ####### NUmber of RESPONSE MSG is 1
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999  Run to calibrate
 Writing to ONLINE from FPS THREAD SUCCESSFUL
```

C06 fps run_to_preset **// Command from Online_V2 terminal**

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => run_to_preset
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  run_to_preset

 Enter target encoder value:
15000
```

we wrote on the socket 32 fps run_to_preset
tar_encr_v 76 29
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 32 20-Mar-2015 12:19:07 fps run_to_preset
tar_encr_v 76 29
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 32
 20-Mar-2015 12:19:07
 fps
  ####### NUmber of RESPONSE MSG is 1
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Run to Reset
 Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps free_run_tow   // Command from Online_V2 terminal

 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => free_run_tow
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME free_run_tow
Enter 0-towards 270deg / 1-towards -10deg::
1
we wrote on the socket 31 fps free_run_tow
 1 0
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 31 20-Mar-2015 12:21:40 fps free_run_tow
 1 0

```
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 31
 20-Mar-2015 12:21:40
 fps
  ###### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999  Exec. OK
 Run Free
 Writing to ONLINE from FPS THREAD SUCCESSFUL
```

C06 fps fpsnull   // Command from Online_V2 terminal

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => fpsnull
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  fpsnull
we wrote on the socket 10  fps null
 Size of Struct is ######### 1638
 ###### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 10 20-Mar-2015 12:25:15 fps null
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 10
 20-Mar-2015 12:25:15
 fps
  ###### NUmber of RESPONSE MSG is 4
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
```

```
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999  Exec. OK
 Feed Calibrated and Idle
 EncCount = 1508
 Rpm = 0
 Writing to ONLINE from FPS THREAD SUCCESSFUL
```

C06 fps read_version   // Command from Online_V2 terminal

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => read_version
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  read_version
we wrote on the socket 25  fps read_version
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 25 20-Mar-2015 12:25:58 fps read_version
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 25
 20-Mar-2015 12:25:58
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999  Exec. OK
 Read Version: 8.5
 Writing to ONLINE from FPS THREAD SUCCESSFUL
```

C06 fps read_Max_angle   // Command from Online_V2 terminal

```
 CMD[0] => C06
 CMD[1] => fps
```

```
 CMD[2] => read_Max_angle
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  read_Max_angle
we wrote on the socket 28  fps read_Max_angle
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 28 20-Mar-2015 12:26:22 fps read_Max_angle
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 28
 20-Mar-2015 12:26:22
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999  Exec. OK
 Read Max Angle, 17284
 Writing to ONLINE from FPS THREAD SUCCESSFUL
```

C06 fps read_Min_angle   // Command from Online_V2 terminal

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => read_Min_angle
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  read_Min_angle
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 29 20-Mar-2015 12:27:05 fps read_Min_angle
 Size of Struct is ######### 1638
```

Size of Response Struct => 4698
 MCM => 1
 29
 20-Mar-2015 12:27:05
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
 Read Min Angle, 1468
 Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps read_Brake_dd   // Command from Online_V2 terminal

 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => read_Brake_dd
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME read_Brake_dd
we wrote on the socket 23 fps read_Brake_dd
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 23 20-Mar-2015 12:27:54 fps read_Brake_dd
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 23
 20-Mar-2015 12:27:54
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999

999 Exec. OK
 Read Break Count Diff, 4
 Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps read_tpoint // Command from Online_V2 terminal

 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => read_tpoint
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME read_tpoint
we wrote on the socket 20 fps read_tpoint
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 20 20-Mar-2015 12:28:22 fps read_tpoint
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 20
 20-Mar-2015 12:28:22
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
 Read Turning Point, target: 300
 Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps read_low_rpm // Command from Online_V2 terminal

 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => read_low_rpm
Command for C06 ANTENNA
ANTENNA C06 C06

System fps
OP NAME read_low_rpm
we wrote on the socket 22 fps read_low_rpm
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 22 20-Mar-2015 12:28:45 fps read_low_rpm
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 22
 20-Mar-2015 12:28:45
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
 Read Lower Ramp Limit, 649 int 325
 Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps read_rampupcnt // Command from Online_V2 terminal

 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => read_rampupcnt
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME read_rampupcnt
we wrote on the socket 24 fps read_rampupcnt
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 24 20-Mar-2015 12:29:14 fps read_rampupcnt
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1

24
20-Mar-2015 12:29:14
fps
####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
Read Ramp up Count, 20
Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps read_rampdcnt // Command from Online_V2 terminal

CMD[0] => C06
CMD[1] => fps
CMD[2] => read_rampdcnt
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME read_rampdcnt
####### Element in Command Queue fps
INSERTING in Command Queue fps

>> we wrote on the socket 21 20-Mar-2015 12:29:47 fps read_rampdcnt
Size of Struct is ######### 1638
Size of Response Struct => 4698
MCM => 1
21
20-Mar-2015 12:29:47
fps
####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
Read Ramp Down Count, Slope: 80
Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps read_Max_pwm_cnt // Command from Online_V2 terminal

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => read_Max_pwm_cnt
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  read_Max_pwm_cnt
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 27 20-Mar-2015 12:30:21 fps read_Max_pwm_cnt
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 27
 20-Mar-2015 12:30:21
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999  Exec. OK
 Read Max PWM Count, 80
 Writing to ONLINE from FPS THREAD SUCCESSFUL
```

C06 fps read_stoptimecnt // Command from Online_V2 terminal

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => read_stoptimecnt
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  read_stoptimecnt
we wrote on the socket 26  fps read_stoptimecnt
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps
```

>> we wrote on the socket 26 20-Mar-2015 12:30:44 fps read_stoptimecnt
Size of Struct is ######### 1638
Size of Response Struct => 4698
MCM => 1
26
20-Mar-2015 12:30:44
fps
NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
Read Stop Count, 20
Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps set_tpoint // Command from Online_V2 terminal

CMD[0] => C06
CMD[1] => fps
CMD[2] => set_tpoint
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME set_tpoint

Enter turning point position difference:
200
we wrote on the socket 11 fps set_tpoint
set_tpoint 100 0
Size of Struct is ######### 1638
Element in Command Queue fps
INSERTING in Command Queue fps

>> we wrote on the socket 11 20-Mar-2015 12:31:58 fps set_tpoint
set_tpoint 100 0
Size of Struct is ######### 1638
Size of Response Struct => 4698
MCM => 1
11
20-Mar-2015 12:31:58

fps
####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
Set Turning Point, target: 200
Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps set_Max_pwm_cnt // Command from Online_V2 terminal

CMD[0] => C06
CMD[1] => fps
CMD[2] => set_Max_pwm_cnt
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME set_Max_pwm_cnt
Enter max PWM cnt:
50
we wrote on the socket 17 fps set_Max_pwm_cnt
set_Max_pwm_cnt 50 0
Size of Struct is ######### 1638
####### Element in Command Queue fps
INSERTING in Command Queue fps

>> we wrote on the socket 17 20-Mar-2015 12:32:46 fps set_Max_pwm_cnt
set_Max_pwm_cnt 50 0
Size of Struct is ######### 1638
Size of Response Struct => 4698
MCM => 1
17
20-Mar-2015 12:32:46
fps
####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999

999 Exec. OK
 Set Max PWM Count, 32
 Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps set_Max_pwm_cnt // Command from Online_V2 terminal

 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => set_Max_pwm_cnt
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME set_Max_pwm_cnt
Enter max PWM cnt:
80
we wrote on the socket 17 fps set_Max_pwm_cnt
set_Max_pwm_cnt 80 0
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 17 20-Mar-2015 12:34:41 fps set_Max_pwm_cnt
set_Max_pwm_cnt 80 0
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 17
 20-Mar-2015 12:34:41
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
 Set Max PWM Count, 50
 Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps set_Max_pwm_cnt // Command from Online_V2 terminal

 CMD[0] => C06

```
 CMD[1] => fps
 CMD[2] => set_Max_pwm_cnt
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  set_Max_pwm_cnt
Enter max PWM cnt:
80
we wrote on the socket 17  fps set_Max_pwm_cnt
set_Max_pwm_cnt 80 0
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 17 20-Mar-2015 12:35:14 fps set_Max_pwm_cnt
set_Max_pwm_cnt 80 0
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 17
 20-Mar-2015 12:35:14
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999  Exec. OK
 Set Max PWM Count, 50
 Writing to ONLINE from FPS THREAD SUCCESSFUL
```

C06 fps set_Max_angle // Command from Online_V2 terminal

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => set_Max_angle
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  set_Max_angle
```

