Online_v2 – An upgraded control & monitor software for GMRT

Presenter : Raju Uprade

Date : 23/04/2015

Online_v2 Team :-

Santaji Katore **Charu Kanade** **N.G.Kantharia**

Deepak Bhong **Naresh Sisodiya** **S.Nayak**

Sachin sherkar **Mahadev Misal**

Raju Uprade **C Sateesh**

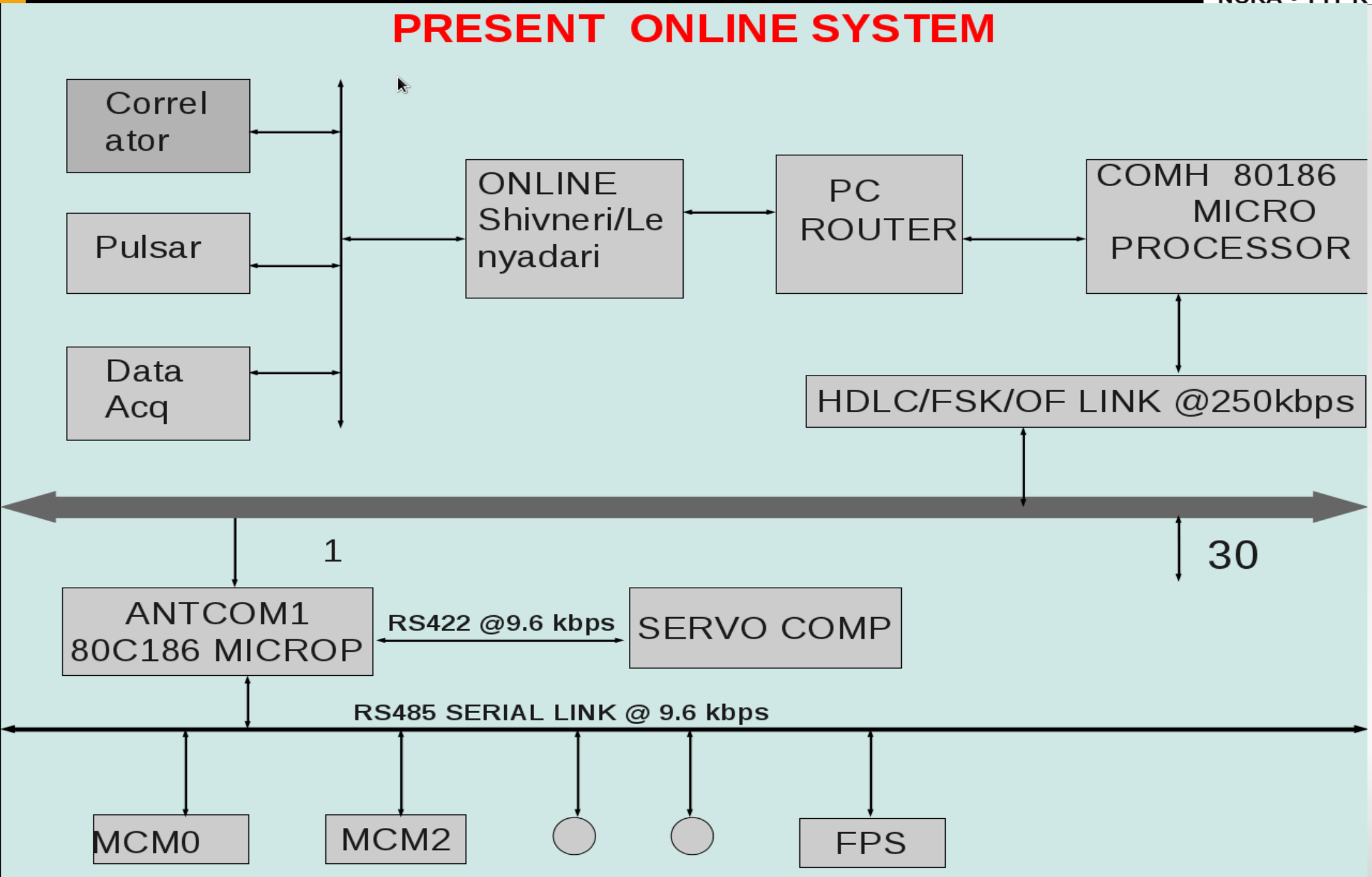
Outline of Presentation

NCRA • TIFR

1. Background
2. Architecture block diagram of Online_v2.
3. Online_v2 software
4. New MCM Rabbit card
5. QT/QML GUI
6. Python Enviornment
7. Monitoring Tools
8. Online_v2 testing
9. Antenna base configuration
10. Summary

Background

NCRA • TIFR



Background

Online_v2

• An expanded system using present day technology and other features of the GMRT upgrade is desirable.

• The hardware work for this was started a few years ago with the development of a new MCM.

• An upgraded version of ONLINE is currently under development at NCRA and is referred to as Online_v2.

• Online_v2 is Linux based and aims at reducing the RFI footprint at the antenna base by not requiring a separate computer.

• Instead Online_v2 focuses on exploiting the power of fast 1 Gbps Ethernet connection and in-built capabilities of the Rabbit processor on the MCM card.

•Online_v2 uses and expands the control algorithms developed for ONLINE on a new framework.

Features of Online_v2

The new features of OnlineV2 include :

(1) Parallel & fast control of 30 antenna

(2) Extensive web-based control data monitoring tools allowing for real time and statistical studies

(3) Full web based tools & support for observing in absentia

(4) Generalized framework to support future expansion

(5)Customized graphical interfaces for operators, engineers and astronomers

(6) Fast background monitoring of system parameters

(7) Higher level Python Environment

Difference between Online_v2 & Online

ONLINE V2 :

Linux
Absentee observing support
C, C++
Desktop GUI for Operator Engineer
MCM – Rabbit processor Based Card
Control data monitoring :
    Real time : shared memory
    Offline : From database
Web-based interfaces for Astronomer
Ethernet communication
Environment – python
No separate Antenna base computer
- reduction in internal RFI

ONLINE

Solaris
No absentee observing support
Fortran
No GUI
MCM – 8051 microcontroller
Control data monitoring -
    Real time – shared memory
    Offline – log file
No web-based components
Serial communication
Environment - AIPS
Antenna based computer – 8087
    microprocessor

