VIRGO Final Design

5000 Electronic and Software

May 18, 1995

5000. Introduction

This chapter describes all the needed activities starting from the output of the local electronic up to the data ready to be analyses for Gravitational Waves search. It covers four fields:

• The online activity. It concerns the controls of the interferometer, the acquisition, online processing, and monitoring of the data produced by the various detectors or control process.
• The detector simulation activity. It concerns the generation of gravitational events like coalescent binaries, pulsars, supernovae and the simulation of the response of the interferometer to such signals and to all possible noises.
• The offline activity. It concerns the basic tools for the refined analysis of the large amount of collected data.
• The detector commissioning and calibration activity. It concerns the strategy to check parts or all of the system and the calculation of all calibration parameters.

A coherent view of all these activities will optimize the design, the commissioning, the running and the understanding of the first results produced by the interferometer. In addition to the discution of these four fields, the standardization and rules, the hardware implementation and the cost estimate will be also presented.

Most of the solutions presented here have already been described in the Virgo note PJT93-16 and in its updated version from September 94.

5000.1. The online Architecture

The various active parts of the detector which produce digitized information or 'data' are represented on figure 5000.1. Most of the signals produced by the different sensors are processed by a layer of local controls which compute and apply local corrections to keep the controlled elements (laser, mirror, vacuum pump,..) within a given set of tolerances. These controls produce data like status informations or feedback values. Several local controls may be supervised by a higher level control, the Global Control in charge of the locking or the alignment .

The data generated by these detectors and controls (the 'raw data') are collected by a local readout process, concentrated and structured by an Frame builder and written to tape (Data archiving). They are processed ('reconstructed') to convert ADC counts and feedback signals to an h value, and 'filtered' to reduce their amount to a level manageable by the offline analysis. The data quality is permanently monitored by surveying the noise level and a known signal produced by a calibration device stimulating permanently the interferometer as a gravitational wave would do it. Finally, the 'filtered' data are send to a storage and distribution system which is the experiments front end for the offline analysis. The system provides also tools for histogramming and data editing (including monitoring data).

All these controls are coordinated by a Supervisor and synchronized with a central Timing system.

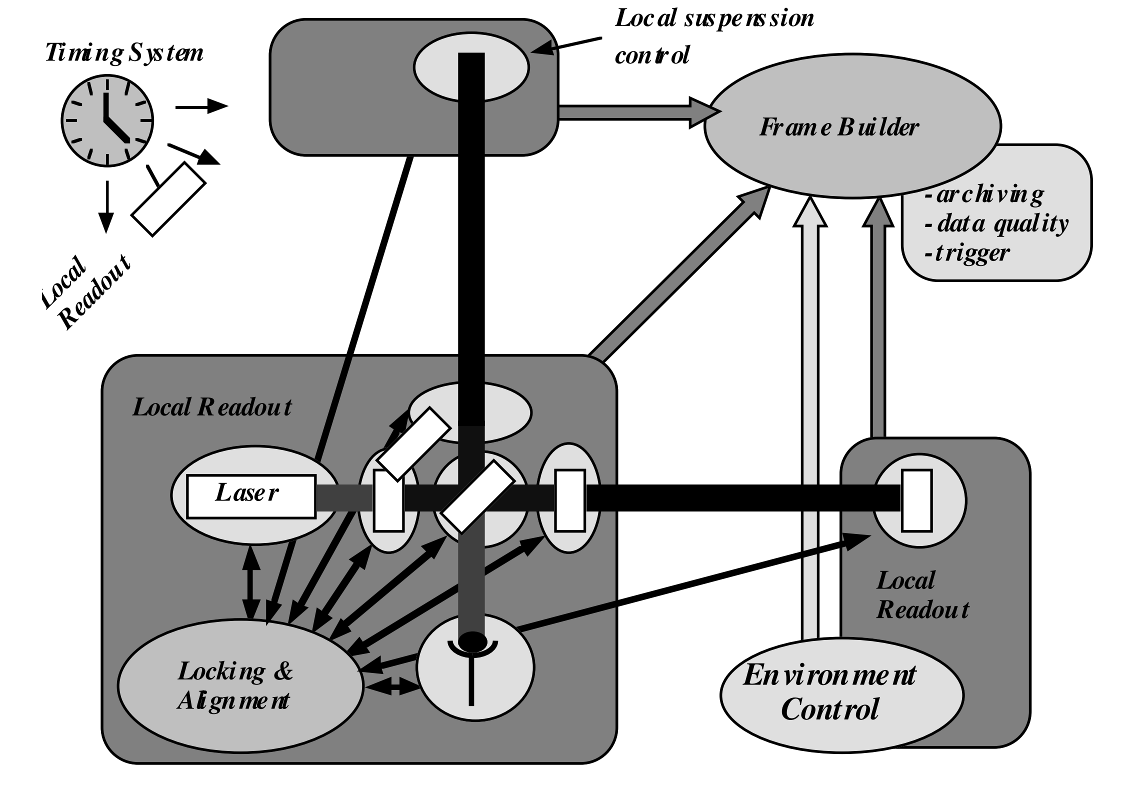

figure 5000.1 The online architecture

5000.1.1. Specifications:

The main specifications for the online system are:

- a real time control system with fast decision taking processes and correlation capability (up to 20 kHz) all over the site,
- a data acquisition system with a maximum sampling frequency of 20 kHz and a data rate up to 10 MBytes/sec,
- a dead time between the user interface and the real time process lower than one second,
- a system easy to reconfigure and to use on small test bench,
- test capability at all levels.

5000.1.2. The architecture

The interferometer is kept at its working point by a set of controls which process data provided by captors and adjust accordingly its main components: the laser, the two mirrors of each Fabry Perot cavity, the end mirror of the mode cleaner, the recycling mirror, the beam splitter, the injection and detection benches.

The architecture of the control and read out systems has to account for the large distances which may separate two components. To preserve flexibility in the interferometer setting up, the control systems are organized as much as possible into independent units in charge of the adjustment of a well defined component. They are run locally in standalone mode and accessed remotely by some high level control process.

Similarly, the readout systems are implemented into independent units in charge of the concentration of the data produced near the components located in a same building: end mirror west and north, mode cleaner, and main buildings. A central data acquisition system, located in the control room, collects and assembles these data before writing them to tape.

The knowledge of the precise timing of the various measurements and actions performed around the interferometer is one of the key points of its operation as a gravitational wave detector. This is implemented with a central timing system, located in the control room, and set up to distribute a well defined clocking sequence all over the site.

5000.1.3. The hardware options

To achieve a few khz bandwidth on the interferometer sensitivity, one has to design its various servo loops with a much higher bandwidth in the range of 10 kHz. As a consequence, one has to implement the system with ‘fast’ captors, processors and actuators connected with ‘fast’ links. To execute the various feedback loops within a constant time the system has to be build with conflict free accesses.

Sensors, processors, and their actuators sampled at those high rates are thus implemented within the commercially available VME standard. High transfer rate digital servo loop are implemented with a dedicated bus. Conflictual bus accesses are avoided by housing only exclusive controls in the same crate. To keep proper track of all the measurements only digital informations can be exchanged. Higher level control process transfer their informations from building to building via a Digital Optical Link (DOL). Short distance transfers between different controls are performed through mirrored memories connected on a local vertical bus or using a DOL.

The environment status is generally measured with captors sampled at a low rate. They are thus implemented in VME or G64, depending on their availability and cost. They are read out by slow monitoring systems and the data are exchanged using a dedicated ETHERNET network (the slow monitoring network) extending all over the site.

The data sampled at high rates are collected by the local fast readout systems. Then they are transferred to the main control room by a digital optical link.

All the processors are networked via ETHERNET or FDDI and accessible for control and file exchange by all workstations. Figure 5000.2 shows the networking which has to be implemented between the main building, an end mirror building and the control room. The basic rule is that hardware links are used for any real time connections while computer networks are used for state control and for slow monitoring.

The use of standard hardware is mandatory since it allows easy reconfiguration and long term maintenance.