Enter angle count:
17300
we wrote on the socket 18 fps set_Max_angle
set_Max_angle 202 33
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 18 20-Mar-2015 12:37:28 fps set_Max_angle
set_Max_angle 202 33
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 18
 20-Mar-2015 12:37:28
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
 Set Max Angle, 17300
 Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps set_min_angle   // Command from Online_V2 terminal

 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => set_min_angle
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME set_min_angle

Enter angle count:
1450
we wrote on the socket 19 fps set_min_angle
set_min_angle 213 2
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps

INSERTING in Command Queue fps

>> we wrote on the socket 19 20-Mar-2015 12:38:07 fps set_min_angle
set_min_angle 213 2
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
19
20-Mar-2015 12:38:07
fps
NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
Set Min Angle, 1450
Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps set_Brake_dd   // Command from Online_V2 terminal

CMD[0] => C06
CMD[1] => fps
CMD[2] => set_Brake_dd
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME set_Brake_dd
Enter Break Cnt difference::
6
we wrote on the socket 14 fps set_Brake_dd
set_Brake_dd 3 0
Size of Struct is ######## 1638
Element in Command Queue fps
INSERTING in Command Queue fps

>> we wrote on the socket 14 20-Mar-2015 12:38:45 fps set_Brake_dd
set_Brake_dd 3 0
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1

14
20-Mar-2015 12:38:45
fps
####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
Set Break Count Diff, 6
Writing to ONLINE from FPS THREAD SUCCESSFUL
C06 fps set_low_rpm

CMD[0] => C06
CMD[1] => fps
CMD[2] => set_low_rpm
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME set_low_rpm
Enter Lower RPM limit:
630
Enter Check-Interval(ms)::
20
we wrote on the socket 13 fps set_low_rpm
set_low_rpm 4 0
Size of Struct is ######### 1638
####### Element in Command Queue fps
INSERTING in Command Queue fps

>> we wrote on the socket 13 20-Mar-2015 12:39:29 fps set_low_rpm
set_low_rpm 4 0
Size of Struct is ######### 1638
Size of Response Struct => 4698
MCM => 1
13
20-Mar-2015 12:39:29
fps
####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999

999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
 Set Lower Ramp Limit, 0 int 20
 Writing to ONLINE from FPS THREAD SUCCESSFUL
C06 fps set_low_rpm

 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => set_low_rpm
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME set_low_rpm
Enter Lower RPM limit:
300
Enter Check-Interval(ms)::
20
we wrote on the socket 13 fps set_low_rpm
set_low_rpm 4 0
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 13 20-Mar-2015 12:40:22 fps set_low_rpm
set_low_rpm 4 0
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 13
 20-Mar-2015 12:40:22
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
 Set Lower Ramp Limit, 0 int 20
 Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps read_low_rpm

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => read_low_rpm
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  read_low_rpm
we wrote on the socket 22  fps read_low_rpm
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 22 20-Mar-2015 12:40:41 fps read_low_rpm
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 22
 20-Mar-2015 12:40:41
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999  Exec. OK
 Read Lower Ramp Limit, 0 int 20
 Writing to ONLINE from FPS THREAD SUCCESSFUL
```

>> C06 fps reboot

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => reboot
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  reboot
we wrote on the socket 35  fps reboot
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
```

INSERTING in Command Queue fps

>> we wrote on the socket 35 20-Mar-2015 12:44:41 fps reboot
Size of Struct is ######### 1638
Size of Response Struct => 4698
MCM => 1
35
20-Mar-2015 12:44:41
fps
####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
Reboot
Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps read_Max_angle

CMD[0] => C06
CMD[1] => fps
CMD[2] => read_Max_angle
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME read_Max_angle
we wrote on the socket 28 fps read_Max_angle
Size of Struct is ######### 1638
####### Element in Command Queue fps
INSERTING in Command Queue fps

>> we wrote on the socket 28 20-Mar-2015 12:45:05 fps read_Max_angle
Size of Struct is ######### 1638
Size of Response Struct => 4698
MCM => 1
28
20-Mar-2015 12:45:05
fps
####### NUmber of RESPONSE MSG is 1
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999

999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Read Max Angle, 17284
 Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps run_fine_tune // Command from Online_V2 terminal

 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => run_fine_tune
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME run_fine_tune

 Enter target encoder value:
1550

Enter PWM cnt:
90
we wrote on the socket 33 fps run_fine_tune
tar_encr_v 7 3
pwm_cnt 90 144
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 33 20-Mar-2015 12:45:59 fps run_fine_tune
tar_encr_v 7 3
pwm_cnt 90 144
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 33
 20-Mar-2015 12:45:59
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999

```
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999  FPS rejected command
 Read Max Angle, 17284
 Writing to ONLINE from FPS THREAD SUCCESSFUL
```

C06 fps run_to_cal // Command from Online_V2 terminal

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => run_to_cal
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  run_to_cal
we wrote on the socket 30  fps run_to_cal
 Size of Struct is ######### 1638
 ###### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 30 20-Mar-2015 12:46:22 fps run_to_cal
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 30
 20-Mar-2015 12:46:22
 fps
  ###### NUmber of RESPONSE MSG is 1
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999  Exec. OK
 Writing to ONLINE from FPS THREAD SUCCESSFUL
```

C06 fps run_to_preset // Command from Online_V2 terminal

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => run_fine_tuneC06
 CMD[3] => fps
 CMD[4] => run_to_preset
```

Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME run_fine_tuneC06

 >> C06 fps run_to_preset

 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => run_to_preset
Command for C06 ANTENNA
ANTENNA C06 C06
System fps
OP NAME run_to_preset

 Enter target encoder value:
15000
we wrote on the socket 32 fps run_to_preset
tar_encr_v 76 29
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 32 20-Mar-2015 12:48:30 fps run_to_preset
tar_encr_v 76 29
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 32
 20-Mar-2015 12:48:30
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 Exec. OK
 Run to Reset
 Writing to ONLINE from FPS THREAD SUCCESSFUL

C06 fps run_fine_tune // Command from Online_V2 terminal

```
 CMD[0] => C06
 CMD[1] => fps
 CMD[2] => run_fine_tune
Command for C06 ANTENNA
ANTENNA C06 C06
System  fps
OP NAME  run_fine_tune

 Enter target encoder value:
15050

Enter PWM cnt:
70
we wrote on the socket 33  fps run_fine_tune
tar_encr_v 101 29
pwm_cnt 70 112
 Size of Struct is ######### 1638
 ####### Element in Command Queue fps
 INSERTING in Command Queue fps

 >> we wrote on the socket 33 20-Mar-2015 12:49:54 fps run_fine_tune
tar_encr_v 101 29
pwm_cnt 70 112
 Size of Struct is ######### 1638
Size of Response Struct => 4698
 MCM => 1
 33
 20-Mar-2015 12:49:54
 fps
  ####### NUmber of RESPONSE MSG is 2
888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888 888
888 888 888 888 888 888 888 888 888 888 888 888 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
999  Exec. OK
 Run to Fine Tune
 Writing to ONLINE from FPS THREAD SUCCESSFUL
```

E02 Testing

ANTENNA DEVICE COMMUNICATION is BROKEN
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.115
E02 thread opened succesfully=> 0
############ SERVER WANTING FOR CLIENT CONNECTION #####
E02 sentinel set

CMD[0] => E02
CMD[1] => sentinel
CMD[2] => set
E02 Antenna
Element in Command Queue sentinel
INSERTING in Command Queue sentinel

>> we wrote on the socket 11 3-Apr-2015 11:17:46 sentinel set
dmask ffff 5555
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
11
3-Apr-2015 11:17:46
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 810 864 856 854 794 812 843 842 832 841 837 829 852 832 840 836 842 832 829 820 819 832 813 835 816 820 818 831 834 836 830 837 844 836 839 832 838 832 829 819 820 818 805 823 823 830 822 832 274 387 381 257 428 769 804 609 289 382 609 675 498 336 781 998 System validated : Sentinel
Command validated : Sentinal Controlling
Digital Mask : ffff 5555
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
E02 fiber_optics mon

CMD[0] => E02
CMD[1] => fiber_optics
CMD[2] => mon
E02 Antenna
Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics

>> we wrote on the socket 10 3-Apr-2015 11:18:00 fiber_optics mon
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
10
3-Apr-2015 11:18:00
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 823 862 829 839 827 810 833 807 818 807 808 838 811 819 828