Time Line

Project started: October 2012

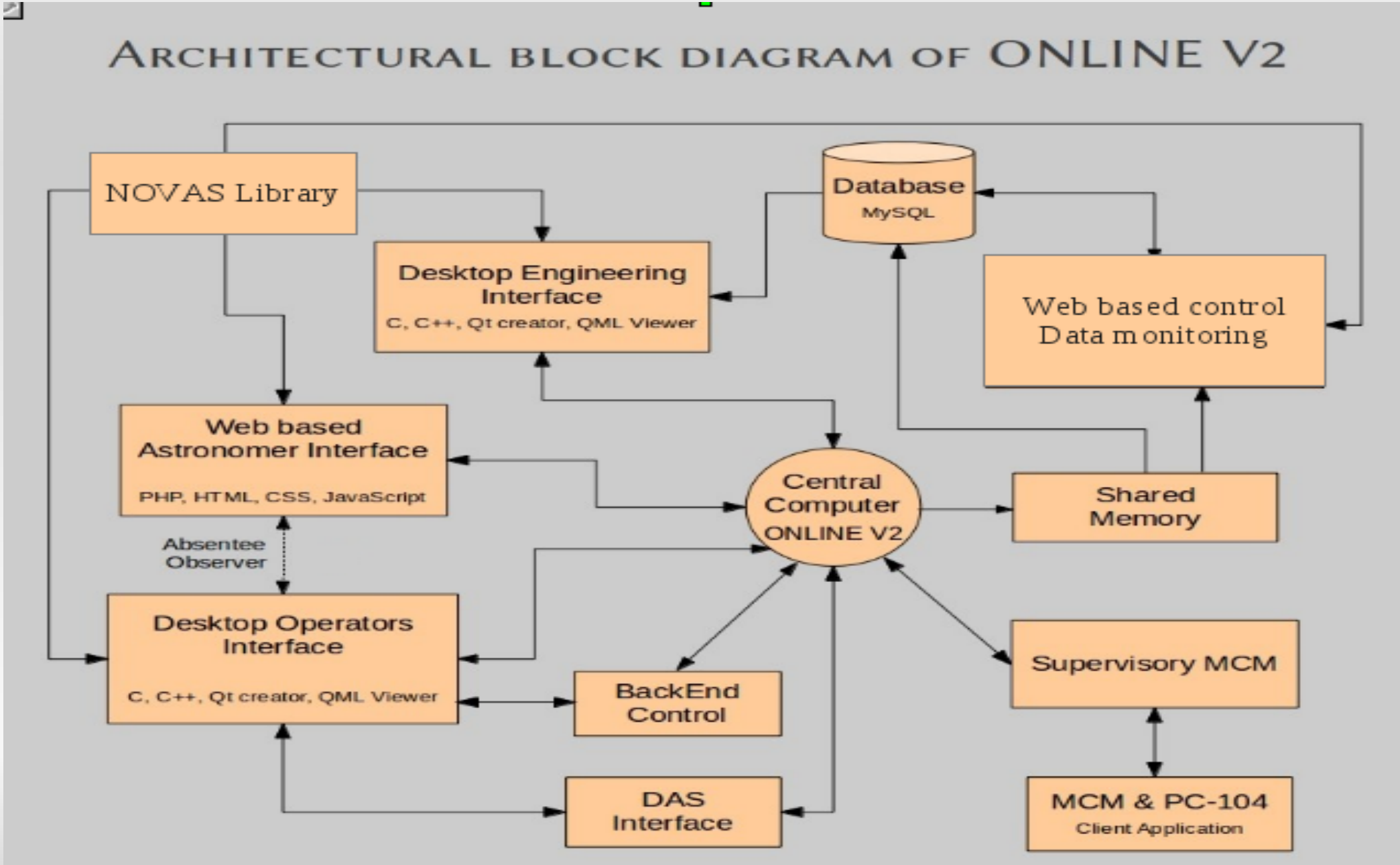
Prototype Demonstration: April 2014

Three antennae tested software: October 2014

16 Antennae tested software : April 2015

Online_v2 Architecture diagram

NCRA • TIFR



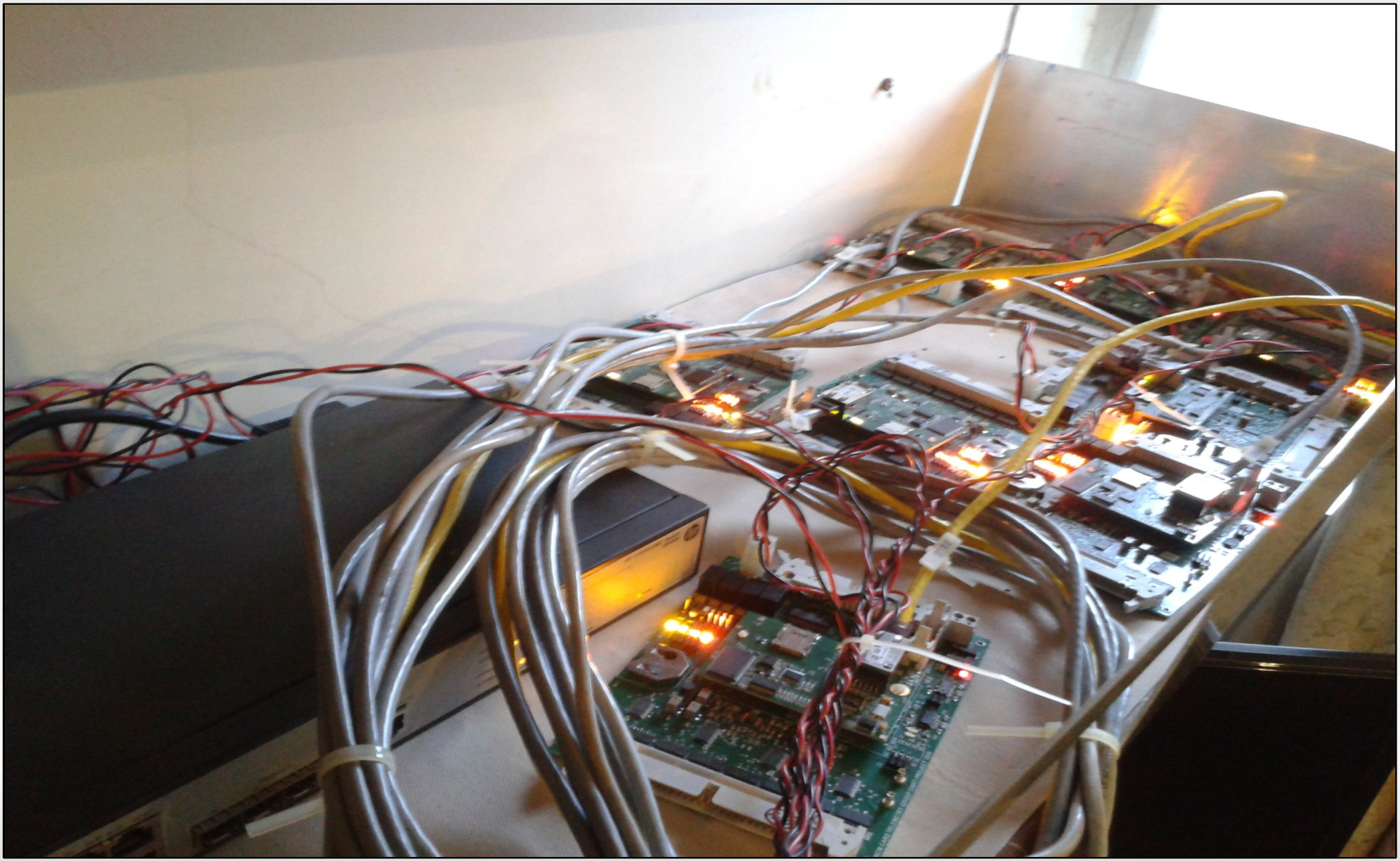
Online_v2

- Online_V2 is the New control & monitoring software being developed for GMRT.We have used experience of present Online, TELESET-ABCCOM & 15 M NCRA CMS in Online_V2.
- It is a multithreaded server based on Producer-Consumer principle where all antenna sub-systems act as client devices. Communications between server/client happen using TCP/IP based Ethernet communication channel with 1 sec response time.
- Online_V2 takes input through three input channel

  1. Python Environment 2. QML/QT based GUI 3. User terminal
- Online_V2 write all command & response in shared memory.
- Online_V2 also create a Message Queue for communication with DAS SERVER.
- Various commands related to user creation like create user,showuser, cmd2sub has been implemented and tested successfully.
- Online_v2 has successfully controlled & monitored all sub-systems in the lab & almost all commands for all sub-systems have bee implemented.
- Eight MCM and 1 PC104 cards 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.Basic servo commands have been successfully tested in Servo Lab with their test setup.

Online_v2 Lab set up

NCRA • TIFR



Online_v2 details

Software Languages used :

Insistence on using Open Software - LAMP

Software Language used: C, C++, Dynamic C, Perl, Python, PHP,HTML, Javascript, gnuplot, QT/QML

Database: MySQL

Libraries: XML, Readline, TCP/IP, HTTP,Pthread

Online_v2 core software

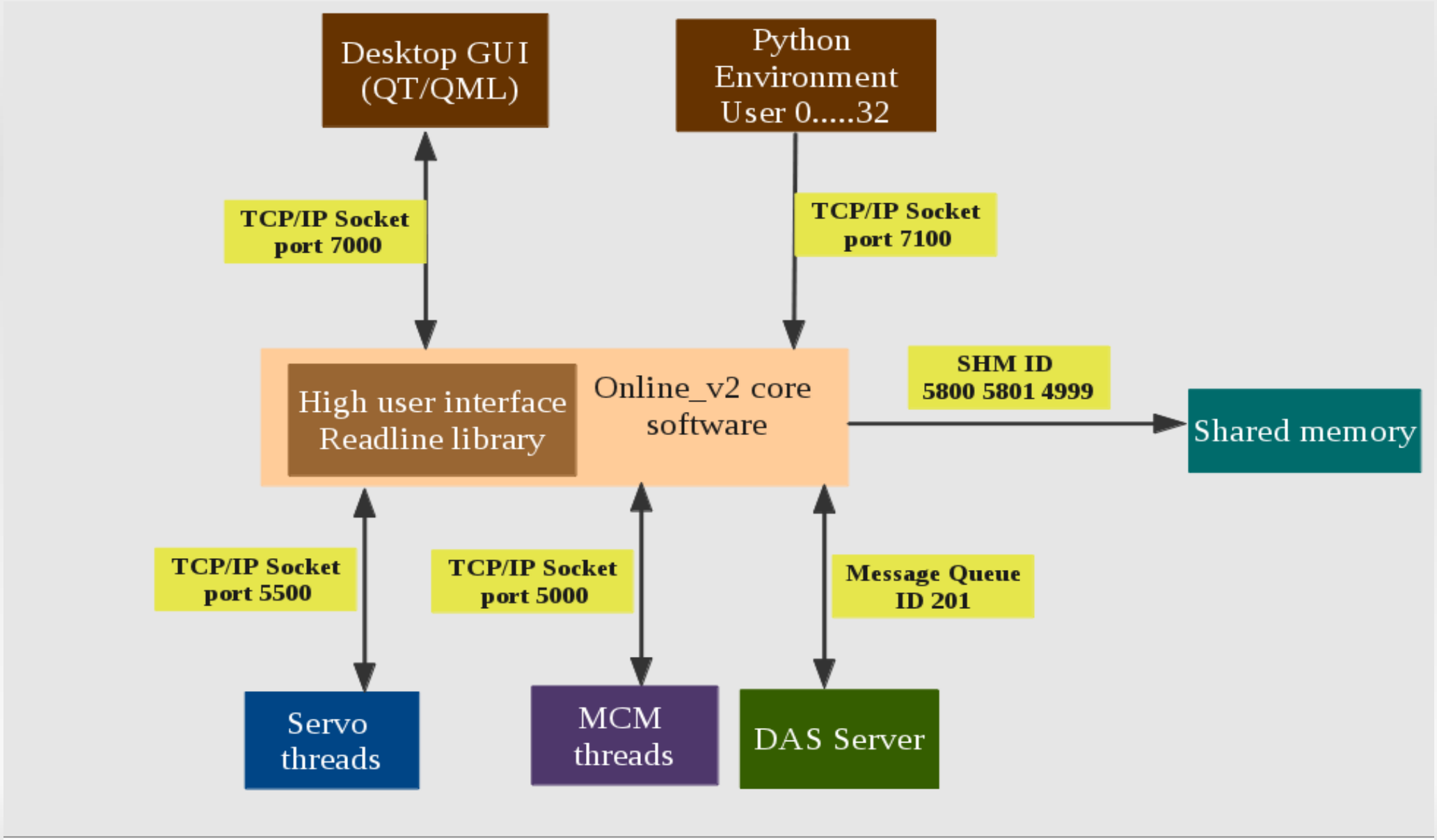


Fig : Block Diagram of Online_v2 core Software

Online_v2 command syntax

NCRA • TIFR

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

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

Online_v2 command syntax

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. 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	-

Online_v2 command syntax

NCRA • TIFR

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	-

Online_v2 communication protocol

In Online_v2 we are following a structure based communication protocol to communicate with antenna sub-system. We have two separate communication structure to communicate with MCM card & PC104 card.