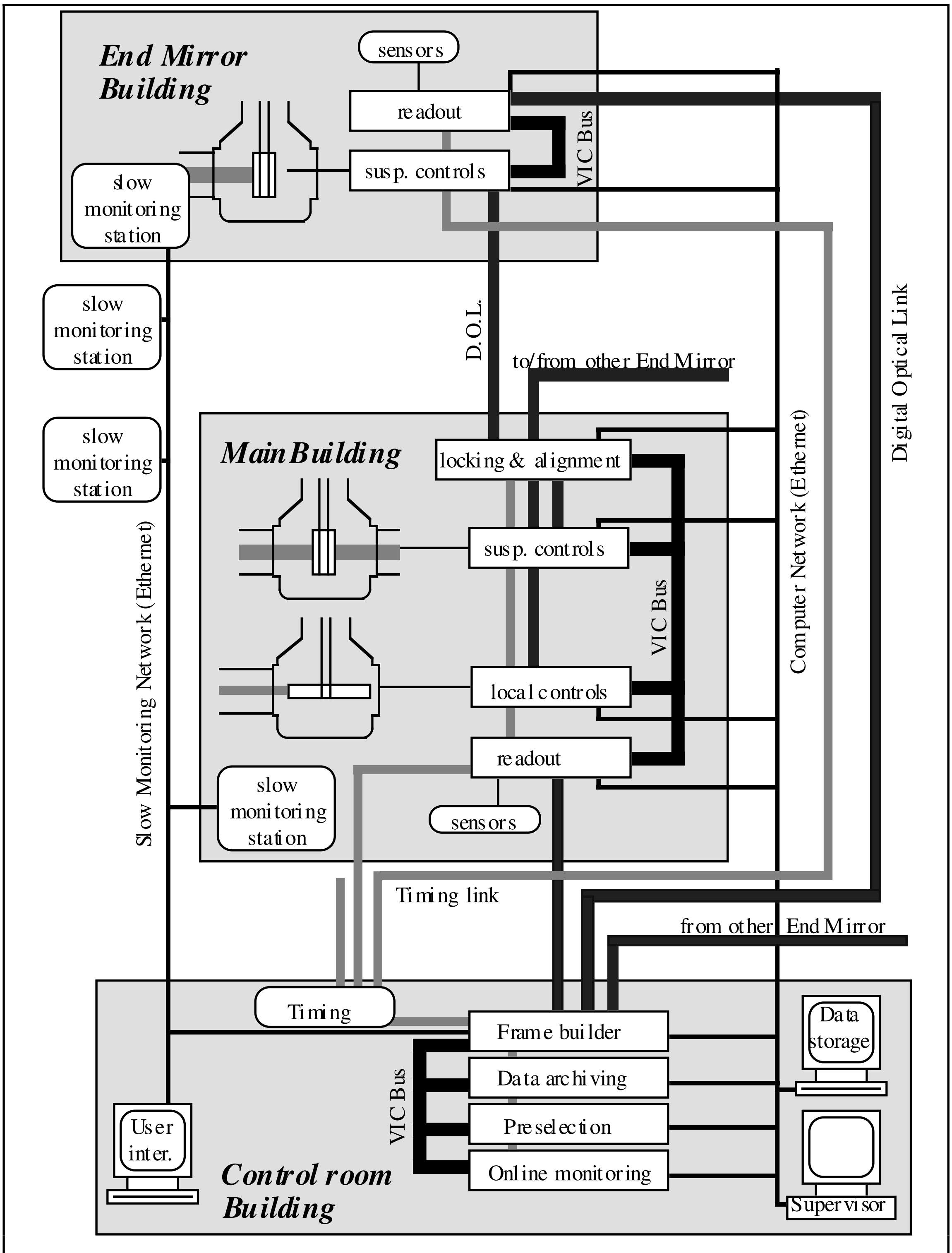

Figure 5000.2 The network architecture implemented between one end mirror building, the main building and the control room.

5000.1.4. The software options

The steady operation of the interferometer is implemented in the context of distributed processing synchronized by a ‘central’ timing system. The various control processes are designed as standalone tasks getting data from a dedicated local captor and/or from another control process. The different readout tasks are organized to operate under the mastership of a central DAQ task.

Each control process (fig. 5000.3) is a server built in the framework of the client-server model. It is a real time program usually running on a VME CPU. It has direct access to the hardware required by its functionality (typically ADC's, servoloop,...). It should be able to run and to deal with its possible error conditions, minimizing its access to the computer network. For fast control proccess, only data for test, configuration parameters and status information should be exchanges using the computer network.

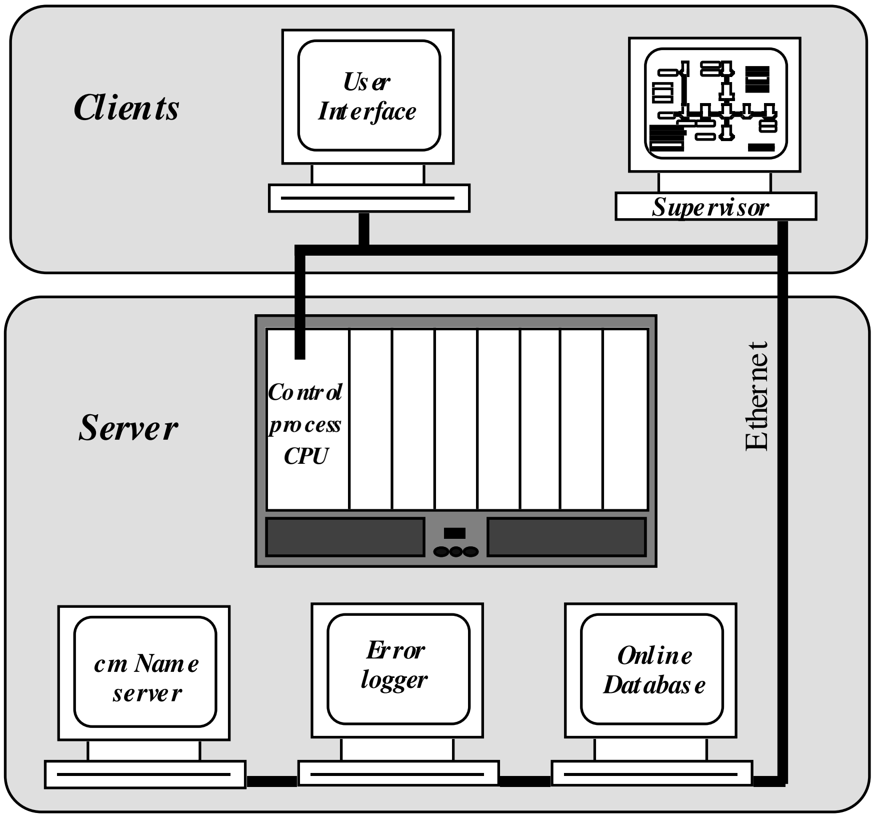

Figure 5000.3 client/server process organization

The user interface (a client of the control process) configure the server and/or monitor and display the data provided by the server. Several user interface clients can be simultaneously used to monitor its informations, but only one client at a time (the master client) has the privilege to configure a server. Possible conflicts are locally solved using standard rules and information provided by the online database. The user’s interfaces are run on workstations and they don't need to be real time.

The cm library is used for ALL the communications between process on the computer network. This package provides platform independent communications tools and access to the process using logical name. The correspondence between logical names and physical addresses is provided by the cm name server. This server is needed for all applications using cm and therefore it should always be running. There is a unique cm name server for the experiment.

To provide an easy control of all the process by a single user interface (The supervisor) all the running servers are able to respond to a standard set of messages. These messages allow the supervisor to control the basic server operations and some monitoring facility. The list of the standard message is given in table 5000.1. Figure 5000.4 and 5000.5 gives additional information about the exchanges messages. In addition to this standard list, messages specific to a type of server are available. They should always start with the two key letters of the process type.

Request		Expected return information	
Message type	Data send	Message type	Data send
SuGetStatus	none	SuServerStatus	status (see note 1)
SuBecomeStatusUser	none	SuServerStatus	status (see note 1)
SuNoMoreStatusInfo	none	SuServerStatus	status (see note 1)
SuLoadConfigFromDisk	file name	SuServerStatus	status (see note 1)
SuLoadConfigFromDB	file name	SuServerStatus	status (see note 1)
SuPutConfig	none	SuServerStatus	status (see note 1)
SuGetConfig	none	SuSendConfig	config file
SuSaveConfigOnDisk	file name	SuSaveConfigStatus	success or failed
SuLoadConfig	none	SuServerStatus	status (see note 1)
SuCheckConfig	none	SuCheckStatus	success or failed
SuRunServer	none	SuServerStatus	status (see note 1)
SuGoToLevel	an integer	SuServerStatus	status (see note 1)
SuStopServer	none	SuServerStatus	status (see note 1)
SuKillServer	none	SuServerStatus	status (see note 1)
SuWhoIsMaster	none	SuMasterIs	master name
SuBecomeMaster	none	SuMasterIs	master name
SuDelegateMaster	client name	SuMasterIs	master name

note 1: The text given the server status should start with one of the following words:
• 8 character for the current status: `-stable-`, `runLvlnn` (where `nn` is a number), `configLd`, `configAv`, `noConfig` (see figure 2.4):
• 6 characters for the time with the format: hhmmss
• 4 characters for the config checksum (computed with a short)
Then additional text could describe the changes like: waiting for DataBase, trying to load,...

Table 5000.1

The stable state is reached when the server and its hardware are operated in a mode which allows stable data taking. Running is for the cases where the system try to reach the stable case like pumping down, trying to align, to lock,...Several running levels could be specified.

All the parameters needed to run a server are stored in configuration files which could be stored locally, by the server, or by the Online Data Base. These configuration files are just an ASCII text to allow easy understanding and debugging. In fact one of the main tasks of the user interface is to be a intelligent configuration editor. The Online Data Base store also all the information about privileges access to the server. This is done by using the machine name and the user account as selection parameters.

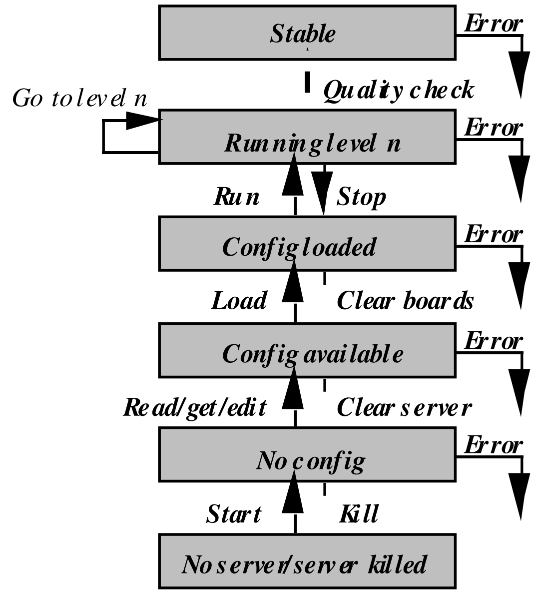

figure 5000.4 Server states. Changes from one state to an other is triggered by one of the messages given in table 2.1 except for the error and for running/stable which is done using some quality check.