828 816 821 830 829 830 820 841 835 835 831 824 830 820 823 829 820 825 837 822 816 823 844 828 837 833 832 834 834 839 820 816 818 322 300 425 595 857 841 358 459 538 352 448 821 518 821 728 864 System validated : Fiber Optics
 Command validated : Fiber optics Monitoring
 Fiber Optics Monitoring Done
 Voltage(+12V) : 0.55
 Voltage(-5.0V) : 1.00
 Voltage(-3.1V) : 0.73
 Voltage(-1.1V) : 0.77
 Voltage(GND) : 0.61
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

W01 testing

############## SERVER WANTING FOR CLIENT CONNECTION #####
 ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.121
W01 Sentinel thread opened succesfully=> 0
############## SERVER WANTING FOR CLIENT CONNECTION #####
W01 sentinel mon

 CMD[0] => W01
 CMD[1] => sentinel
 CMD[2] => mon
W01 Antenna
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel

 >> we wrote on the socket 10 3-Apr-2015 11:46:57 sentinel mon
 Size of Struct is ######## 1638
Size of Response Struct => 4698
 MCM => 1
 10
  3-Apr-2015 11:46:57
 sentinel
  ###### NUmber of RESPONSE MSG is 3
No summery !! 830 835 835 824 844 807 812 805 802 805 816 827 825 828 840 844 845 837 852 836 839 828 828 816 813 825 813 820 802 810 833 828 840 828 829 825 836 836 840 840 825 823 840 820 808 814 816 816 720 636 638 670 820 828 818 819 811 773 709 692 633 789 841 732 System validated : Sentinel
 Command validated : Sentinal Monitoring
 Sentinel Monitoring Done
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
W01 fiber_optics mon

 CMD[0] => W01
 CMD[1] => fiber_optics
 CMD[2] => mon
W01 Antenna

Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics

>> we wrote on the socket 10 3-Apr-2015 11:47:07 fiber_optics mon
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
10
3-Apr-2015 11:47:07
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 882 834 849 832 822 838 812 808 828 839 819 813 820 815 806 806 834 827 827 854 833 828 835 824 848 836 817 808 840 815 822 812 794 818 805 811 811 816 830 832 827 827 837 822 847 832 838 832 790 802 806 811 817 824 700 624 558 684 871 780 846 817 826 735 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 0.77
Voltage(-5.0V) : 0.77
Voltage(-3.1V) : 0.82
Voltage(-1.1V) : 0.82
Voltage(GND) : 0.86
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

W01 fiber_optics set

CMD[0] => W01
CMD[1] => fiber_optics
CMD[2] => set
W01 Antenna
Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics

>> we wrote on the socket 11 3-Apr-2015 11:49:04 fiber_optics set
rf_attn 25 35
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
11
3-Apr-2015 11:49:04
fiber_optics
NUmber of RESPONSE MSG is 3
No summery !! 849 822 824 838 836 823 830 827 816 834 832 829 821 822 816 801 804 825 814 816 841 825 830 828 824 820 834 836 844 827 825 814 809 817 812 824 829 819 818 820 831 825 836 832 835 823 848 828 598 800 807 795 818 829 648 476 400 599 804 824 809 814 848 921 System validated : Fiber Optics
Command validated : Fiber optics Controlling
RF Attenuation : 25 35
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

W01 sentinel mon

 CMD[0] => W01
 CMD[1] => sentinel
 CMD[2] => mon
W01 Antenna
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel

 >> we wrote on the socket 10 3-Apr-2015 11:49:43 sentinel mon
 Size of Struct is ######## 1638
Size of Response Struct => 4698
 MCM => 1
 10
  3-Apr-2015 11:49:43
 sentinel
  ###### NUmber of RESPONSE MSG is 3
No summery !! 811 815 806 800 808 857 839 840 843 826 844 832 832 815 827 820 839 834 813 817 801 816 810 827 822 836 828 840 833 833 825 840 838 840 834 832 840 814 816 807 816 815 816 810 833 844 825 830 815 817 816 716 500 536 490 680 822 808 824 813 786 720 695 670 System validated : Sentinel
 Command validated : Sentinal Monitoring
 Sentinel Monitoring Done
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

W01 sentinel mon

 CMD[0] => W01
 CMD[1] => sentinel
 CMD[2] => mon
W01 Antenna
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel

 >> we wrote on the socket 10 3-Apr-2015 11:53:18 sentinel mon
 Size of Struct is ######## 1638
Size of Response Struct => 4698
 MCM => 1
 10
  3-Apr-2015 11:53:18
 sentinel
  ###### NUmber of RESPONSE MSG is 3
No summery !! 826 823 825 845 840 825 824 823 824 820 820 814 835 840 818 826 820 820 832 824 813 826 820 824 817 822 824 828 829 821 822 832 830 832 828 817 825 832 832 822 828 828 829 829 832 824 820 820 329 617 576 556 432 179 773 704 357 337 436 1285 595 372 360 588 System validated : Sentinel
 Command validated : Sentinal Monitoring
 Sentinel Monitoring Done

Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

C13

C13 fiber_optics mon

CMD[0] => C13
CMD[1] => fiber_optics
CMD[2] => mon
C13 Antenna
Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics

>> we wrote on the socket 10 3-Apr-2015 12:24:57 fiber_optics mon
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
10
3-Apr-2015 12:24:57
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 797 807 803 822 810 836 838 836 837 830 848 834 851 832 842 833 849 832 831 823 806 814 803 817 818 831 832 828 829 830 836 828 837 842 847 832 838 828 814 816 804 818 818 825 832 834 832 828 848 837 805 839 841 832 846 782 803 798 776 736 785 823 884 1017 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 0.74
Voltage(-5.0V) : 0.74
Voltage(-3.1V) : 0.83
Voltage(-1.1V) : 0.83
Voltage(GND) : 0.90
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

C13 fiber_optics set

CMD[0] => C13
CMD[1] => fiber_optics
CMD[2] => set
C13 Antenna
Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics

>> we wrote on the socket 11 3-Apr-2015 12:25:46 fiber_optics set
rf_attn 25 35
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
11

3-Apr-2015 12:25:46
fiber_optics
NUmber of RESPONSE MSG is 3
No summery !! 856 848 852 828 828 847 844 826 825 814 810 819 812 827 817 826 827 827 833 829 833 833 846 839 846 833 831 821 821 817 806 819 811 828 825 827 836 828 831 832 839 841 859 830 839 826 823 812 792 728 665 400 199 202 386 693 760 833 849 1160 443 318 265 599 System validated : Fiber Optics
Command validated : Fiber optics Controlling
RF Attenuation : 25 35
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

C13 sentinel mon

CMD[0] => C13
CMD[1] => sentinel
CMD[2] => mon
C13 Antenna
Element in Command Queue sentinel
INSERTING in Command Queue sentinel

>> we wrote on the socket 10 3-Apr-2015 12:26:21 sentinel mon
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
10
3-Apr-2015 12:26:21
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 803 827 815 842 854 824 838 833 840 832 836 822 834 834 842 840 829 820 811 819 804 817 807 828 825 832 835 829 837 823 833 832 845 832 837 831 822 817 806 812 811 816 825 824 834 827 832 835 781 701 387 157 166 266 590 782 791 805 702 1092 274 210 348 718 System validated : Sentinel
Command validated : Sentinal Monitoring
Sentinel Monitoring Done
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

C13 sentinel set

CMD[0] => C13
CMD[1] => sentinel
CMD[2] => set
C13 Antenna
Element in Command Queue sentinel
INSERTING in Command Queue sentinel

>> we wrote on the socket 11 3-Apr-2015 12:27:34 sentinel set
dmask ffff 5555
Size of Struct is ######## 1638
Size of Response Struct => 4698

MCM => 1
11
3-Apr-2015 12:27:34
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 859 835 848 825 824 821 805 819 812 819 809 812 812 819 822 820 833 832 830 825 838 828 843 836 850 834 834 818 807 813 806 819 812 822 830 811 821 828 828 823 842 833 846 839 841 827 814 813 894 846 862 759 494 237 179 212 585 804 840 1155 666 616 410 572 System validated : Sentinel
Command validated : Sentinal Controlling
Digital Mask : ffff 5555
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

S03

S03 fiber_optics mon

CMD[0] => S03
CMD[1] => fiber_optics
CMD[2] => mon
S03 Antenna
Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics