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;
```

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;
```

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;
```

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.

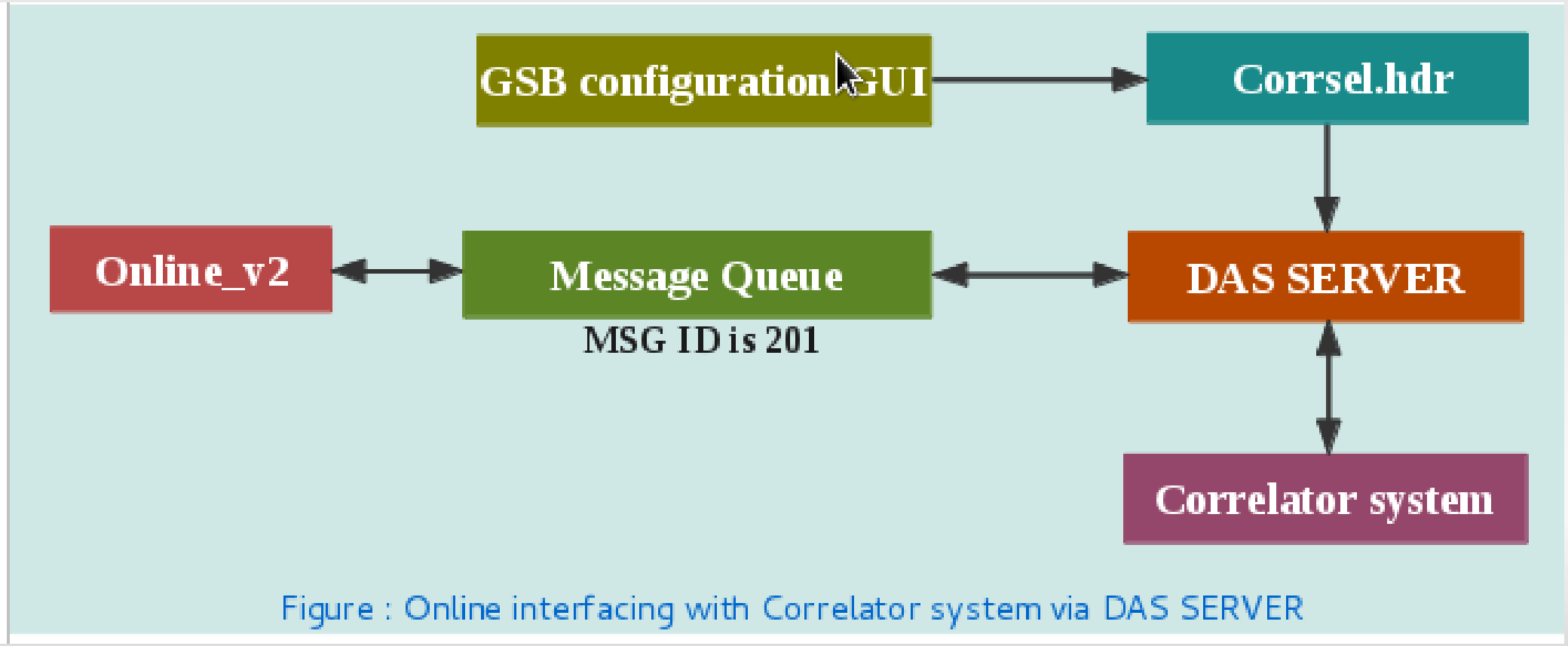
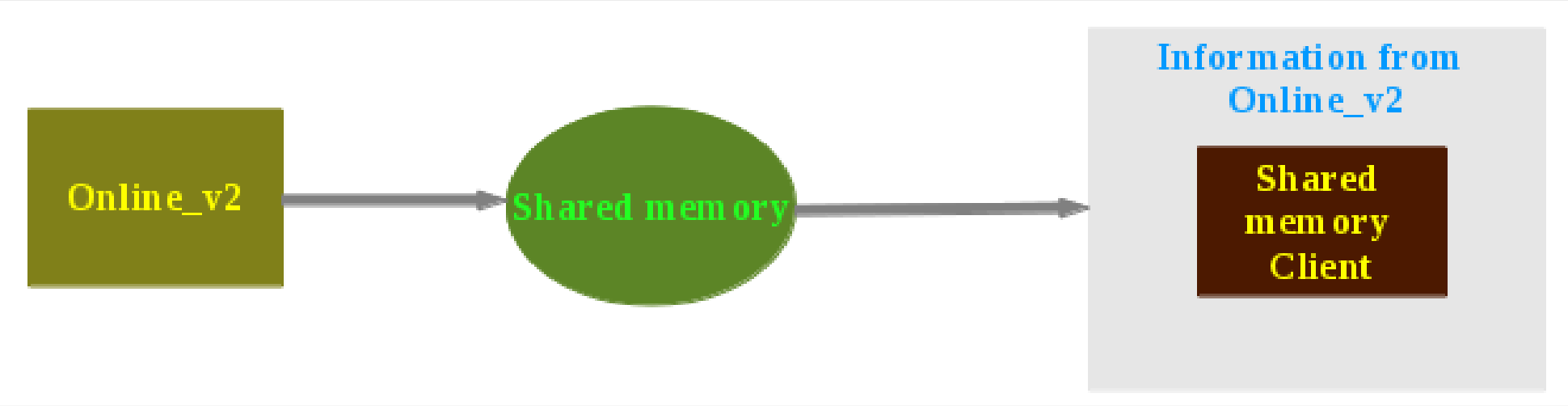


Figure : Online interfacing with Correlator system via DAS SERVER

shared memory

Shared Memory :

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.

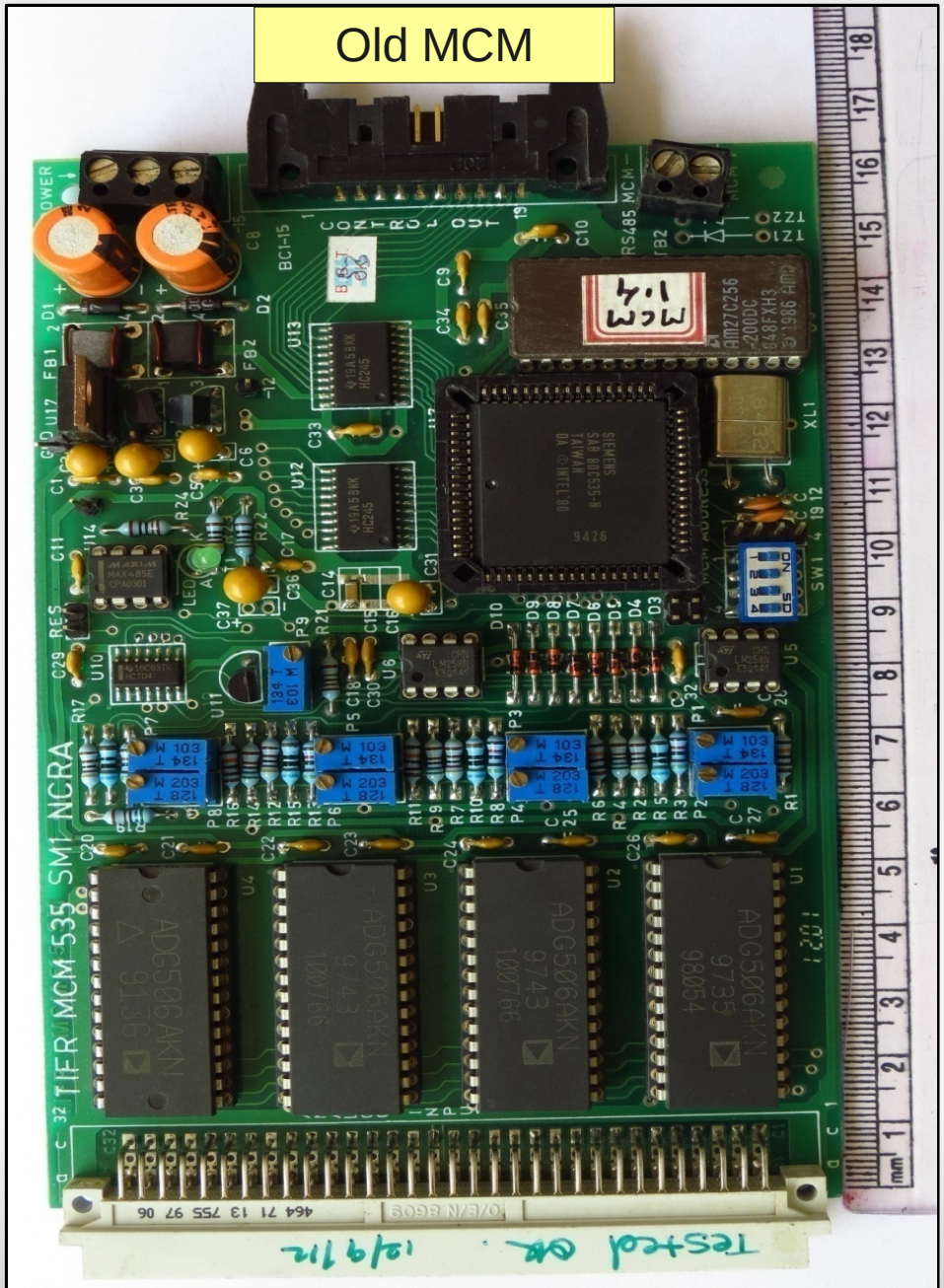


Monitor & Control Module (MCM)

NCRA • TIFR

New Rabbit MCM

Old MCM

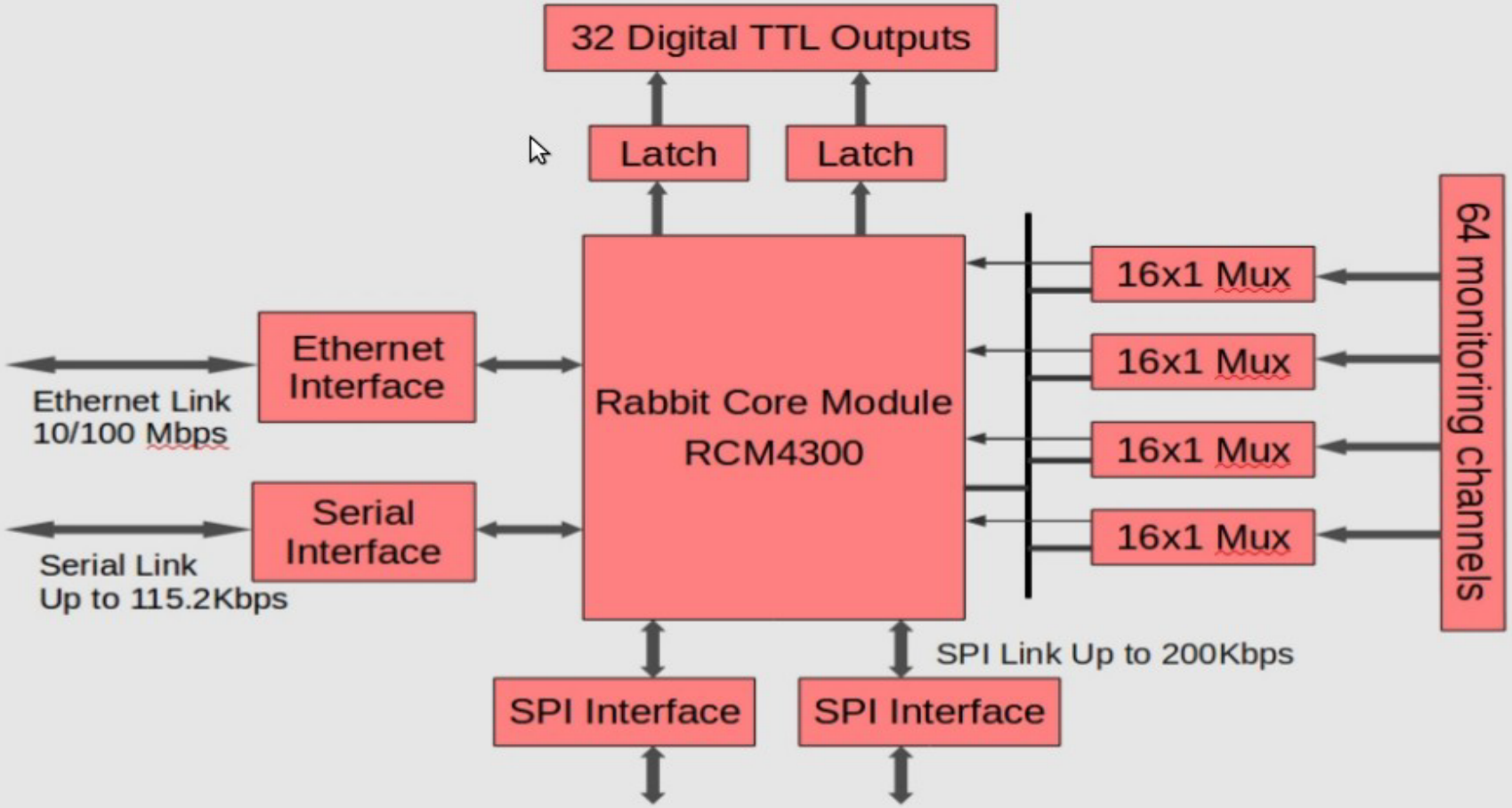


Performance Improvment in MCM

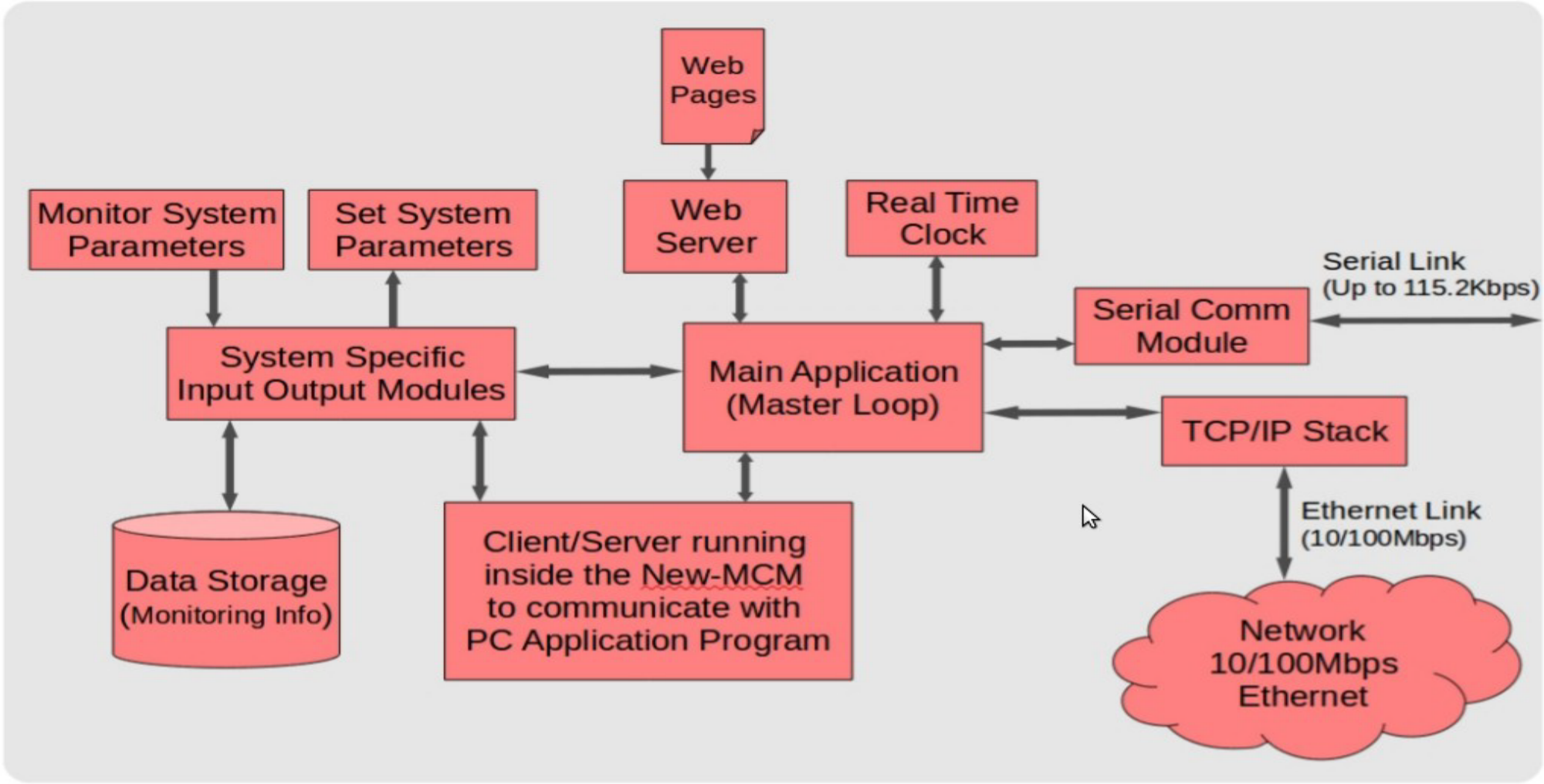
NCRA • TIFR

1. New MCM is 30 times faster than the existing one.
2. Higher program and data memory and 1GB storage
3. Higher ADC resolution
4. Double control bits
5. Ethernet, RS485 and SPI connectivity