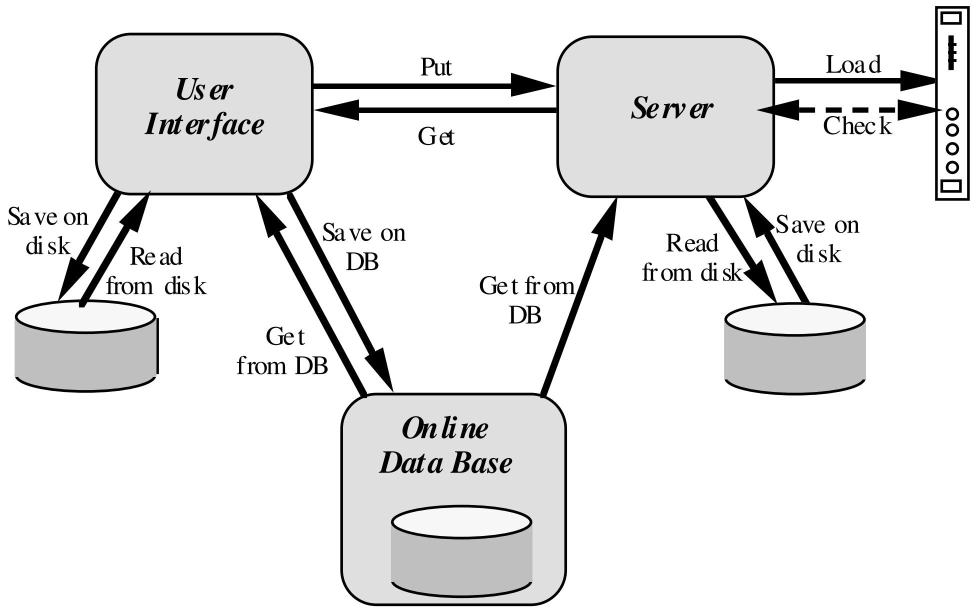

Figure 5000.5 The possible moves for the configuration files

The error logger is the last server which should always be running. It collects the information and error messages from all the process. The error display is the error logger client which provides the tools to select and display error messages collected by the error logger.

5000.1.5. Security

It is important to be able to protect the hardware against any kind of wrong action. We first have to remenber that the hardware is connected to a VME CPU which run the server. The first protection will be to limit the number of account on these processors.

Then the server is driven by a client which run on a different computer. Since all connections use Cm, the next protection will be the limitation (using domaine name for instance) of computers able to be connected to the name server. The Cm connection will also provide the user name for the client. Therfore we can restrict the connections to a limited number of account run on a limited number of workstations. The server will get these informations from the online data base. When two clients want to control the same server, the server will ask the online data base to know which one has the higgest priority. If the connection with the data base is not possible, the first connected client will have the priority.

5000.1.6. The timing system

A central timing system is required to synchronize perfectly the interferometers control, servo loops and readout systems. This is achieved with a master clock driving a bunch of timers which reduce the original frequency to the frequency specific to a given device (see figure 5000.6). Practically, the master clock is derived from a GPS system located in the control room building. Clocking signals are distributed by an optical fiber link to the different buildings where they are converted to TTL signals for local use.

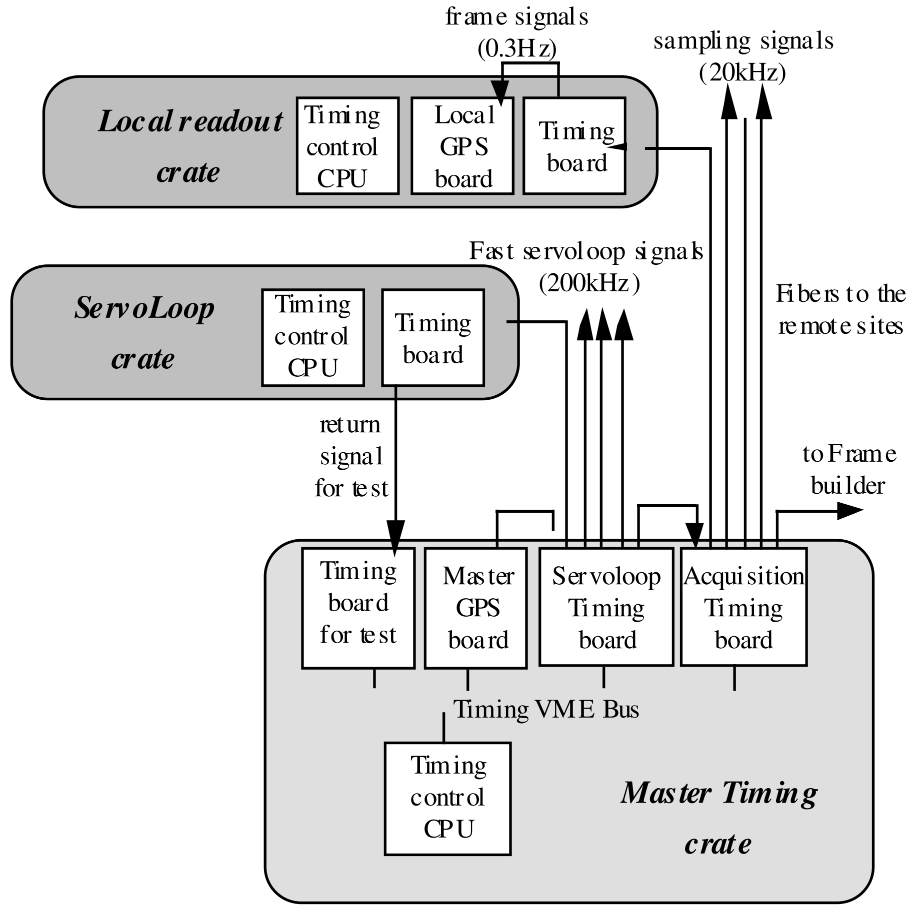

figure 5000.6 The timing system

Whereas the different captors and controls are sampled and operated at rates adapted to their sensitivities, the basic acquisition cycle is set to the sampling frequency of the dark fringe. The timing system produces the necessary interrupts (at 20 khz) to trigger the acquisitions with a 'sampling number' for their later identification. Depending on the required sensitivity, one may be lead to group together several acquisition cycles, covering thus larger time intervals called frames. The timing system provides the frame number for their later identification. The frame starting time is given by the GPS system at reception of this interrupt. It is the time stamp of the data acquisition.

The different controls and feedbacks are operated on fractions or multiples (i.e 200khz) of the basic acquisition cycle. Their action is thus precisely known with respect to the acquisition time. The implementation of this clocking sequence is the corner stone of the safe operation of the detector.

For simplification, the number of clock will be minimized. We forseen three basic clocks:
• The fast servo loop clock (about 200 kHz). It will be use to oversample ADC used by the precise measurement (signal detection or suspension damping)
• The sampling frequency (10 to 20 kHz). It will be use for most of the digital servo loop (locking, suspension control) and for the main data acquisition
• The seismic frequency (100 Hz). It will be use to record monitoring accelerometer.

Each timing board is able to generate a frequency signal by its own in order to guarantee a clock signal to the ADC and DAC boards, even if the connection with the master source is lost. There is a timing server for each crate holding a timing board.

The full timing system could be tested and will be permanently monitored by measuring the sampling time in each local readout crate using independent GPS boards or by using a return signal for fast servo loop. This show the advantage to use this system instead of only a set of independant GPS boards where no cross-checks are possible and where the relative phases between clocks are much difficult to control.

5000.1.7. The Slow monitoring network organization:

The slow monitoring network is designed to control and collect data from Slow Monitoring Station (SMS). A SMS is a standalone system which controls some hardware and which provides a small amount of data with no precise timing information (the foreseen timing accuracy is 1 second). A typical SMS is a Tube control station which controls a pumping station.

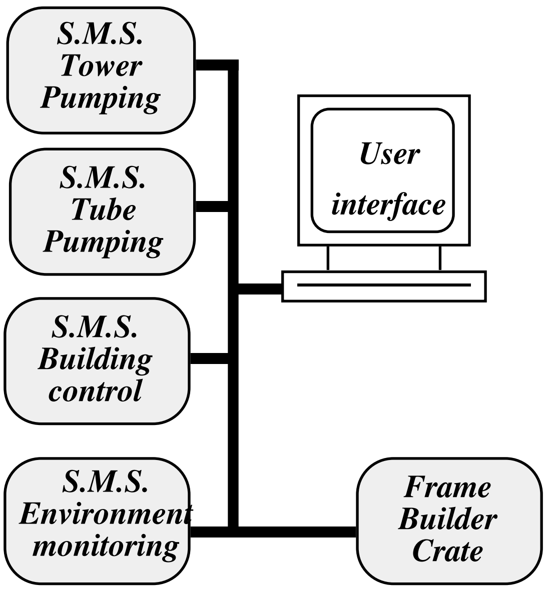

Figure 5000.7 Logical organization of the slow monitoring

A SMS follows the standard software rules. It is a server with user interface run on different machines(see figure 5000.7). The data read by the SMS are formated and send to the frame builder on its request once per frame. The frame builder will send these data to the data distribution and the archiving system even if the fast data acquisition is not running. The history of the parameters can be displayed on a workstation using the Data display and the data distribution systems. For this data collection the SMS has to be able to answer to the following additional cm messages:

- message type: FbGetAllSmsData: the SMS should return all the data to be recorded
- message type: FbGetUpdSmsData: the SMS should return only the data which have changed since the last update request

The SMS answer to these requests is a message with message type: FbSmsData. The message data sent is an ASCII text where all SMS data are described. The format of this text is the following where all the words are separated by a space:

A header:
- SMS station name (4 characters) (see section on standardization)
- the day of the year (ddd)
- time with the format: hhmmss
- 'ALL' or 'UPD' according to the type of data send
- time interval since last update in second
- time interval between the request and the measurement (in second)

Data blocks with for each data:
- the data name
- the data value starting with 1 characters given the data type:
    - i for integer
    - e for exponent
    - f for float
    - s for a string (could be a quoted string)
    - v for a vector. in this case the vector size is given just after v and is followed by the values. Example: v3 i20 i23 i19
    - o for open (for a valve) or for on (a pump for instance)
    - c for close or for off

A end of data trailer:
- the 'EndOfData' string

The figure 5000.8 gives an example of on data block send by a SMS.

```
TOBS 123 084521 UPD 20 1
G31 e5.01e-8
G32 e2.34e-6
SamplInt i200
CPartCh1 i32222
EndOfData
```

figure 5000.8 example of SMS message data. The data comes from the slow monitoring station TWBS (tower control for the beam splitter). The data have been send the day 123 at 8h45m21s. It is a data update over a 20 s. interval.. Four data values are send. Unlike on this figure no newline character should be included in the message.

To preserve the network bandwith, a dedicated network is used for the slow monitoring. The only connected CPU on this network are the SMS, the needed workstations to run the user interfaces and the fame builder.

5000.2. The control of the interferometer.

The various optical and electrical components of the interferometer are driven by local, standalone, controls. Several components can be correlated by a higher level process (locking & alignment) acting through local controls. During setting up, any local control can be operated using the supervisor. The figure 5000.9 is a sketch of this general control architecture which will be detailed in the following sections.

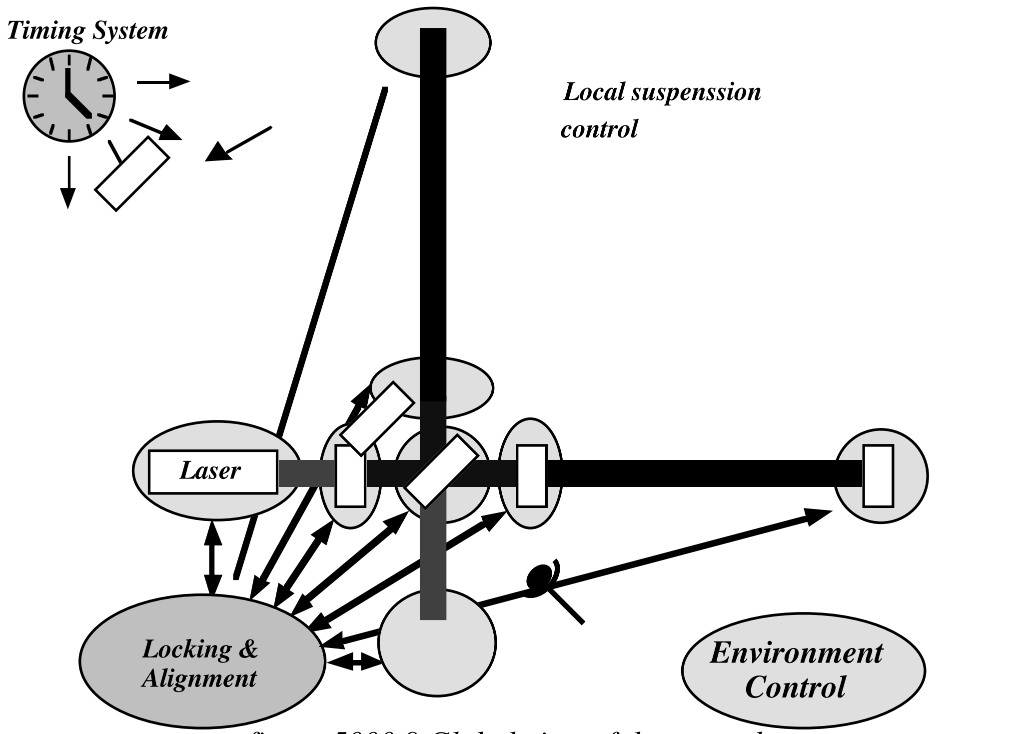

figure 5000.9 Global view of the controls.

Figure 5000.10 presents the chain of hardware component needed for the control implementation. It is a distributed digital servo loop running at a frequency around 10 kHz (this frequency is limited by the transfert time between the component, given their relative distances). The status of each component is control by the supervisor using the computer network, while the real time data are exchanges using hardware connection (DOL). The data exchanges in the links are:
- From signal detection: power seen by the various photodiodes in Watts (DC and demodulated signals)
- From the local position measurement: Three absolute positions and angles
- From the suspension control to the locking & alignement: the mirror absolute position from the local position, the force apply on the mirror and any relevent status information.
- From the locking to the suspension control: the position and angle error.

This scheme is basically the same for the longitudinal locking and for the alignment and strong connection between them will be possible, though we will try to disconnect as much as possible all the functions. The next sections will give more details on the box contents.

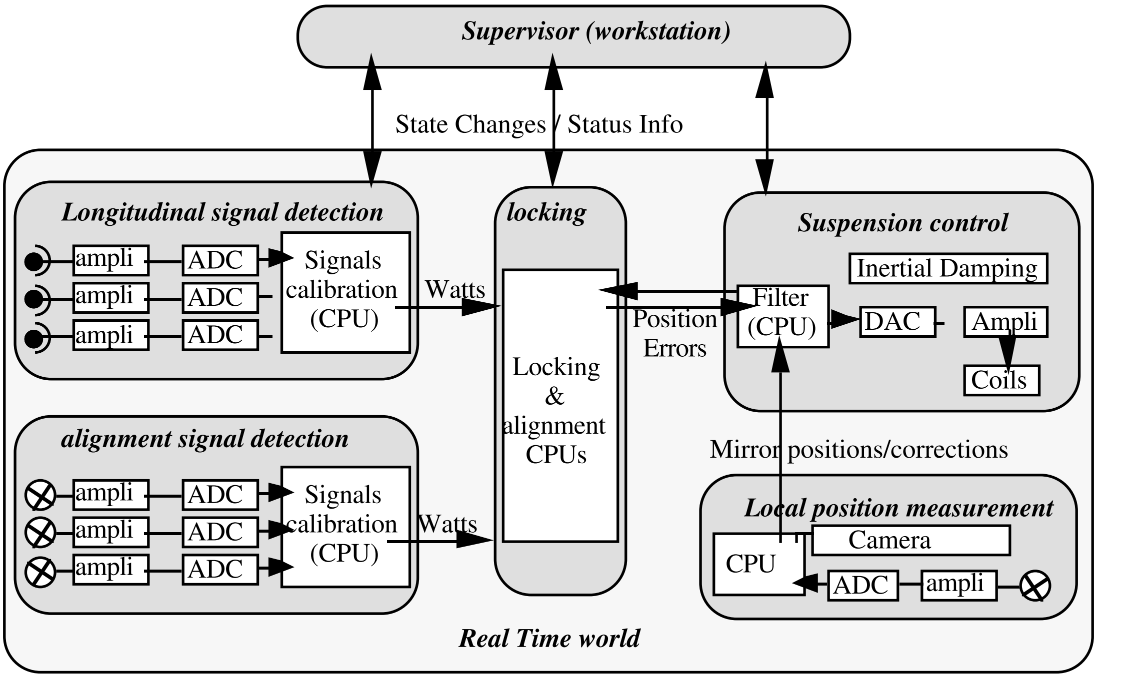

Figure 5000.10 Principle of the control implementation.

5000.2.1. The suspension control

Each optical component hangs on a suspension housed in a tower: the two pairs of Fabry Perot input and end mirrors, the beam splitter, the recycling mirror, the mode cleaner end mirror, the input and detection benches. They are put and kept in position with several mechanisms :
a) coils behind the mirror: for the fine setting of the mirror angles and position along the optical axis
b) marionetta: complements the action of the coils
c) second stage damping
d) upper mechanism: to set the vertical axis of the attenuator and a first approximation of the angle of the mirror with respect to the vertical axis
e) differential vacuum mechanism which separates upper and lower tower vacuum It has to be adjusted according to the mirror final alignment to set the vacuum conductance around the marionetta wire.

The digital hardware necessary for the implementation of such a system is shown on figure 5000.11. Given the large number of boards needed to control a suspension it may be necessary to set up the suspension control in a few VME crates tightly connected and controlled by a single CPU.

Under normal conditions mechanisms a), b) are controlled with feedback loops. The error signals are absolute reference positions measured locally (mainly a DC error signal) or correction signals provided by the locking system or by the linear alignment system trough the Digital Optical Links. A local VSB bus is used as private bus for these servoloops. The absolute reference positions are measured with the imaging system or with quadrant photodiodes or PSD looking at a reference laser beam. They can also be modified by the non-linear alignment system.

The dumping of the second stage is run as an independent 'black box', but its parameters could be controled by the suspension control CPU as well as all the other feedack parameters.