>> we wrote on the socket 10 3-Apr-2015 12:58:52 fiber_optics mon
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
10
3-Apr-2015 12:58:52
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 682 1238 408 934 941 862 942 774 548 156 108 359 540 963 1156 917 824 832 807 836 832 830 825 829 843 824 832 831 828 824 833 838 836 832 815 816 832 827 824 831 824 818 838 816 826 832 819 820 833 746 781 373 438 654 804 788 836 836 764 941 512 500 736 992 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 0.94
Voltage(-5.0V) : 0.91
Voltage(-3.1V) : 1.61
Voltage(-1.1V) : 0.38
Voltage(GND) : 1.86
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

S03 sentinel mon

CMD[0] => S03
CMD[1] => sentinel

CMD[2] => mon
S03 Antenna
Element in Command Queue sentinel
INSERTING in Command Queue sentinel

>> we wrote on the socket 10 3-Apr-2015 13:00:52 sentinel mon
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
10
3-Apr-2015 13:00:52
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 1228 934 790 654 641 252 664 608 815 1184 668 690 218 162 128 399 836 834 820 835 838 828 832 824 824 820 834 816 846 819 827 833 819 818 832 824 835 842 840 822 835 826 830 819 832 821 814 836 440 237 682 902 690 458 452 352 374 422 526 1142 88 345 266 533 System validated : Sentinel
Command validated : Sentinal Monitoring
Sentinel Monitoring Done
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

S03 fiber_optics mon

CMD[0] => S03
CMD[1] => fiber_optics
CMD[2] => mon
S03 Antenna
Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics

>> we wrote on the socket 10 3-Apr-2015 13:02:39 fiber_optics mon
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
10
3-Apr-2015 13:02:39
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 1188 713 365 904 574 456 1025 1086 660 665 603 364 123 140 499 492 837 816 823 840 828 828 824 821 818 824 834 812 839 840 824 827 830 824 836 838 839 822 824 825 825 825 832 828 820 843 830 835 733 834 826 820 781 690 407 581 544 804 796 811 832 798 666 772 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 2.34
Voltage(-5.0V) : -0.07
Voltage(-3.1V) : -0.35
Voltage(-1.1V) : -0.58
Voltage(GND) : 0.35

Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

S03 fiber_optics mon

CMD[0] => S03
CMD[1] => fiber_optics
CMD[2] => mon
S03 Antenna
Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics

>> we wrote on the socket 10 3-Apr-2015 13:03:30 fiber_optics mon
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
10
3-Apr-2015 13:03:30
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 1040 940 273 595 578 1025 788 1013 664 744 352 240 232 589 934 701 838 820 827 834 828 834 823 832 840 819 835 824 836 829 828 834 818 840 833 824 819 824 838 830 828 827 824 832 835 839 840 836 784 827 819 739 638 424 443 422 800 801 867 901 794 672 571 803 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 0.49
Voltage(-5.0V) : 1.40
Voltage(-3.1V) : 0.98
Voltage(-1.1V) : 0.35
Voltage(GND) : -1.04
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

S01

############## SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.120
S01 sentinel thread opened succesfully=> 0
############## SERVER WANTING FOR CLIENT CONNECTION #####
S01 fiber_optics mon

CMD[0] => S01
CMD[1] => fiber_optics
CMD[2] => mon
S01 Antenna
Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics

>> we wrote on the socket 10 3-Apr-2015 13:30:58 fiber_optics mon
Size of Struct is ######## 1638

Size of Response Struct => 4698
 MCM => 1
 10
  3-Apr-2015 13:30:58
 fiber_optics
  ###### NUmber of RESPONSE MSG is 8
No summery !! 715 1034 1099 1270 742 606 576 146 200 113 552 716 1090 844 1079 446 822 817 833 832 837 834 828 826 834 828 827 830 827 820 841 825 823 819 830 834 830 828 825 823 826 815 821 833 833 833 826 832 748 739 576 588 641 811 822 826 822 787 799 741 492 716 751 792 System validated : Fiber Optics
 Command validated : Fiber optics Monitoring
 Fiber Optics Monitoring Done
 Voltage(+12V) : -0.35
 Voltage(-5.0V) : -0.47
 Voltage(-3.1V) : 0.57
 Voltage(-1.1V) : 1.40
 Voltage(GND) : 1.60
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

S01 sentinel mon

 CMD[0] => S01
 CMD[1] => sentinel
 CMD[2] => mon
S01 Antenna
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel

 >> we wrote on the socket 10 3-Apr-2015 13:31:56 sentinel mon
 Size of Struct is ######## 1638
Size of Response Struct => 4698
 MCM => 1
 10
  3-Apr-2015 13:31:56
 sentinel
  ###### NUmber of RESPONSE MSG is 3
No summery !! 634 1241 192 827 412 707 592 775 1081 566 446 268 795 250 512 530 840 841 840 820 828 823 834 828 840 842 838 822 829 833 828 825 822 820 832 844 822 827 828 829 824 832 822 829 842 820 836 824 227 661 873 336 391 241 344 342 236 412 880 1219 521 456 398 777 System validated : Sentinel
 Command validated : Sentinal Monitoring
 Sentinel Monitoring Done
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

S01 fiber_optics mon

 CMD[0] => S01
 CMD[1] => fiber_optics
 CMD[2] => mon

S01 Antenna
Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics

>> we wrote on the socket 10 3-Apr-2015 13:32:37 fiber_optics mon
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
10
3-Apr-2015 13:32:37
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 960 1212 1132 718 946 311 539 393 509 790 860 594 608 557 234 342 830 820 820 835 832 835 840 824 831 829 838 832 825 840 822 834 834 830 827 827 827 828 822 818 817 823 823 830 827 832 828 840 420 283 332 261 747 1003 896 272 364 261 377 1291 323 250 804 520 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 2.46
Voltage(-5.0V) : 0.48
Voltage(-3.1V) : 0.37
Voltage(-1.1V) : 0.09
Voltage(GND) : 0.91
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

S01 fiber_optics mon

CMD[0] => S01
CMD[1] => fiber_optics
CMD[2] => mon
S01 Antenna
Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics

>> we wrote on the socket 10 3-Apr-2015 13:33:24 fiber_optics mon
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
10
3-Apr-2015 13:33:24
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 592 1392 206 720 838 832 696 846 880 486 390 268 745 338 560 824 832 826 817 824 830 832 832 824 840 832 832 837 835 832 826 834 824 826 839 834 835 823 831 823 828 830 820 819 818 824 807 832 418 637 850 181 342 422 373 480 249 777 904 1143 535 265 344 894 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 1.28

Voltage(-5.0V) : -1.72
Voltage(-3.1V) : 2.97
Voltage(-1.1V) : 0.56
Voltage(GND) : 1.95
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

Central Square Antennae (except C02,C04 & C11)

[teleset@tellab2 Online]$ ./online_v2
HIGHUSER thread CREATED=> 0
SERVO thread CREATED=> 0
GUI INTERFACE thread CREATED=> 0
PYTHON INTERFACE thread CREATED=> 0
==> SERVO SERVER WANTING FOR CLIENT CONNECTION ===>MCM SYSTEM thread CREATED=> 0

>>
msgget: Calling msgget(0xc9,01600)
msgget: msgget succeeded: msqid = 0
Sucessfully Created MESSAGE QUEUE ID=0
SERVER WANTING FOR GUI CLIENT TO CONNECT
$$$ SERVER WANTING FOR PYTHON ENVIRONMENT CLIENT TO CONNECT $$$$
############# SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.105
C05 SEN thread opened succesfully=> 0
############# SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.108
C08 SEN thread opened succesfully=> 0
############# SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.110
C10 Sentinel thread opened succesfully=> 0
############# SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.100
SENTINEL thread opened succesfully=> 0
############# SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.33.2
W06 Sentinel thread opened succesfully=> 0
############# SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.32.5
W05 Sen/OF thread opened succesfully=> 0
############# SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.31.2
BACKEND thread opened succesfully=> 0
############# SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.31.3
S06 sentinel thread opened succesfully=> 0
############# SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.32.2
W02 sentinel thread opened succesfully=> 0

############ SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.109
C09 Sentinel thread opened succesfully=> 0
############ SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.32.3
W03 Sentinel thread opened succesfully=> 0
############ SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.103
C03 SEN thread opened succesfully=> 0
############ SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.106
C06 SEN thread opened succesfully=> 0
############ SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.32.4
W04 Sentinel thread opened succesfully=> 0
############ SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.114
C14 Sentinel thread opened succesfully=> 0
############ SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.31.4
FRONT thread opened succesfully=> 0
############ SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.112
C12 Sentinel thread opened succesfully=> 0
############ SERVER WANTING FOR CLIENT CONNECTION #####
ACCEPTED CONNECTION FROM MCM DEVICE 192.168.21.101
C01 SENTINEL thread opened succesfully=> 0
############ SERVER WANTING FOR CLIENT CONNECTION #####
Client=> 192.168.8.87 Port 51451
====>Size of read is 1448
PYTHON ANTENNA name C00
PYTHON SYSTEM name fiber_optics
param_number 0
Element in Command Queue fiber_optics
INSERTING in Command Queue fiber_optics
====>Size of read is 350
we wrote on the socket 1 3-Apr-2015 14:33:34 fiber_optics mon
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
1
3-Apr-2015 14:33:34
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 838 1352 680 903 790 1220 1399 1355 1330 1317 806 704 1144 1276 1341 905 818 835 816 824 827 828 819 832 831 824 827 819 832 832 828 832 836 819 828 828 821 828 821 834 821 819 826 821 828 830 825 825 122 240 251 150 126 35 0 0 0 86 289 576 252 224 232 332 System validated : Fiber Optics
Command validated : Fiber optics Monitoring

Fiber Optics Monitoring Done
Voltage(+12V) : -0.13
Voltage(-5.0V) : 0.78
Voltage(-3.1V) : 1.09
Voltage(-1.1V) : -0.99
Voltage(GND) : -3.55
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
====>Size of read is 1448
PYTHON ANTENNA name C00
PYTHON SYSTEM name fps
param_number 0
Element in Command Queue fps
INSERTING in Command Queue fps
====>Size of read is 350
====>Size of read is 1798
PYTHON ANTENNA name C00
PYTHON SYSTEM name sentinel
param_number 0
Element in Command Queue sentinel
INSERTING in Command Queue sentinel
PYTHON ANTENNA name C01
PYTHON SYSTEM name sentinel
param_number 0
Element in Command Queue sentinel
INSERTING in Command Queue sentinel
PYTHON ANTENNA name C03
PYTHON SYSTEM name sentinel
param_number 0
Element in Command Queue sentinel
INSERTING in Command Queue sentinel
PYTHON ANTENNA name C05
PYTHON SYSTEM name sentinel
param_number 0
Element in Command Queue sentinel
INSERTING in Command Queue sentinel
PYTHON ANTENNA name C06
PYTHON SYSTEM name sentinel
param_number 0
Element in Command Queue sentinel
INSERTING in Command Queue sentinel
PYTHON ANTENNA name C08
PYTHON SYSTEM name sentinel
param_number 0
Element in Command Queue sentinel
INSERTING in Command Queue sentinel
PYTHON ANTENNA name C09
PYTHON SYSTEM name sentinel
param_number 0
Element in Command Queue sentinel

INSERTING in Command Queue sentinel
PYTHON ANTENNA name C10
PYTHON SYSTEM name sentinel
param_number 0
Element in Command Queue sentinel
INSERTING in Command Queue sentinel
PYTHON ANTENNA name C12
PYTHON SYSTEM name sentinel
param_number 0
Element in Command Queue sentinel
INSERTING in Command Queue sentinel
PYTHON ANTENNA name C14
PYTHON SYSTEM name sentinel
param_number 0
Element in Command Queue sentinel
INSERTING in Command Queue sentinel
we wrote on the socket 1 3-Apr-2015 14:40:39 sentinel mon
Size of Struct is ######## 1638
we wrote on the socket 1 3-Apr-2015 14:40:39 sentinel mon
Size of Struct is ######## 1638
we wrote on the socket 1 3-Apr-2015 14:40:39 sentinel mon
Size of Struct is ######## 1638
we wrote on the socket 1 3-Apr-2015 14:40:39 sentinel mon
Size of Struct is ######## 1638
we wrote on the socket 1 3-Apr-2015 14:40:39 sentinel mon
Size of Struct is ######## 1638
we wrote on the socket 1 3-Apr-2015 14:40:39 sentinel mon
Size of Struct is ######## 1638
we wrote on the socket 1 3-Apr-2015 14:40:39 sentinel mon
Size of Struct is ######## 1638
we wrote on the socket 1 3-Apr-2015 14:40:39 sentinel mon
Size of Struct is ######## 1638
we wrote on the socket 1 3-Apr-2015 14:40:39 sentinel mon
Size of Struct is ######## 1638
we wrote on the socket 1 3-Apr-2015 14:40:39 sentinel mon
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
1
3-Apr-2015 14:40:39
sentinel
NUmber of RESPONSE MSG is 3
No summery !!   982 1933 41 1494 1155 858 832 826 820 822 822 826 825 828 824 829 828 823 828 828 819 827 827 826 829 829 829 822 825 821 827 828 821 827 826 823 828 820 822 828 824 829 828 830 826 824 828 825 826 824 829 827 822 825 829 828 825 824 826 988 844 825 227 765 System validated : Sentinel
Size of Response Struct => 4698
Command validated : Sentinal Monitoring
MCM => 1

1
3-Apr-2015 14:40:39
sentinel
Sentinel Monitoring Done
NUmber of RESPONSE MSG is 3
No summery !! 976 1913 24 1153 1496 894 829 824 827 824 818 828 829 825 822 827 824 827 827 824 824 829 826 822 828 830 823 825 825 825 828 827 828 825 824 823 822 825 826 831 828 817 826 821 824 828 821 827 771 828 829 830 822 828 826 829 827 823 829 978 839 829 828 210 System validated : Sentinel
Command validated : Sentinal Monitoring
Sentinel Monitoring Done
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
1
3-Apr-2015 14:40:39
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 808 808 801 838 843 821 828 823 824 821 826 831 820 832 824 830 823 824 822 817 829 832 821 818 831 826 832 828 828 820 816 819 827 819 824 825 824 828 828 832 832 815 819 834 816 823 826 819 851 1156 1056 563 725 389 156 205 269 539 723 910 987 686 531 281 System validated : Sentinel
Command validated : Sentinal Monitoring
Sentinel Monitoring Done
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
1
3-Apr-2015 14:40:39
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 836 825 822 827 826 820 825 825 829 827 820 826 832 822 826 826 830 828 817 824 832 826 826 820 827 825 822 818 822 830 824 832 823 829 829 827 824 832 827 828 829 825 827 824 822 829 829 824 182 204 334 582 645 815 819 772 649 142 431 1236 316 586 365 376 System validated : Sentinel
Command validated : Sentinal Monitoring
Sentinel Monitoring Done
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
1
3-Apr-2015 14:40:39
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 981 1932 27 1154 1497 894 829 826 828 828 828 830 828 828 827 827 826 823 828 824 820 827 828 827 827 826 826 828 830 828 826 822 821 830 830 828 826 832 824 823 830 822 826 828 826 828 830 828 829 828 827 828 830 826 822 828 826 825 824 1009 847 827 208 769 System validated : Sentinel

Command validated : Sentinal Monitoring
Sentinel Monitoring Done
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
1
3-Apr-2015 14:40:39
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 980 1924 25 1504 1172 864 824 825 825 828 828 821 828 820 825 825 831 825 824 827 828 828 824 826 828 825 826 826 830 823 827 821 828 827 830 826 824 826 825 825 828 824 824 828 827 826 827 825 771 826 825 828 822 826 826 821 824 825 824 825 824 826 827 222 System validated : Sentinel
Command validated : Sentinal Monitoring
Sentinel Monitoring Done
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
1
3-Apr-2015 14:40:39
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 980 1928 30 1466 1154 859 829 828 827 827 831 828 828 829 829 826 828 826 828 825 829 823 825 828 828 830 829 827 825 824 831 828 832 832 829 829 829 829 828 829 830 826 828 829 826 828 828 822 775 826 829 827 824 832 827 832 829 826 828 982 844 826 829 237 System validated : Sentinel
Command validated : Sentinal Monitoring
Sentinel Monitoring Done
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
1
3-Apr-2015 14:40:39
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 981 1914 25 1500 1147 859 825 827 826 825 828 821 823 821 828 828 830 821 830 830 828 826 828 826 829 828 822 829 824 829 826 829 827 827 829 826 829 825 825 826 828 826 830 825 829 823 825 830 824 832 832 824 828 824 831 826 830 828 823 983 844 828 285 774 System validated : Sentinel
Command validated : Sentinal Monitoring
Sentinel Monitoring Done
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
1
3-Apr-2015 14:40:39
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 978 1930 24 1511 1173 864 826 829 827 829 828 827 827 821 832