6. High level programming language and method
7. 20dB less EMI
8. Additional features like : Real Time Clock, Battery Backup
9. ASCII structure based Command structure
10. Failure rate will be minimum

New MCM Hardware Architecture



New MCM Software Architecture



Rabbit MCM web server display

NCRA • TIFR

New MCM Monitoring Window

IP : 192.168.21.102 Antenna : C00 System : SENTINEL

64 MONITORING CHANNELS

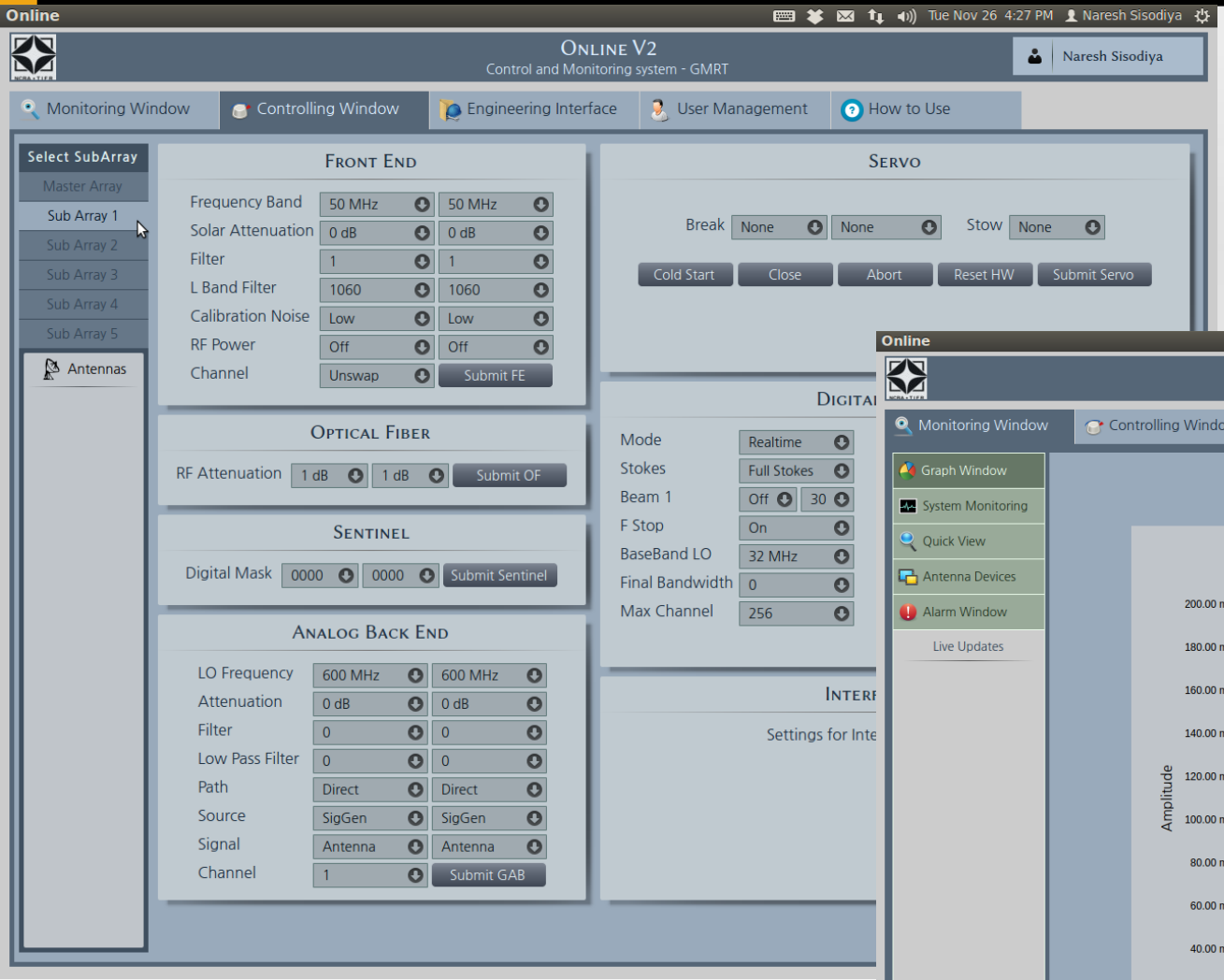
01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
977 0.01	1921 -4.80	35 4.82	1478 -2.54	1152 -0.88	862 0.60	827 0.78	826 0.79	830 0.77	828 0.78	828 0.78	830 0.77	832 0.76	820 0.82	828 0.78	825 0.79
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
831 0.76	832 0.76	824 0.80	830 0.77	828 0.78	827 0.78	829 0.77	827 0.78	823 0.80	827 0.78	824 0.80	822 0.81	830 0.77	831 0.76	832 0.76	822 0.81
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