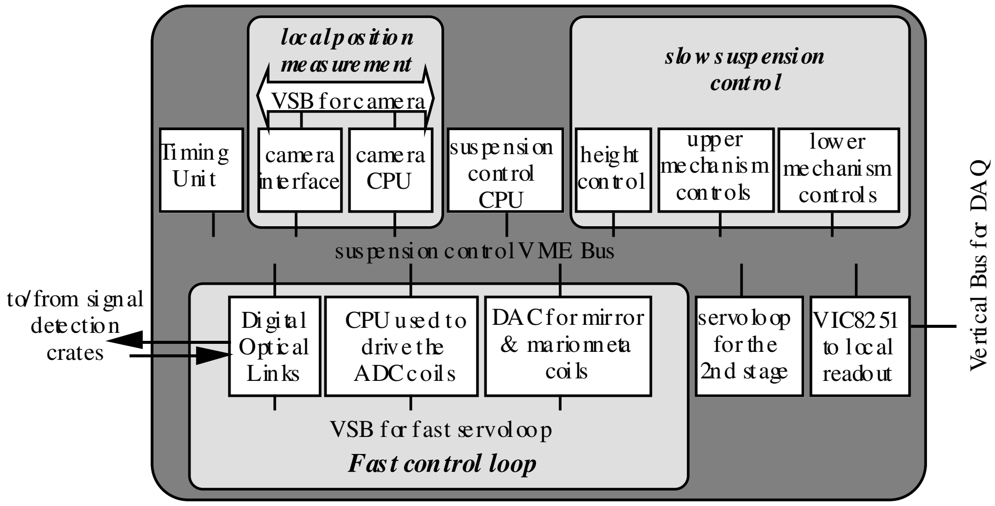

Figure 5000.11 VME crate for a suspension control (crates for the analogic electronic are not shown)

Mechanisms d) and e) are not used in normal operation but they are monitored to check if an action may be required. The absolute mirror position, the suspending wire position relative to the edge of the differential vacuum mechanism are measured with photodiodes, PSD or beam imaging system.

The component control should stay safe, even if error signals are produced by the locking system or if the linear alignment system is misleading. For test purposes the suspension control could be operated as a standalone system using only local data without any global positioning informations.

Each suspension control provides status informations and reference positions to the data acquisition system by putting these data in the mirror memory. It has the capability to send feedback information to the data acquisition when detailed studies are performed.

5000.2.2. Longitudinal signal detection

The signal detection crate is in charge to read the photodiodes and to calibrate the signals. The value computed by the Longitudinal signal detection (DC and AC values) are transmitted to the locking system using a D.O.L. or a VIC board. The figure 5000.12 gives a possible implementation for this crate which in fact has been split in two for noise reasons. This crate is located in the central building because the photodiodes are located in the detection bench or in the input bench. Twin systems are forseen in the end mirror building to measure the transmitted light. Given the expected light power on the dark fringe several photodiodes are foreseen for the main beam. We will have more ADC chanels than the four real signals. The photodiode ADC's are sampled at fast sampling rate (typically 200 kHz) and an inboard CPU reduce this sampling rate to about 20 kHz. Then the photodiode CPU reads the data trough the VSB and a local vertical bus connected to the ADC crate. This CPU sums the channels coming from the same beam. Since the photodiode have tw outputs with two different dynamic corresponding to the two type of signals (one for locking and one for the dark fringe signal), two CPU are used for readout. A low gain channel is also read in parallel to extend the signal dynamic during the start up procedure.

The errors signals and the status of the servo-loops are also written in the vertical bus memory to be recorded by the data acquisition.

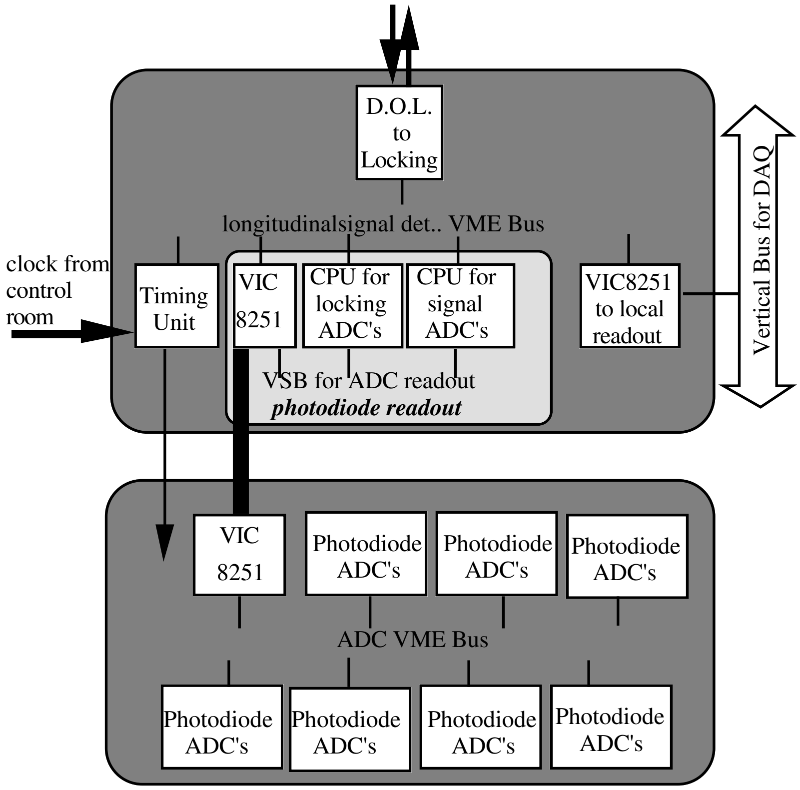

figure 5000.12The signal detection crates

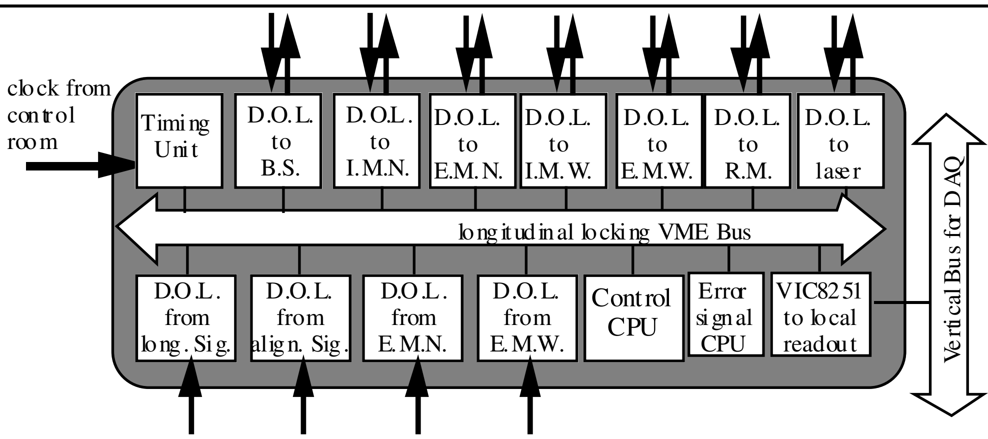

figure 5000.13 The Locking&alignment crates

5000.2.3. Global Control

This first component of the Global Control is located in the Locking and Alignment crate, in the main building: It is completed by the Global Control Survey (cf. online processing) in the Control Room building

The real time Global Control is in charge of the longitudinal locking and alignment of the interferometer. It is activated by the Supervisor once the interferometer has been brought close enough to the linear regime for the dedicated softwares to be able to lock it on the dark fringe. These Global Control softwares are flexible enough to accomodate different kinds of locking schemes, to handle longitudinal as well as alignment drifts, and to be able to guarantee smooth operation of the interferometer both in its linear and non-linear regimes. A hierarchy of several layers of cpu's is foreseen. This hierarchy is based on the time response of the various cpu's; each using its own (more or less sophisticated and, hence, time consuming) model of the interferometer. During smooth operations, only the simplest and fastest of these models is to be used while the other cpu layers are working only in spy mode to check the validity of the responses of the fast model. When necessary, if the quality of models is no longer sufficient, a higher hierarchy layer can update the parameters of lower level models or, if needed, take care directly of the locking and alignment tasks.

This Global Control (fig. 5000.13) uses the error signal provided by the various signal detection systems to compute the time dependent parameters needed to describe the status of the interferometer. It is also kept informed of the relevant actions decided locally, for example, by the servo-loops in charge of the suspension chains. Its principal task is to compute the positions and the corrections to be applied on the current positions of the main optical elements. These correction requests are sent to the corresponding local controls which are in charge of implementing them. The corrections, together with the error signals and the status of Global Control and local servo-loops are sent to the DAQ, for monitoring and for the Global Control Survey.

5000.2.4 Non-linear Global Control

At setting up, an initialization operation is performed to bring the different components of the interferometer in positions close enough to the ideal ones to allow the software of Global Control to be activated. When this activation has been successfully performed - i.e when the interferometer is in its normal running mode - the non-linear alignment process is operated in spy mode, to check and update the various reference positions.

The non-linear alignement task is a high level task closely related to the linear aligment one. It also acts on the various component positions by sending requests to the local servo-loops. This task is achieved mostly by manual operations through a dedicated user interface, using general resources, among which the beam-imaging system.

5000.2.5. Laser control

It controls the diodes power supply, the laser power and some prestabilisations.The diodes power supply control will be in VME crate connected to the slow monitoring network. The functions required by this control are:
-turn the laser on or off (diodes supply on or off).
-monitor parameters such as diodes state, diodes and Yag temperature, global current.
-prestabilise the laser power acting on the global current in the diodes.
-keep the Yag in monomode operation.
-maintain a fixed temperature in the reference FP cavity.