829 829 829 829 826 829 832 835 827 827 826 828 827 826 826 822 829 824 832 828 832 826 829 831 830 829 824 827 824 825 823 829 825 766 832 828 828 829 830 825 826 828 828 828 984 842 830 830 192 System validated : Sentinel
 Command validated : Sentinal Monitoring
 Sentinel Monitoring Done
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
 MCM => 1
 1
  3-Apr-2015 14:40:39
 sentinel
  ###### NUmber of RESPONSE MSG is 3
No summery !! 1012 777 592 1053 1258 1423 1596 1405 1311 1115 861 989 1196 890 1250 1116 819 836 816 833 841 824 819 830 823 826 821 825 826 837 827 823 824 832 830 824 823 827 827 836 820 820 820 836 828 829 826 830 139 147 192 64 34 0 0 0 103 107 239 948 221 200 131 165 System validated : Sentinel
 Command validated : Sentinal Monitoring
 Sentinel Monitoring Done
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

====>Size of read is 1448
PYTHON ANTENNA name C00
PYTHON SYSTEM name fiber_optics
 param_number 0
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C01
PYTHON SYSTEM name fiber_optics
 param_number 0
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C03
PYTHON SYSTEM name fiber_optics
 param_number 0
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C05
PYTHON SYSTEM name fiber_optics
 param_number 0
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C06
PYTHON SYSTEM name fiber_optics
 param_number 0
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C08
PYTHON SYSTEM name fiber_optics
 param_number 0

```
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C09
PYTHON SYSTEM name fiber_optics
 param_number 0
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C10
PYTHON SYSTEM name fiber_optics
 param_number 0
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C12
PYTHON SYSTEM name fiber_optics
 param_number 0
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C14
PYTHON SYSTEM name fiber_optics
 param_number 0
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
====>Size of read is 350
we wrote on the socket 2  3-Apr-2015 14:43:11 fiber_optics mon
 Size of Struct is ######## 1638
we wrote on the socket 2  3-Apr-2015 14:43:11 fiber_optics mon
 Size of Struct is ######## 1638
we wrote on the socket 2  3-Apr-2015 14:43:11 fiber_optics mon
 Size of Struct is ######## 1638
we wrote on the socket 2  3-Apr-2015 14:43:11 fiber_optics mon
 Size of Struct is ######## 1638
we wrote on the socket 2  3-Apr-2015 14:43:11 fiber_optics mon
 Size of Struct is ######## 1638
we wrote on the socket 2  3-Apr-2015 14:43:11 fiber_optics mon
 Size of Struct is ######## 1638
we wrote on the socket 2  3-Apr-2015 14:43:11 fiber_optics mon
 Size of Struct is ######## 1638
we wrote on the socket 2  3-Apr-2015 14:43:11 fiber_optics mon
 Size of Struct is ######## 1638
we wrote on the socket 2  3-Apr-2015 14:43:11 fiber_optics mon
 Size of Struct is ######## 1638
we wrote on the socket 2  3-Apr-2015 14:43:11 fiber_optics mon
 Size of Struct is ######## 1638
Size of Response Struct => 4698
 MCM => 1
 2
  3-Apr-2015 14:43:11
 fiber_optics
  ###### NUmber of RESPONSE MSG is 8
```

No summery !! 980 1930 36 1492 1152 861 821 825 827 829 823 822 823 831 825 824 823 824 832 829 822 823 828 825 823 828 824 821 826 828 821 828 823 832 824 826 830 830 823 828 827 821 830 829 832 822 826 827 829 828 823 829 828 830 828 830 824 831 827 990 845 825 237 768 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 0.03
Voltage(-5.0V) : -4.84
Voltage(-3.1V) : 4.81
Voltage(-1.1V) : -2.62
Voltage(GND) : -0.84
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
2
3-Apr-2015 14:43:11
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 979 1923 24 1503 1175 859 825 820 826 825 828 826 829 828 828 824 828 822 828 825 826 825 823 827 820 828 825 827 825 820 829 826 825 826 827 827 825 827 822 830 826 828 825 828 827 826 827 824 764 828 826 828 828 826 828 827 824 825 827 825 826 830 827 200 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 0.00
Voltage(-5.0V) : -4.81
Voltage(-3.1V) : 4.88
Voltage(-1.1V) : -2.65
Voltage(GND) : -0.98
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
2
3-Apr-2015 14:43:11
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 984 1923 29 1464 1152 862 828 832 827 832 829 829 826 824 831 827 831 827 827 830 832 822 826 830 828 829 828 830 830 827 832 824 830 830 832 829 829 827 829 829 827 828 832 829 831 829 828 831 772 833 828 829 825 829 828 828 822 825 828 983 839 826 830 248 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : -0.02
Voltage(-5.0V) : -4.84
Voltage(-3.1V) : 4.84
Voltage(-1.1V) : -2.49
Voltage(GND) : -0.89
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698

MCM => 1
2
3-Apr-2015 14:43:11
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 981 1925 28 1153 1502 890 828 829 827 828 824 824 830 824 828 825 831 824 826 830 829 829 827 827 829 828 829 822 825 824 827 828 827 821 826 825 824 828 830 829 824 827 830 825 827 827 829 824 826 827 826 827 826 824 822 827 826 830 827 1016 843 831 350 782 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : -0.00
Voltage(-5.0V) : -4.85
Voltage(-3.1V) : 4.87
Voltage(-1.1V) : -0.87
Voltage(GND) : -2.64
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
2
3-Apr-2015 14:43:11
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 981 1910 29 1160 1499 896 824 823 825 822 827 828 826 828 825 826 828 830 824 822 828 827 830 824 826 828 829 830 830 820 826 823 829 828 827 827 825 823 828 826 827 826 827 828 826 824 821 829 766 830 828 819 827 828 826 828 828 825 826 976 836 832 827 183 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 0.01
Voltage(-5.0V) : -4.76
Voltage(-3.1V) : 4.85
Voltage(-1.1V) : -0.89
Voltage(GND) : -2.65
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
2
3-Apr-2015 14:43:11
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 806 814 849 826 836 814 819 828 822 821 826 824 824 823 820 824 826 827 822 828 819 823 820 826 830 818 822 825 823 823 820 817 836 826 827 828 819 822 816 828 824 835 824 825 818 822 821 824 460 645 775 940 625 383 173 159 464 554 847 1332 1086 497 393 388 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 0.80
Voltage(-5.0V) : 0.75

Voltage(-3.1V) : 0.69
Voltage(-1.1V) : 0.73
Voltage(GND) : 0.83
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
2
3-Apr-2015 14:43:11
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 833 836 826 816 832 820 828 824 838 836 826 825 827 824 827 830 825 826 821 832 830 828 837 830 830 823 825 822 822 833 825 822 824 824 832 827 824 824 826 825 819 824 825 822 825 826 827 829 208 169 307 259 604 665 784 726 594 350 237 1267 199 424 641 1052 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 0.82
Voltage(-5.0V) : 0.77
Voltage(-3.1V) : 0.81
Voltage(-1.1V) : 0.78
Voltage(GND) : 0.80
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
2
3-Apr-2015 14:43:11
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 980 1926 26 1509 1170 862 830 828 824 830 828 827 829 830 822 831 828 830 827 832 821 826 823 828 831 824 829 828 830 825 824 829 829 826 828 826 824 828 830 826 828 832 829 832 824 824 830 822 772 825 829 828 832 829 828 824 832 830 825 984 842 825 825 240 System validated : Fiber Optics
Command validated : Fiber optics Monitoring
Fiber Optics Monitoring Done
Voltage(+12V) : 0.00
Voltage(-5.0V) : -4.83
Voltage(-3.1V) : 4.84
Voltage(-1.1V) : -2.68
Voltage(GND) : -0.95
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
2
3-Apr-2015 14:43:11
fiber_optics
NUmber of RESPONSE MSG is 8
No summery !! 980 1920 26 1497 1158 860 827 828 824 824 826 826 824 830 829 826 830 822 825 827 827 830 825 826 829 828 828 825 827 821 828 830 822 827 829 831 828 820 824 827 825 826 829 824 828 822 826 824 820 822 825 825 828 827 827 824 828 825 824 977 843

824 269 772 System validated : Fiber Optics
 Command validated : Fiber optics Monitoring
 Fiber Optics Monitoring Done
 Voltage(+12V) : 0.01
 Voltage(-5.0V) : -4.76
 Voltage(-3.1V) : 4.89
 Voltage(-1.1V) : -2.64
 Voltage(GND) : -0.89
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
 MCM => 1
 2
  3-Apr-2015 14:43:11
 fiber_optics
  ###### NUmber of RESPONSE MSG is 8
No summery !! 1241 1655 585 656 900 740 1204 1418 1348 1342 1176 896 1252 1280 1184 1154 820 842 835 824 830 828 824 832 822 824 827 830 831 817 816 827 830 820 815 824 825 827 832 832 835 828 826 827 819 826 820 832 78 134 193 188 117 56 48 0 0 6 126 478 238 281 219 198 System validated : Fiber Optics
 Command validated : Fiber optics Monitoring
 Fiber Optics Monitoring Done
 Voltage(+12V) : 0.57
 Voltage(-5.0V) : -3.31
 Voltage(-3.1V) : 1.78
 Voltage(-1.1V) : -2.89
 Voltage(GND) : -3.02
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL

PYTHON ANTENNA name C00
PYTHON SYSTEM name sentinel
 param_number 1
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel
PYTHON ANTENNA name C01
PYTHON SYSTEM name sentinel
 param_number 1
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel
PYTHON ANTENNA name C03
PYTHON SYSTEM name sentinel
 param_number 1
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel
PYTHON ANTENNA name C05
PYTHON SYSTEM name sentinel
 param_number 1
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel
PYTHON ANTENNA name C06

PYTHON SYSTEM name sentinel
 param_number 1
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel
PYTHON ANTENNA name C08
PYTHON SYSTEM name sentinel
 param_number 1
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel
PYTHON ANTENNA name C09
PYTHON SYSTEM name sentinel
 param_number 1
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel
PYTHON ANTENNA name C10
PYTHON SYSTEM name sentinel
 param_number 1
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel
PYTHON ANTENNA name C12
PYTHON SYSTEM name sentinel
 param_number 1
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel
PYTHON ANTENNA name C14
PYTHON SYSTEM name sentinel
 param_number 1
 ###### Element in Command Queue sentinel
 INSERTING in Command Queue sentinel
====>Size of read is 350
we wrote on the socket 3 3-Apr-2015 14:45:24 sentinel set dmask ffff 5555
 Size of Struct is ######## 1638
we wrote on the socket 3 3-Apr-2015 14:45:24 sentinel set dmask ffff 5555
 Size of Struct is ######## 1638
we wrote on the socket 3 3-Apr-2015 14:45:24 sentinel set dmask ffff 5555
 Size of Struct is ######## 1638
we wrote on the socket 3 3-Apr-2015 14:45:24 sentinel set dmask ffff 5555
 Size of Struct is ######## 1638
we wrote on the socket 3 3-Apr-2015 14:45:24 sentinel set dmask ffff 5555
 Size of Struct is ######## 1638
we wrote on the socket 3 3-Apr-2015 14:45:24 sentinel set dmask ffff 5555
 Size of Struct is ######## 1638
we wrote on the socket 3 3-Apr-2015 14:45:24 sentinel set

dmask ffff 5555
Size of Struct is ######## 1638
we wrote on the socket 3 3-Apr-2015 14:45:24 sentinel set
dmask ffff 5555
Size of Struct is ######## 1638
we wrote on the socket 3 3-Apr-2015 14:45:24 sentinel set
dmask ffff 5555
Size of Struct is ######## 1638
we wrote on the socket 3 3-Apr-2015 14:45:24 sentinel set
dmask ffff 5555
Size of Struct is ######## 1638
Size of Response Struct => 4698
MCM => 1
3
3-Apr-2015 14:45:24
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 814 824 825 830 827 832 825 820 817 824 826 821 829 824 824 832 823 825 816 831 823 830 831 821 825 831 822 821 827 831 824 829 821 825 827 824 824 817 824 819 824 826 827 820 821 824 808 828 429 664 812 816 710 449 185 248 261 267 241 745 210 674 572 392 System validated : Sentinel
Command validated : Sentinal Controlling
Digital Mask : ffff 5555
Size of Response Struct => 4698
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
MCM => 1
3
3-Apr-2015 14:45:24
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 966 1916 14 1449 1141 857 825 827 826 827 820 821 829 828 830 823 829 822 820 827 824 828 826 822 825 832 826 825 826 830 826 827 823 825 820 823 830 830 828 826 823 822 823 825 824 824 827 828 786 827 823 828 828 825 825 822 826 822 826 973 841 824 832 251 System validated : Sentinel
Command validated : Sentinal Controlling
Digital Mask : ffff 5555
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
3
3-Apr-2015 14:45:24
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 843 818 825 820 818 814 813 835 812 830 824 825 816 826 820 824 832 812 817 813 818 825 814 826 827 820 813 822 822 811 825 818 827 828 822 825 826 825 811 829 823 824 819 826 815 827 833 810 832 510 220 109 330 755 536 737 1292 928 764 941 630 448 466 740 System validated : Sentinel
Command validated : Sentinal Controlling
Digital Mask : ffff 5555

Size of Response Struct => 4698
MCM => 1
3
3-Apr-2015 14:45:24
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 967 1914 8 1139 1486 891 823 821 826 820 820 827 824 824 826 824 828 827 828 828 824 823 825 824 821 826 822 826 821 824 826 823 821 828 816 828 821 830 822 826 826 827 822 826 824 822 824 826 824 824 825 826 829 825 824 822 823 815 828 996 840 827 230 766 System validated : Sentinel
Command validated : Sentinal Controlling
Digital Mask : ffff 5555
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
3
3-Apr-2015 14:45:24
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 970 1917 20 1502 1160 859 827 825 829 826 827 825 826 825 826 824 827 821 817 823 821 820 825 825 828 831 832 827 822 826 820 829 823 827 829 831 830 828 826 827 829 832 822 824 832 821 831 825 770 830 828 829 832 828 826 822 825 826 824 979 840 827 829 242 System validated : Sentinel
Command validated : Sentinal Controlling
Digital Mask : ffff 5555
Size of Response Struct => 4698
MCM => 1
3
3-Apr-2015 14:45:24
sentinel
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
NUmber of RESPONSE MSG is 3
No summery !! 963 1922 26 1479 1141 859 823 820 825 830 816 822 828 818 824 829 827 824 824 830 817 826 823 824 822 825 822 824 823 822 817 825 824 824 828 823 821 825 826 826 823 822 825 824 822 820 821 824 824 824 825 828 826 823 820 819 824 823 825 974 844 824 210 762 System validated : Sentinel
Command validated : Sentinal Controlling
Digital Mask : ffff 5555
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
3
3-Apr-2015 14:45:24
sentinel
NUmber of RESPONSE MSG is 3
No summery !! 972 1911 12 1486 1162 854 824 827 820 826 824 819 822 821 823 825 825 826 820 823 819 823 821 817 822 824 819 820 825 827 818 829 818 825 824 818 823 824 826 826 827 824 827 825 820 822 821 824 764 822 815 818 824 821 827 822 824 822 820 814 828

822 821 188 System validated : Sentinel
 Command validated : Sentinal Controlling
 Digital Mask : ffff 5555
Size of Response Struct => 4698
 MCM => 1
 3
  3-Apr-2015 14:45:24
 sentinel
  Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
 ###### NUmber of RESPONSE MSG is 3
No summery !! 960 1898 8 1144 1489 892 828 830 824 824 823 822 826 822 831 825 827 824 826 825 823 818 828 832 823 830 824 816 825 829 817 824 826 819 823 823 827 827 819 823 825 826 820 824 826 826 824 826 768 824 819 823 828 824 817 823 827 826 827 967 839 822 828 210 System validated : Sentinel
 Command validated : Sentinal Controlling
 Digital Mask : ffff 5555
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
 MCM => 1
 3
  3-Apr-2015 14:45:24
 sentinel
  ###### NUmber of RESPONSE MSG is 3
No summery !! 965 1912 12 1487 1145 860 820 822 820 824 824 828 821 827 827 820 825 828 821 828 823 828 826 830 816 823 830 825 829 825 822 828 819 822 826 828 822 817 819 819 825 823 827 829 826 820 820 823 825 820 825 823 828 829 828 830 821 815 826 963 835 812 209 768 System validated : Sentinel
 Command validated : Sentinal Controlling
 Digital Mask : ffff 5555
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
 MCM => 1
 3
  3-Apr-2015 14:45:24
 sentinel
  ###### NUmber of RESPONSE MSG is 3
No summery !! 1238 1356 942 1114 1740 1392 1301 1270 1140 1149 1008 787 1368 1417 1304 926 835 833 819 833 828 832 828 835 818 828 832 828 822 831 827 820 823 826 832 829 825 824 822 832 817 826 819 823 827 828 828 832 138 62 38 0 0 0 66 68 156 194 368 878 91 0 17 185 System validated : Sentinel
 Command validated : Sentinal Controlling
 Digital Mask : ffff 5555
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
====>Size of read is 1448
PYTHON ANTENNA name C00
PYTHON SYSTEM name fiber_optics
 param_number 1
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics

PYTHON ANTENNA name C01
PYTHON SYSTEM name fiber_optics
 param_number 1
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C03
PYTHON SYSTEM name fiber_optics
 param_number 1
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C05
PYTHON SYSTEM name fiber_optics
 param_number 1
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C06
PYTHON SYSTEM name fiber_optics
 param_number 1
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C08
PYTHON SYSTEM name fiber_optics
 param_number 1
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C09
PYTHON SYSTEM name fiber_optics
 param_number 1
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C10
PYTHON SYSTEM name fiber_optics
 param_number 1
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C12
PYTHON SYSTEM name fiber_optics
 param_number 1
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
PYTHON ANTENNA name C14
PYTHON SYSTEM name fiber_optics
 param_number 1
 ###### Element in Command Queue fiber_optics
 INSERTING in Command Queue fiber_optics
====>Size of read is 350
we wrote on the socket 4 3-Apr-2015 14:46:33 fiber_optics set
rf_attn 10 10
 Size of Struct is ######## 1638

we wrote on the socket 4 3-Apr-2015 14:46:33 fiber_optics set
rf_attn 10 10
 Size of Struct is ######## 1638
we wrote on the socket 4 3-Apr-2015 14:46:33 fiber_optics set
rf_attn 10 10
 Size of Struct is ######## 1638
we wrote on the socket 4 3-Apr-2015 14:46:33 fiber_optics set
rf_attn 10 10
 Size of Struct is ######## 1638
we wrote on the socket 4 3-Apr-2015 14:46:33 fiber_optics set
rf_attn 10 10
 Size of Struct is ######## 1638
we wrote on the socket 4 3-Apr-2015 14:46:33 fiber_optics set
rf_attn 10 10
 Size of Struct is ######## 1638
we wrote on the socket 4 3-Apr-2015 14:46:33 fiber_optics set
rf_attn 10 10
 Size of Struct is ######## 1638
we wrote on the socket 4 3-Apr-2015 14:46:33 fiber_optics set
rf_attn 10 10
 Size of Struct is ######## 1638
we wrote on the socket 4 3-Apr-2015 14:46:33 fiber_optics set
rf_attn 10 10
 Size of Struct is ######## 1638
we wrote on the socket 4 3-Apr-2015 14:46:33 fiber_optics set
rf_attn 10 10
 Size of Struct is ######## 1638
Size of Response Struct => 4698
 MCM => 1
 4
  3-Apr-2015 14:46:33
 fiber_optics
  ###### NUmber of RESPONSE MSG is 3
No summery !!   982 1928 32 1465 1152 863 828 828 826 830 828 832 825 830 828 825 830 830 828 824 830 824 828 828 828 832 825 829 830 826 830 832 826 826 830 833 825 828 824 830 827 830 825 831 830 829 830 828 770 830 830 821 828 824 830 827 832 825 828 985 844 830 829 267 System validated : Fiber Optics
 Command validated : Fiber optics Controlling
 RF Attenuation : 10 10
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
 MCM => 1
 4
  3-Apr-2015 14:46:33
 fiber_optics
  ###### NUmber of RESPONSE MSG is 3
No summery !!   979 1928 24 1501 1172 857 827 824 824 825 823 822 828 824 824 821 828 828 825 822 822 824 824 826 829 825 828 823 821 827 829 828 828 828 825 823 825 824 828 828 824 826 825 828 830 828 828 824 764 823 824 826 824 826 828 827 823 821 827 822 826

828 829 204 System validated : Fiber Optics
 Command validated : Fiber optics Controlling
 RF Attenuation : 10 10
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
 MCM => 1
 4
  3-Apr-2015 14:46:33
 fiber_optics
  ###### NUmber of RESPONSE MSG is 3
No summery !! 978 1909 27 1157 1494 896 824 824 825 821 830 824 824 827 825 827 828 824 825 824 823 826 821 828 819 828 829 827 825 825 828 828 824 826 827 825 825 822 824 821 828 821 828 825 832 828 826 825 770 826 829 830 828 829 827 827 828 824 824 980 845 823 827 221 System validated : Fiber Optics
 Command validated : Fiber optics Controlling
 RF Attenuation : 10 10
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
 MCM => 1
 4
  3-Apr-2015 14:46:33
 fiber_optics
  ###### NUmber of RESPONSE MSG is 3
No summery !! 823 821 812 811 826 810 830 822 825 824 830 828 820 829 822 835 824 823 827 824 827 825 823 832 820 832 826 832 822 825 827 832 829 834 834 812 832 820 830 824 825 832 820 832 834 826 819 824 196 279 555 556 801 1145 479 458 317 424 320 899 708 623 672 1225 System validated : Fiber Optics
 Command validated : Fiber optics Controlling
 RF Attenuation : 10 10
 Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
 MCM => 1
 4
  3-Apr-2015 14:46:33
 fiber_optics
  ###### NUmber of RESPONSE MSG is 3
No summery !! 984 1929 37 1492 1152 864 830 828 828 825 828 828 827 823 819 824 827 828 827 830 828 828 825 821 826 824 832 828 825 821 823 826 829 821 828 824 822 825 820 822 822 832 830 827 825 829 828 828 829 828 824 828 824 828 827 826 826 827 821 989 840 823 211 761 System validated : Fiber Optics
 Command validated : Fiber optics Controlling
Size of Response Struct => 4698
 RF Attenuation : 10 10
 MCM => 1
 4
  3-Apr-2015 14:46:33
 fiber_optics
  Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
 ###### NUmber of RESPONSE MSG is 3

No summery !! 976 1915 19 1492 1154 860 829 824 824 830 827 826 830 830 828 828 828 825 829 834 824 825 829 826 828 825 828 823 827 822 825 825 827 829 822 828 824 828 823 826 825 824 828 825 829 824 822 828 829 828 823 827 822 827 827 830 826 828 828 976 844 828 259 770 System validated : Fiber Optics
Command validated : Fiber optics Controlling
RF Attenuation : 10 10
Size of Response Struct => 4698
MCM => 1
4
3-Apr-2015 14:46:33
fiber_optics
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
NUmber of RESPONSE MSG is 3
No summery !! 985 1927 28 1509 1171 865 830 829 826 831 829 830 832 828 824 823 823 830 828 830 826 827 828 827 824 832 830 828 829 827 832 825 829 832 827 829 827 830 823 828 828 826 828 827 825 828 826 831 769 828 827 828 827 829 829 830 832 829 830 989 844 822 828 211 System validated : Fiber Optics
Command validated : Fiber optics Controlling
RF Attenuation : 10 10
Size of Response Struct => 4698
MCM => 1
4
3-Apr-2015 14:46:33
fiber_optics
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
NUmber of RESPONSE MSG is 3
No summery !! 978 1925 27 1154 1502 894 822 820 825 825 830 828 824 828 828 825 828 829 829 832 829 827 829 829 830 826 828 826 827 827 830 827 829 832 830 823 829 828 827 826 826 824 825 823 827 829 828 826 826 829 826 828 829 830 826 822 829 830 825 964 840 827 270 772 System validated : Fiber Optics
Command validated : Fiber optics Controlling
RF Attenuation : 10 10
Size of Response Struct => 4698
MCM => 1
4
3-Apr-2015 14:46:33
fiber_optics
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
NUmber of RESPONSE MSG is 3
No summery !! 830 820 827 837 827 835 824 818 830 833 829 827 832 832 832 827 828 825 829 825 821 822 828 827 823 827 832 830 827 828 825 825 830 826 829 823 828 827 828 829 830 828 827 835 836 828 839 836 808 520 252 175 174 135 239 412 441 704 691 1198 370 316 766 413 System validated : Fiber Optics
Command validated : Fiber optics Controlling
RF Attenuation : 10 10
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL
Size of Response Struct => 4698
MCM => 1
4

3-Apr-2015 14:46:33
fiber_optics
NUmber of RESPONSE MSG is 3
No summery !! 1433 1645 722 1002 712 1137 955 1404 1459 1510 1260 592 1165 779 1252 940 840 832 836 826 829 828 835 824 821 822 832 829 828 828 822 836 820 820 828 827 828 834 830 831 825 831 824 833 837 828 829 823 0 30 101 230 274 224 95 26 0 0 44 470 194 302 463 408 System validated : Fiber Optics
Command validated : Fiber optics Controlling
RF Attenuation : 10 10
Writing to ONLINE from SENTINEL THREAD SUCCESSFUL