826 0.79	826 0.79	825 0.79	824 0.80	830 0.77	829 0.77	822 0.81	824 0.80	829 0.77	824 0.80	824 0.80	830 0.77	829 0.77	826 0.79	820 0.82	827 0.78
49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
828 0.78	826 0.79	828 0.78	826 0.79	828 0.78	826 0.79	826 0.79	826 0.79	832 0.76	830 0.77	830 0.77	723 1.31	828 0.78	827 0.78	816 0.84	822 0.81

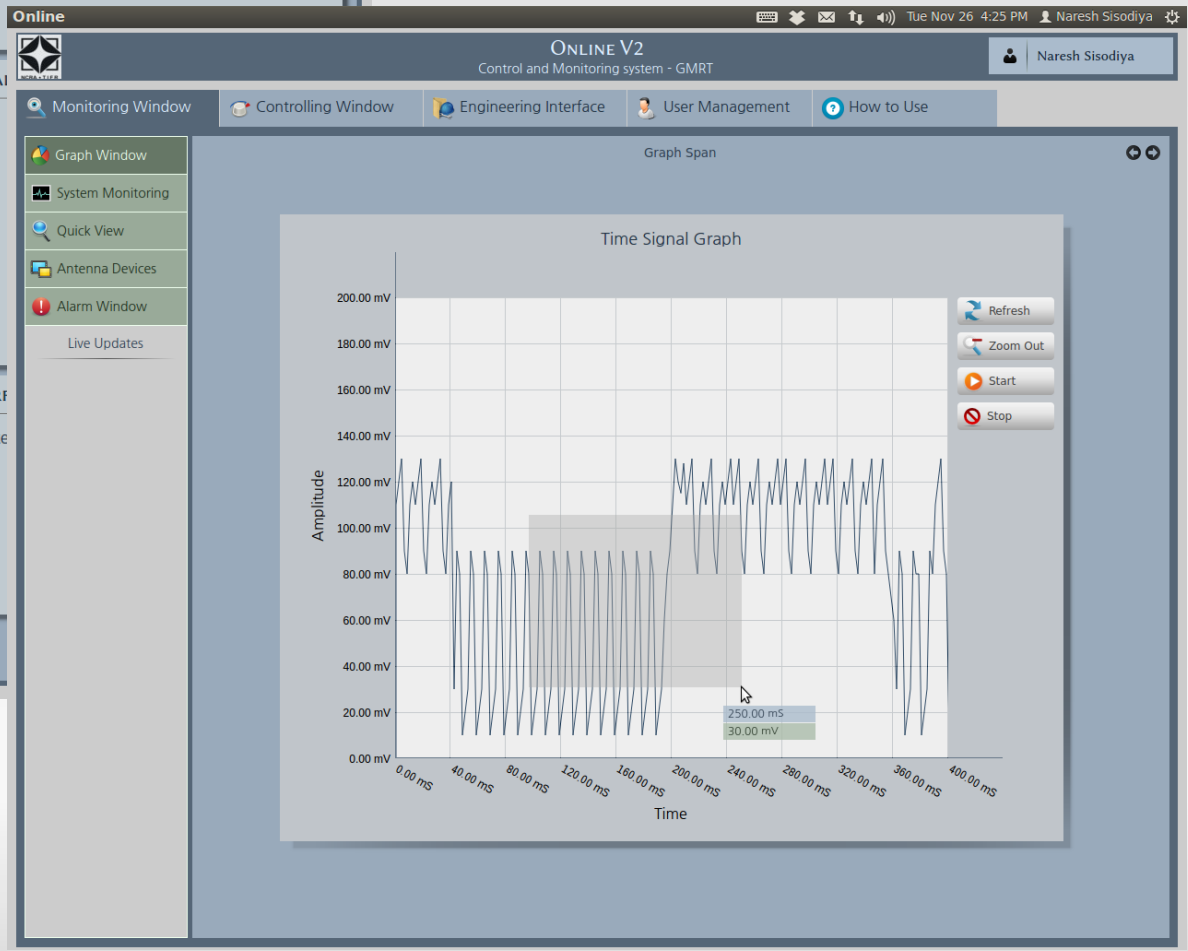
MCM STATUS

Spectrum Spreader	Normal (0 to 50MHz)	Normal (Above 50MHz)
Frequency Doubler	On	
Frequency Divider	1	
MCM Frequency	60 MHz	
Real Time Clock	07 - 03 - 2015	11 : 49 : 12
Digital Mask	0000	0000

SENTINEL SYSTEM STATUS

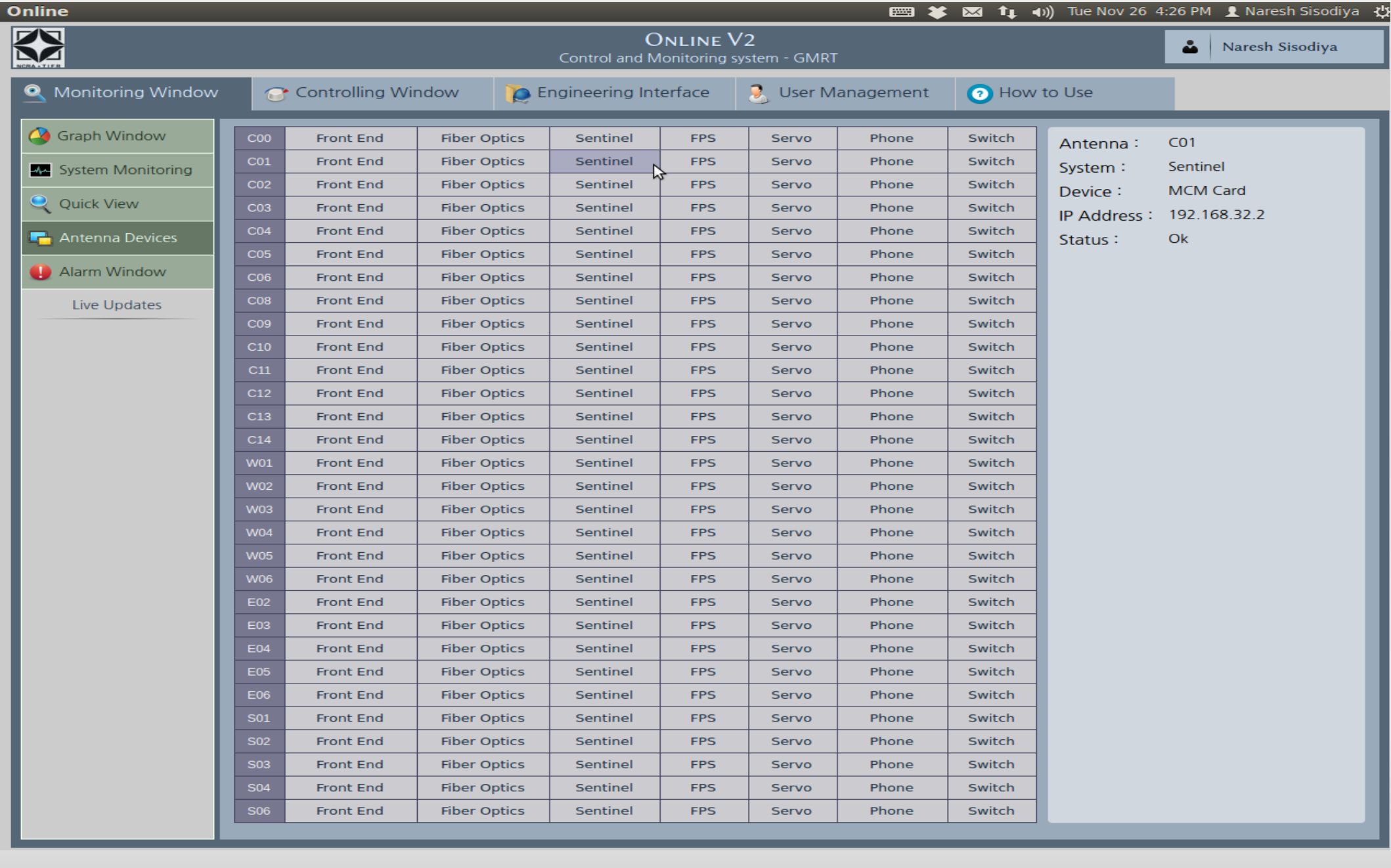
Online_v2 : QT/QML GUI





Online_v2 : QT/QML GUI

NCRA • TIFR



Python Enviornment

cms@incms: ~/pyenv (on incms)

File Edit View Search Terminal Tabs Help

cms@incms: ~/pyenv | cms@incms: ~/pyenv | cms@incms: ~/pyenv