The laser power and frequency prestabilisations are done by analog servo loops which are autonomous, but which can also exchange informations via the VME system with a high level task.
The laser power and the phase noise of the laser relative to the 3km cavity are sampled in ADC's located in VME crate and sent to the data acquisition at the same rate as the dark fringe signal.

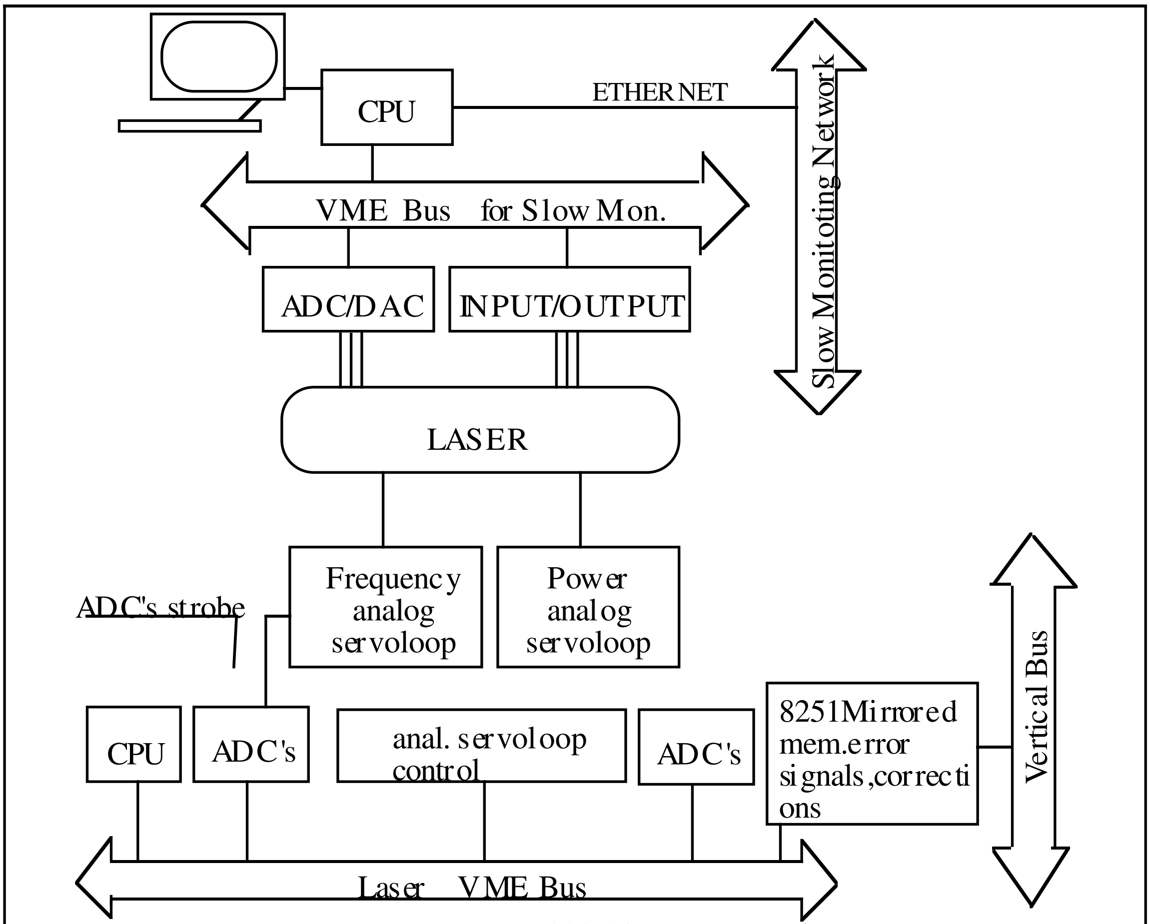

Figure 5000.14 Laser control

5000.2.6. Input bench control

It controls the alignment of the laser in the input bench, the input bench and mode cleaner electronics. It keeps the mode cleaner locked by running servoloop exchanging data with the end mirror mode cleaner control. Monitoring signals are sent to the main building readout through the vertical bus.

5000.2.7. Detection bench control

It controls the electronics and the hardware located on the detection bench. It is connected to the signal detection crate. It runs the local bench alignment and provided status information using the slow monitoring network.

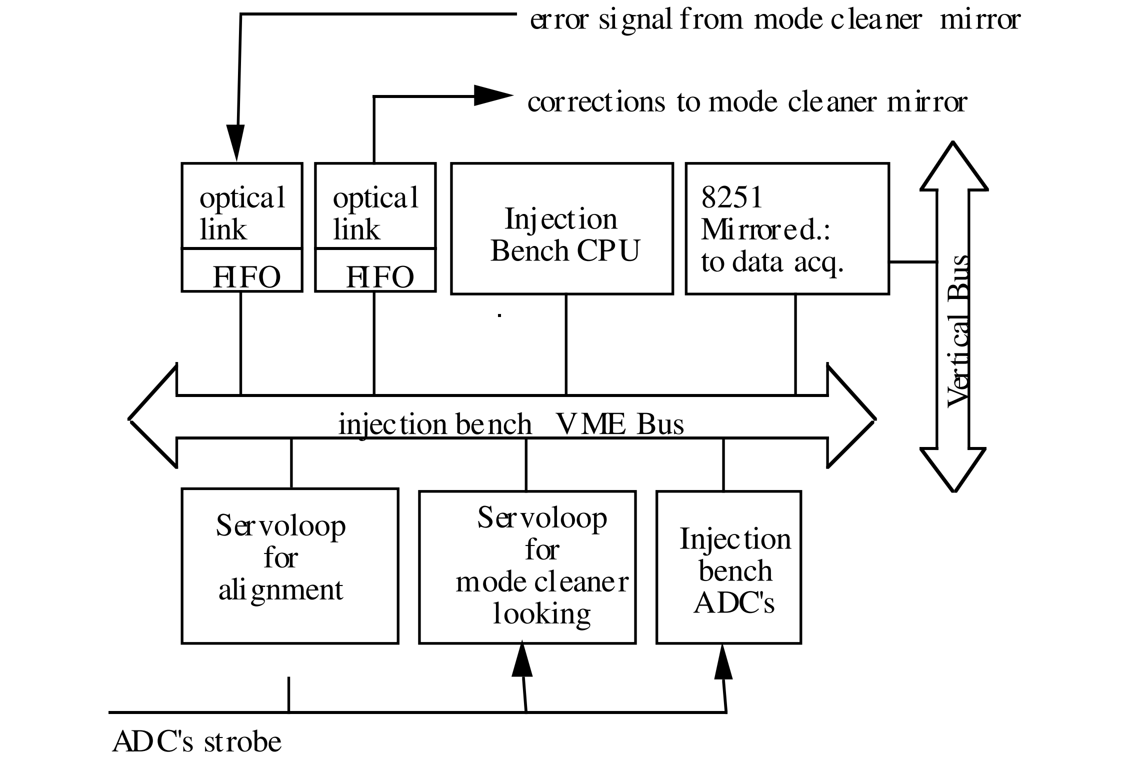

Figure 5000.15 Input Bench control

5000.3. The control and monitoring of the interferometer environment

These control is done using Slow Monitoring Station. The Building control station and the Environment Monitoring Station are described in section 5500.

5000.3.1. The Tube Control Station

It controls all the vacuum system for one pumping station. This includes the baking control, and therefore the temperature control and the power needed to heat the tube. A diffuse light monitoring could also be connected to the tube control station.

A G64 (or a VME) crate is foreseen for the each tube control station.The electronic is controlled by the local CPU. the slow monitoring network. It provides a timed set of vacuum measurements performed at regular time intervals. Summary informations are send on request to a global vacuum control and monitoring user interface and to the frame builder. A dedicated workstation for the vacuum control and monitoring is foreseen in the control room. Furthermore, mobile terminals which can be linked to any crate will allow to work close to the vacuum components (particularly in the pumping station).

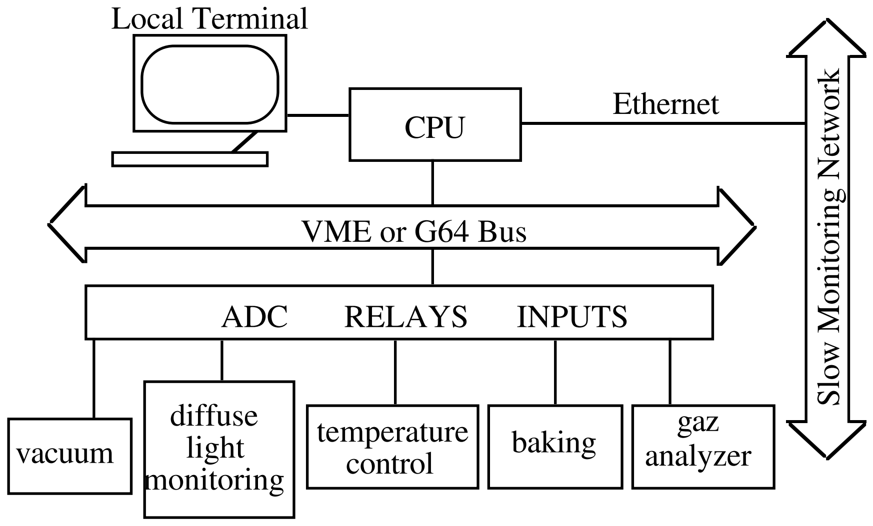

Figure 5000.16 A tube control station

5000.3.2. The Tower Control Station

Like the tube control station it drives all the pumping system and also the tower clean room installation. It controls the backing of the tower, the tower temperature and the gaz analyzer (see note from P. Roudier from 27/6/94 for more details about the components). It is connected to the slow monitoring network and provides a timed set of measurements performed at regular time intervals.

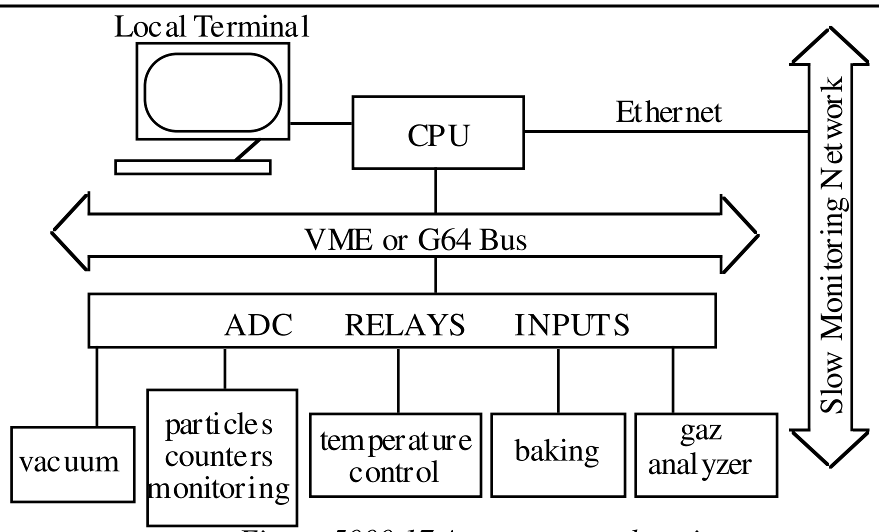

Figure 5000.17 A tower control station

5000.3.3. Imaging system

This is the front end for all the beam imaging systems (camera) which measure beam intensity or wavefront. It starts the various processes which control the cameras and provide images for all the interferometer. Graphical user's interface can be run on any workstation to display the beams. It provides information to the alignment system and sends beam profile to the data acquisition.

5000.3.4. Calibrators control

This controls the calibrators initialization and running (frequency signal, wavefront in the case of the laser calibration). It is a slow monitoring station. The current calibrator status is recorded and the generated signal is send to the local readout process.

5000.4. Offline

The major offline activity is the data analysis aiming at the extraction of a gravity wave signal. This activity where imagination is very important should stay widely open within the collaboration. But one needs a set of tools and rules which should be managed in common to provide basic software environment and to allow for comparisons of different works.

There will be a simple analysis framework to provide the basic functionality of analysis jobs like read/write frames, simple monitoring, run the online selection. Such programs could already be tested with SIESTA.

During data analysis we will need some generic tools, usually in the area of histogramming, plotting and fitting. For this need, we will use as much as possible the tools developed at CERN. This is especially true for the histogramming with HBOOK and interactive plotting with PAW .

The offline reconstruction program which converts the raw data to an h value has also to be standard and identical to the online reconstruction run for the frame selection. Improvements in the knowledge of the transfer functions or of the calibration constants may justify an offline re-run of the h reconstruction.

5000.5. Detector Commissioning and Calibration

It is mandatory to allow for a progressive and independent commissioning of the various components of the experiment. To that purpose, the readout has been organized in autonomous subsystems capable of local standard data acquisition. To exercise locally and independently the various controls, one has to provide the necessary excitations and measurements devices. Once everything is set up, i.e. once the interferometer is running its beams, then a calibration device is foreseen to monitor permanently its sensitivity.

5000.5.1. Data acquisition commissioning

The DAQ of the experiment is performed by a central system which assembles the frames whose details are collected trough optical links fed by the local readouts. Once the local readouts have been set up, one certifies their frame building capability with appropriate test procedures which do not need any special hardware. As an example, the local readouts can be programmed to deliver on output simulated data. This would allow for a test of the optical links and of all the procedures following the frame assembly.

5000.5.2. Controls commissioning

The suspension control is organized in independent system with local position measurement. This gives the capability to test all the actuators and the controls. For the global control, simulated data can be generated and sent in the data links. But we will have also the capability to illuminat the photodiodes with known signal and to test the effectiveness of the global controls.

5000.5.3. Calibration

A direct calibration of the interferometer is obtained by stimulating it as a gravitational wave would do it. This can be achieved by moving an end mirror in a known way , and recording the result in the standard acquisition stream.

This calibration has at least two different goals. The first one is a general study of the Virgo sensitivity and a help for the commissioning of the full antenna (global controls, reconstruction software,...). The other is to provide a continuous monitoring of the interferometer in order to study long term effects. For that purpose, one uses the periodic signal of well known amplitude and very stable frequency produced by a dedicated device. The clock used to stabilize the input signal is different from the one used by the data acquisition system in order to allow cross checks.

From the data management point of view, there will be a calibration control to operate the calibrators (set frequency and intensities). The status of the calibrators will be recorded by the VME data acquisition system. When the calibrator injects a signal with variable frequency, this signal has to be recorded with the standard data.

5100. Global controls (associated software)

5100.1. The Supervisor

The function of the Supervisor is to keep track of, to request, allow or inhibit modifications of the status of the various components of the interferometer. With respect to this functionality directed towards Organisation and Security, it is intended to provide the overall user interface (the master client) of VIRGO. The Supervisor, in particular, provides a graphic display of the status of all the VIRGO components. It will also use the output of the Data Monitoring programs and should later house the expert system in charge of the interferometer. A set of automatic sequences will be available for the interferometer start up for instance.

The Supervisor controls all the servers using the messages described in table 5000.1. It displays the status of all the available servers. When a special action is required for a given server it can start the corresponding client and delegate its privileges.

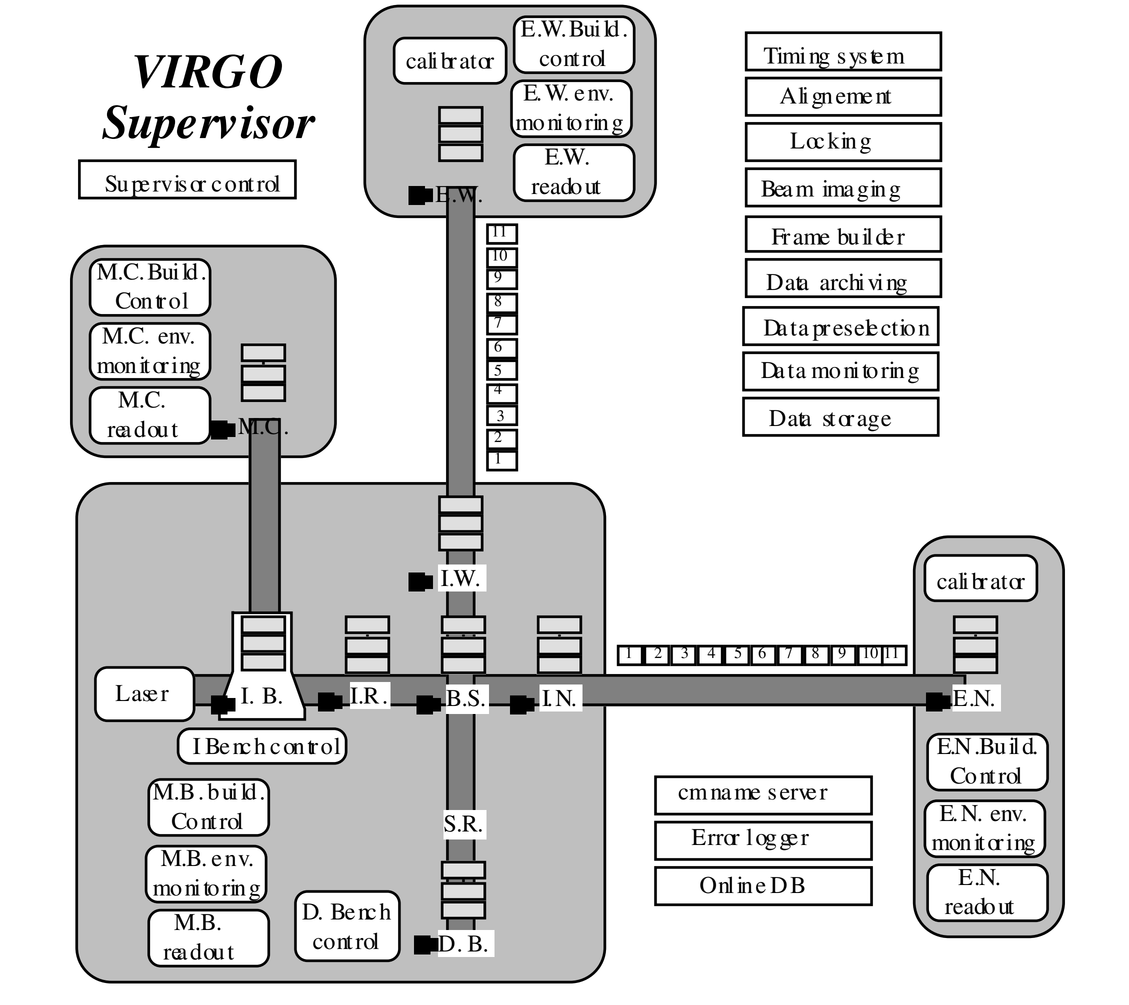

Figure 5100.1 The Supervisor

The list of servers controlled by the supervisor is called the partition. This partition which can be modified, is described in the supervisor configuration. The supervisor is in fact the 'front page' of Virgo as it provides a very short description of the status of all available tasks using a color code. The possible status are:

- Not in partition	grey
- In partition but not connected	black
- Connected and no configuration file	red
- Connected and config available	orange
- Connected and config loaded	yellow
- Connected and running	blue
- Connected and stable	green

In addition, the type of client can be specified (master which is the default, or ordinary client) and a message displayed. A changing status is indicated by a flashing color. The figure 5100.1 shows a possible supervisor.