```
user2@GMRT [10]: set_of
---------------> set_of()
{}
12
C01C04C06
rf_attn
user2@GMRT [11]:
Do you really want to exit ([y]/n)?
cms@incms:~/pyenv$ user2
GMRT ONLINE COMMAND LINE USER INTERFACE
user2@GMRT [1]: connect
--------------> connect()
Connecting as user: 2
user2@GMRT [2]: ante = ['C01','C04','C06']
user2@GMRT [3]: defsub( 2,ante)
user2@GMRT [4]: set_of(rf_attn=(12,12))
rf_attn = (12, 12)
{'rf_attn': (12, 12)}
12
C01C04C06
rf_attn
(12, 12)
(12, 12)
user2@GMRT [5]:
```

Online_v2 monitoring tools

NCRA • TIFR

a) Data collection on the data server.
-> A 'C' program is running on the central server
-> Collect data from shared memory
-> sends the data to data monitoring server.
-> Data size : 1 GB per day per MCM card.

b) Data Display on Web:
-> An Apache web server is running on the monitoring server
-> web display
-> client -> Java-script
->Server side -> PHP script
-> Selection of MCM card is provided.

Online_v2 monitoring tools

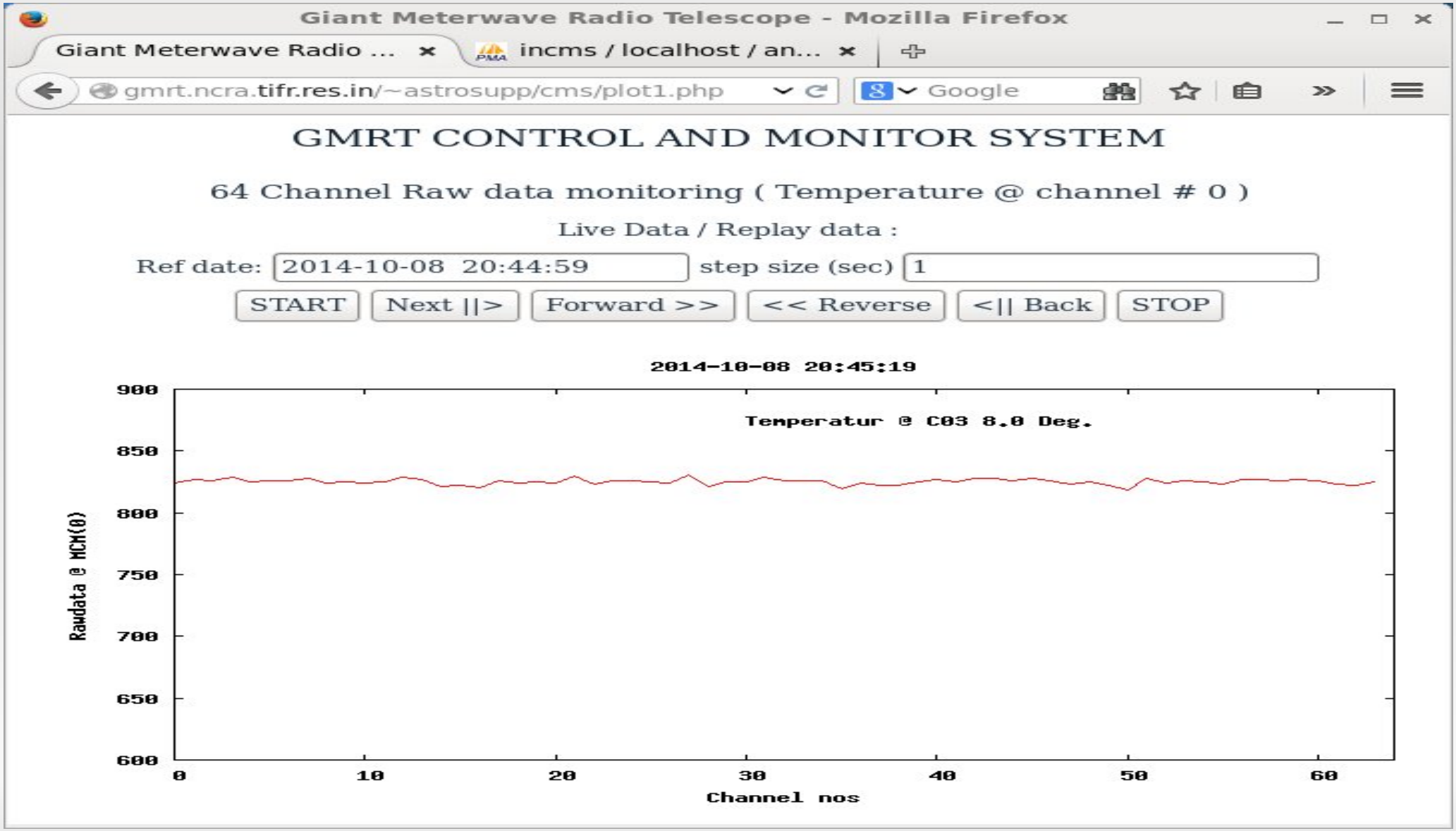


Fig. web interface for raw data monitoring

Online_v2 monitoring tools

c) Temperature monitor test:

One temperature monitoring unit was connect to the MCM card at C03 antenna. The data behavior was similar like existing online with higher resolution.

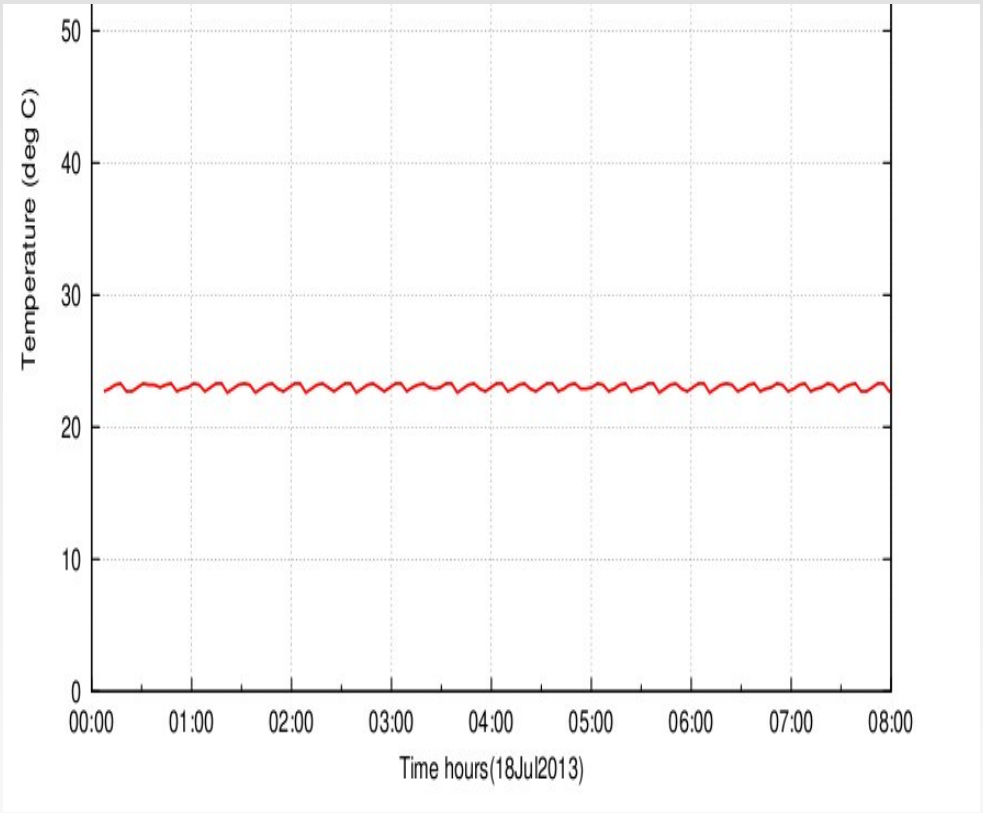


Fig. existing online temperature monitoring @ C03 antenna

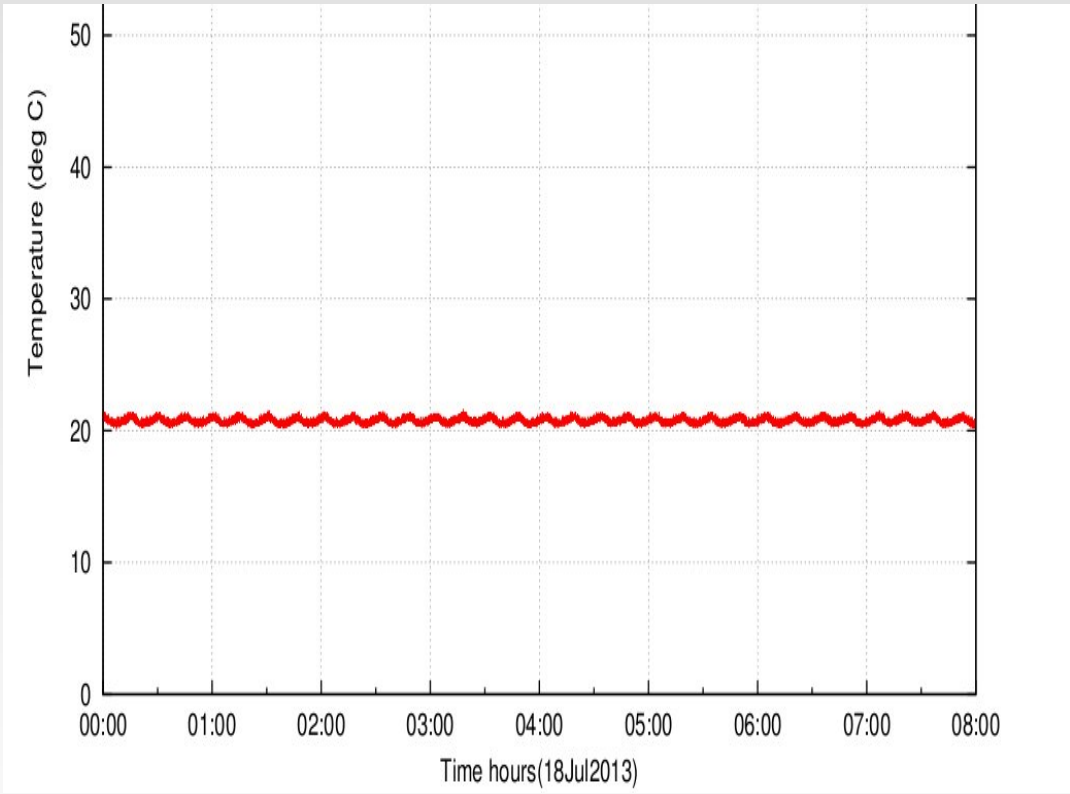
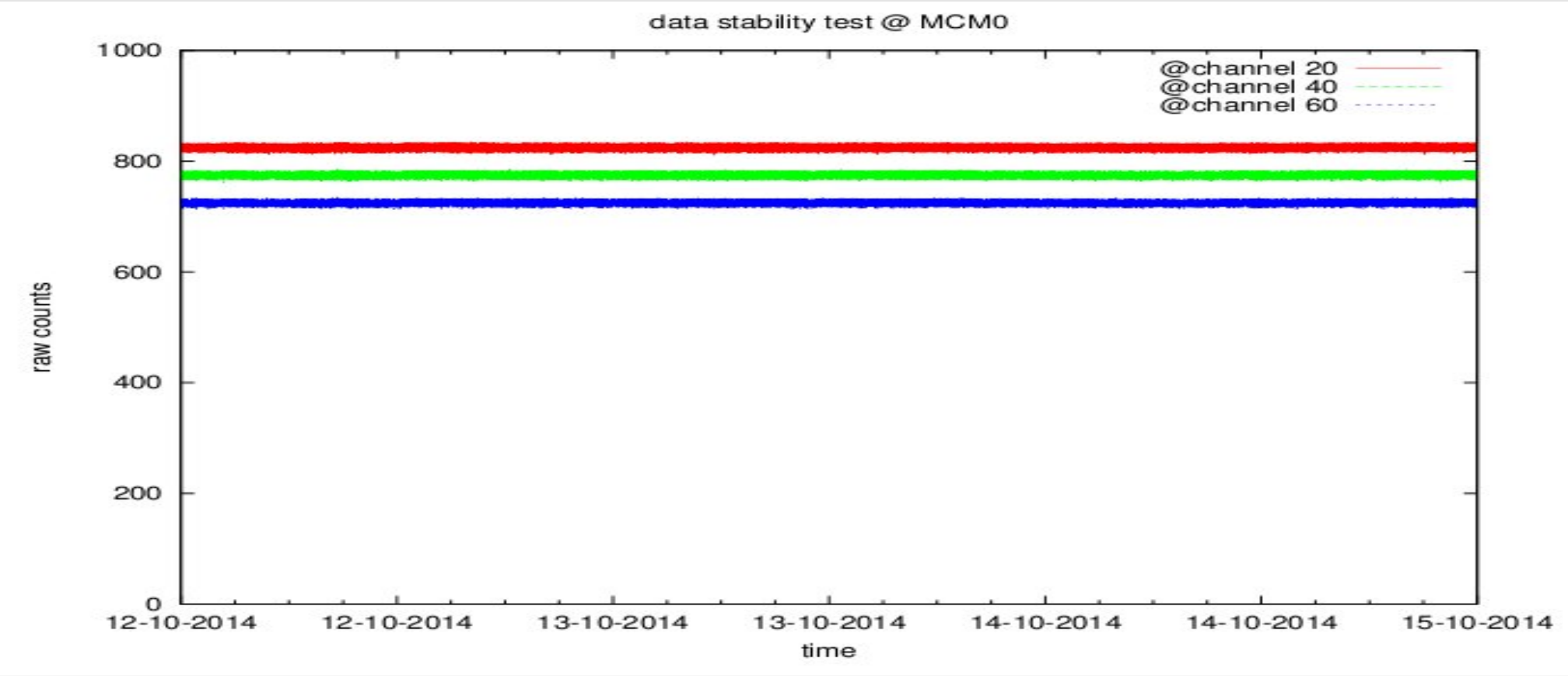


Fig. online_v2 temperature monitoring @ C03 antenna

Online_v2 monitoring tools

d) Raw data recording test:

Long period raw data recording test was conducted to check the stability or any dropouts. Here three different plots indicates three channels 20,40 and 60. Offsets are purposely added to avoid overlap.

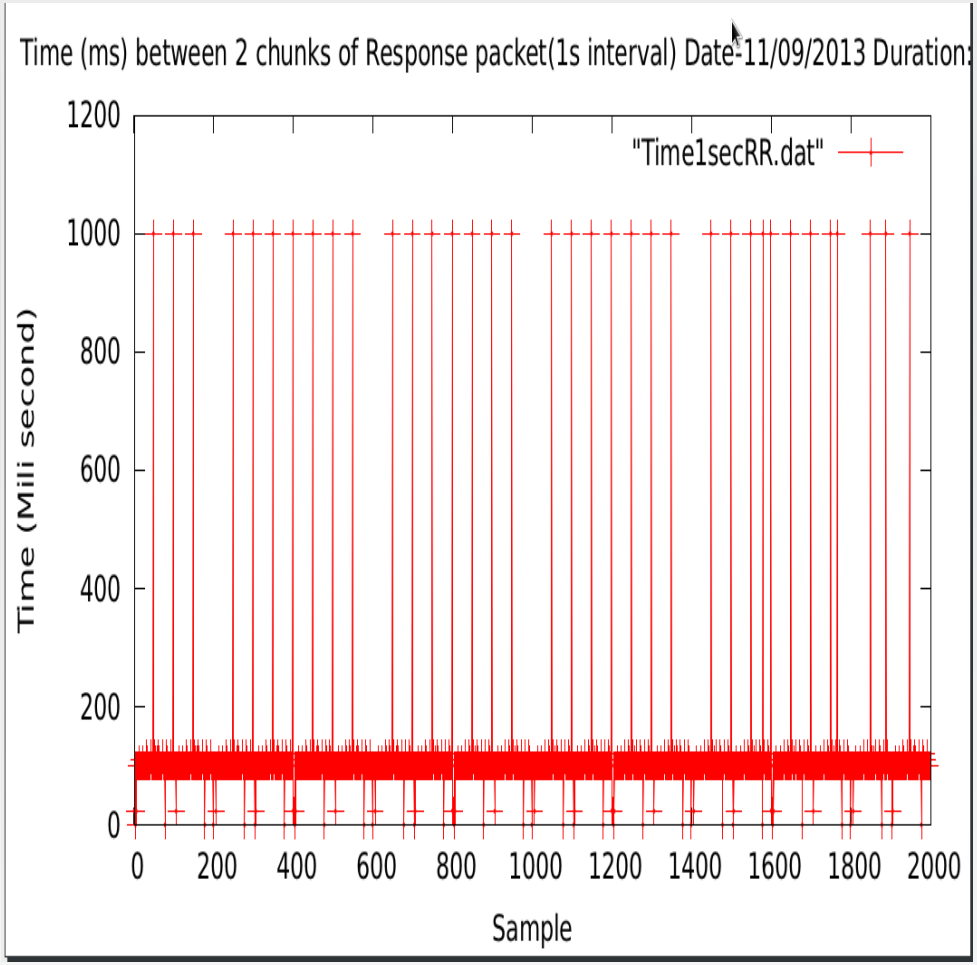


As per the Online V2 design plan, we have successfully implemented and tested the data base for the raw data monitoring. This data can be easily accessed through the web interface. Long period recording test is successfully conducted. Temperature monitoring results are also matching with existing online.

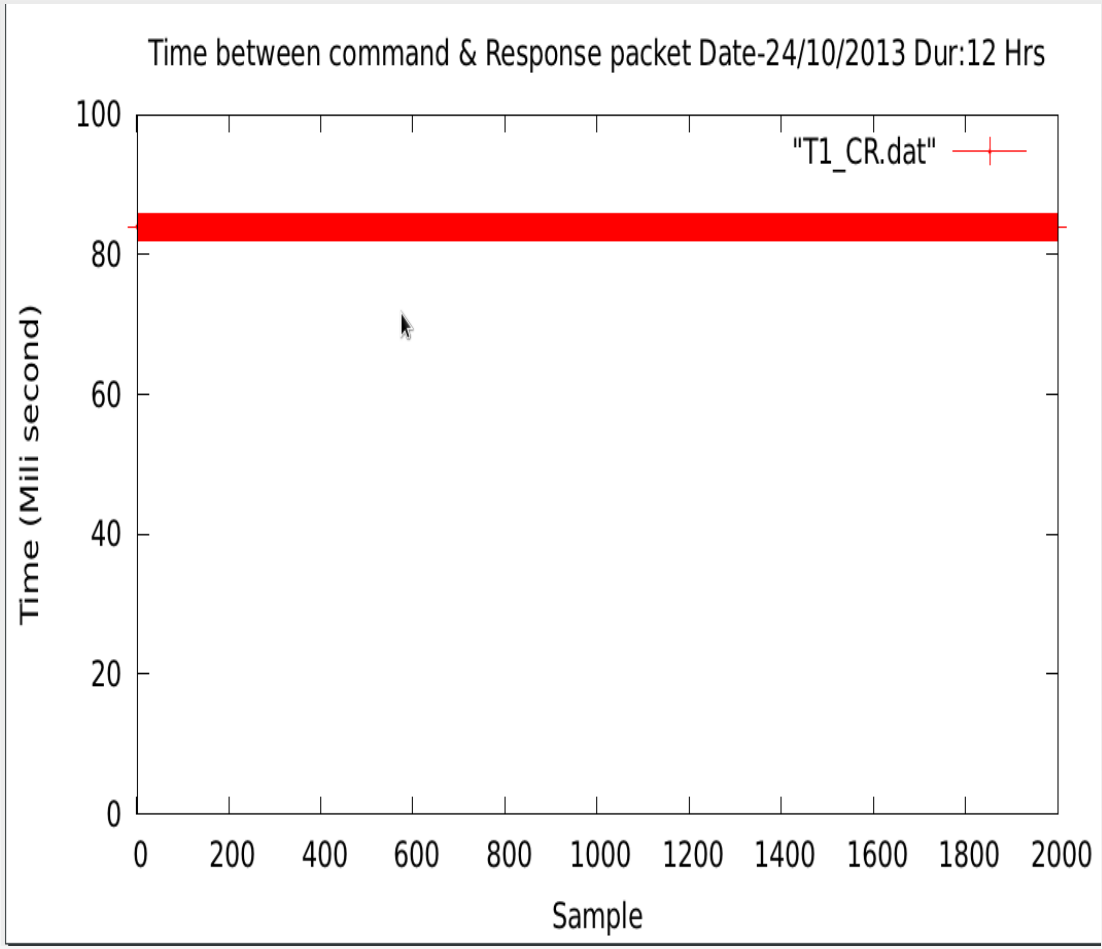
Online_v2 Testing

- New cisco L3 network switch has been configured for Lab setup. Eight MCM card with different subnet have been connected to the switch.Online_V2 is communicating with every MCM within 1 second reponse time.
- We have done several test to measure the round trip time taken by command & response packet.
- Front end,GAB,FPS system has been tested in respective labs.
- MCM cards were installed at C01,C04,C06,C09 & C10. They communicated very well with Online_v2, sending monitoring response every 1 second time interval.
- Servo system PC104 card has been tested using Online_v2 in C01 antenna.
- C03 rabbit MCM card was connected with a temperature sensor which was sending Antenna shell temperature every 1 second.
- GWB correlator has been successfully interfaced with Online_V2 through GPU DAS server. All commands were tested.
- During April 2014 MTAC, We installed a L2 N/W switch in C06 antenna and connected two MCM cards configured as Sentinel and Optical fiber system. And successfully controlled and monitored both the system.
- FPS system has been tested with Online_v2 over Rabbit card serial link in C06 antenna
- During October 2014 MTAC, we tested three antenna C01, C04 & C06 two sub-system(OF+Sentinel) software.
- During April 2015 MTAC we tested 16 antennae two sub-system(OF+Sentinel) using Online_v2 software.

Round time trip plots

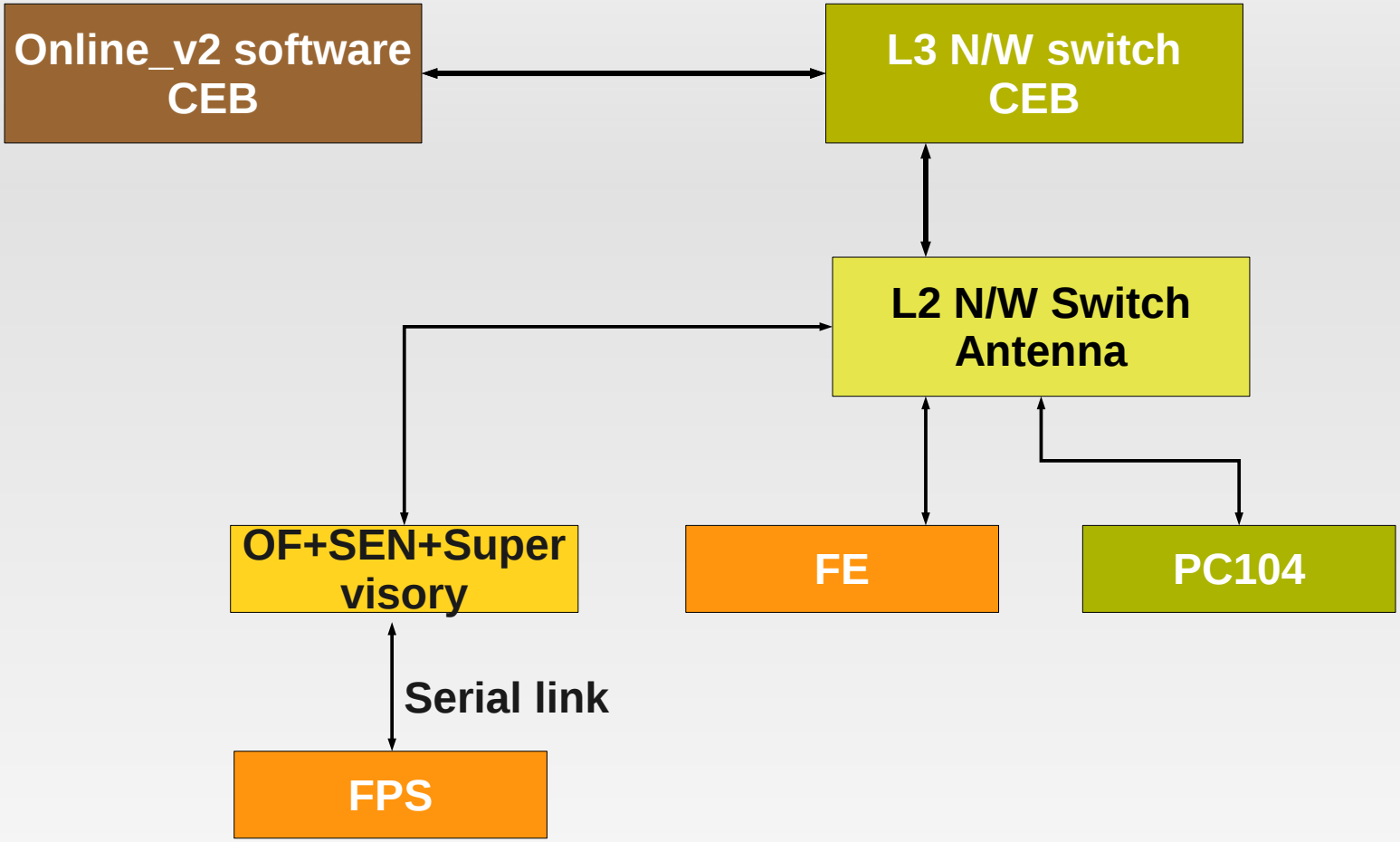


Fig(a) : When connected to
Common network &
8 port N/W switch



Fig(b) : When connected to
Isolated sub net network & basic
L3 24 port N/W switch

Antenna base configuration



** RFI shieding is provided in form of boxes for the MCM & L2 switches.

Summary

NCRA • TIFR

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. Online_v2 has been thoroughly tested in Telemetry Lab for past two years and it runs without any problem.
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.

Ongoing Development

- Interfacing NOVAS library in ONLINE_v2 software.
- GUI to Python interface.
- Completing the QT/QML GUI.
- Testing of Online_v2 software on Antennae and all sub-systems when avilable.
- Rabbit MCM to FE system serial link development.
- Performance testing and optimization of Online_v2.

Acknowledgement

- Prof. Swarna Ghosh
- Prof. Yashwant Gupta
- Prof Jayaram Chengalur
- Shri. R Balasubramaniam, Shri Pramesh Rao
- OF Lab (Suresh Kumar,Pravin,Satish L,Sanchit R)
- FE Lab (Anil Raut,Gaurav P,Sougato,Vishal T)
- GAB Lab (Ajit,Navnathh, Ajay,Abhijeet, Shri Hari)
- FPS Lab (Suresh S, Abhay bhumkar)
- Servo Lab (S bagde,Thiyagu)
- Correlator (Ajit, sandeep,Harsha,sanjay K)
- Control Room (Nilesh,Shilpa,Pramod)
- Telemetry Lab's semi skilled workers (Bharat Sethe & Anil Muley)