By clicking within the Supervisor panel on a server icon, one may perform the following actions on a server with the help of a pop up menu:
- add or remove from the partition
- configure (set parameters like server name,...)
- start server (task)
- ask for connection
- read status
- read and load server configuration file
- run and stop the action performed by the server
- start the corresponding user interface.
- kill the server
- become master or slave user

Under smooth conditions there is only one Supervisor running in the control room with all the privileges for the hardware access. It collects the status information from the servers and checks the error logger for warning or error messages. It asks periodically for status. A non answering server could be considered as an error message. Additional supervisors without privileges for the hardware access could be run in different places to monitor the interferometer.

5100.2. The communication tool: Cm

(To be completed)

5100.3. The ErrorLogger

The error logger collects the information and error messages from all the process. The messages are send using the El library which use Cm.
Each message contain a severity code and a text. The levels of severity are:

- info — just information message
- warning — their is some problem but the standard operation can continue
- severe — some non standard procedure has to be follow to fix the problem. The server goes one state down for severe error.
- fatal — the server is stop or going to kill. It goes more than one state down.

The error display is the error logger client which provides the tools to select and display error messages collected by the error logger.

(To be completed)

5100.4. The Online DataBase

(To be completed)

5200 Standardized components

5200.1. Standardization and Software rules

5200.1.1. Language

- Lynx OS for real time VME CPU (or OS9 for G64 crates)
- UNIX system for workstation
- Motif and X window for graphic
- C with C++ style for language

The software run on various processors will be written in C language as much as possible to be able to use it in the simulation program and analysis programs if needed. This is also true for monitoring task run on the front end workstations which are in charge of the control and monitoring of the real time processor.

5200.1.2. Coding rules

Here is a short list of the main suggest coding rules

• Use `double` instead of `float` (for 'scientific' calculations),
• Use `long` instead of `int` (to make sure we have 32 bits).
• Use a naming convention:
  - All function names, structure types and variables which are not local should start with the two key letters of the library. Example: `FrRawData`
  - Keep UPPER case for `define`.
  - Start local variables with lower case.
  - Start global variables with upper case.
  - To build composed words use upper case instead of underscore.
• Write `simple' C code
• Only one instruction per line (except for `for`, `while`)
• Use indentation for loops, if... (at least 2 spaces)
• No C++ keyword to allow a futur migration to C++. As a reminder here is the list of C++ keywords: `asm`, `catch`, `class`, `delete`, `friend`, `inline`, `new`, `operator`, `private`, `protected`, `public`, `template`, `try`, `this`, `throw`, `virtual`.
• Use comments, especially in the header file to describe each variable defined in a structure.

5200.1.3. File organization

The files organization is presented in figure 5200.1. The reference version for the Virgo software is stored in the machine LAPPSUN18.IN2P3.FR for the time being. Accounts on this machine are available on simple request. The virgoApp directory could be found in /virgoApp. The notes directory is a public space were general information could be install. All packages are stored in separated directory. A readme file should give some basic information like the person in charge of this package and who to contact in case of problem. The table 5200.1 gives the list of the current packages and their key words and manager. Each package as a single person in charge of the updates. The standard version of the software is found by using the 'pro' directory which contain logical links on the various packages. When several people are working on the same package, CVS is used as code manager.

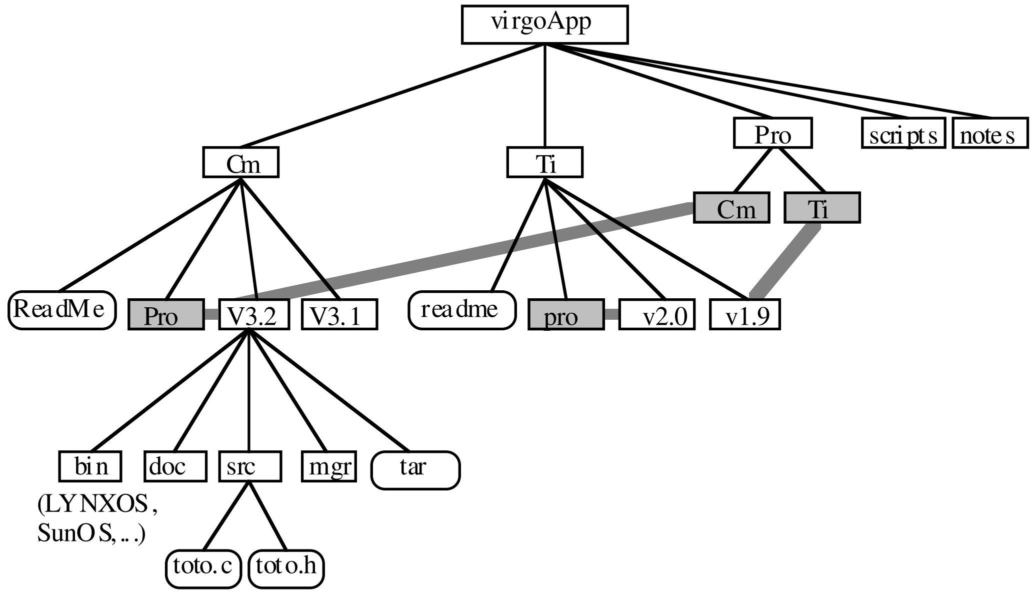

figure 5200.1 The file organization

Package	Key letter	Manager
Communication	Cm,Ci,Cset	C. Arnault
Digital servoloop	Ds	F. Barone
Online database	Db	C. Arnault
Data Display	Dy	X. Grave
Error logger	El	C. Arnault
Frame Builder	Fb	E. Delorme
Frame Lib	Fr	B. Mours (+UJ,UM,UH,UA,US)
Imaging system	GX	C. Drezen
Timing system	Ti	F. Bellachia
Tube control	Tu	
Tower control	To	Orsay
Environment monitoring	Em	F. Barone
Building control	Bc	F. Barone
Laser control	La	
Input bench-control	Ib	
Signal detection	Sd	
Detection bench-control	Db	
Non-linear alignment	Nl	
Linear alignment	Li	
Supervisor	Su	
Local readout	Lr	F. Bellachia
Reconstruction	Re	
Online trigger	Tr	
Data archiving and distribution	Dd	F. Barone
Historical monitoring	Hm	F. Barone
Calibration	Ca	
Simulation	Siesta (+ME,GR,MI,OP,FR,EL,FO)	F. Marion

Table 5200.1 Packages list

5200.1.4. The list of Slow Monitoring Stations

Here is the list of the foreseen slow monitoring station with there name:

Tube control station:	TW01,TW02,...TW11,TN01,...TN11 (for west & north arm)
Tower control:	TOIB (input bench), TOMC (mode cleaner), TOPR (power recycling), TOBS (beam splitter), TOIN (input mirror north), TOEN (end mirror north) TOIW (input mirror west), TOEW (end mirror west) TOSR (signal recycling), TODB (detection bench)
Building control:	BCCE (for central building), BCMC(for mode cleaner) BCEW (for end west), BCEN(for end north)
Environment monit:	EMCE (for central building), EMMC(for mode cleaner) EMEW (for end west), EMEN(for end north)
Laser control:	LASR
Benchs:	IBEN (input), DBEN (detection), NBEN (north arm bench), WBEN (west arm bench)
Calibrator:	CALM

5200.1.5. Data format on tape:

Frame number and sampling number are limited to 65536. Frames collected with the same experimental condition are grouped in 'Run' identify with a run number.

A data file should start with a file header with the following data:
- 1 byte giving the format version (current version number is 0)
- 1 byte = sizeof(short)
- 1 byte = sizeof(int)
- 1 byte = sizeof(long)
- 1 byte = sizeof(float)
- 1 byte = sizeof(double)
- a 4 bytes word containing 0x01020304 (to check the bytes swaping)
- a float (4 bytes) containing PI
- a double(8 bytes) containing PI

The file is structured in records. One record contains one C struct. For each record there is a short header of three long words:
- the first is the number of bytes in the record including the header
- the second is the type the C struct. This number is dynamically generated by the write program. The correspondence with the structure name is given by the dictionnary which is included on tape.
- the third is the instance number (a unique number per structure of one type). This number is dynamically generated by the write program.

The content of the C struct follows this header. Data are written in binary IEEE format (68040 format) in the same order as in the structure. Character strings are preceded by their length coded as a long.

The file starts by the detector description (if any) which written in several records. Then follow the frames which are also written in several records. A Frame starts with the FRameH record and ends with the EndOfFrame record.

When frames are exchanges between client and server, we just send on binary file per frame with the same format as the one used to write on tape.

More informations and examples could be found in the Siesta directory in the file UJobL.h

5200.1.6. Information distribution

News could be broadcast to a list of Virgo people by send a mail to VIRGO-L@FRCPN11.IN2P3.FR. VIRGOS-L is for software people and VIRGO-F for Virgo 'Friends', To be include in this list, please send a mail to MOURS@FRCPN11. A WWW server is available (....). Documentation could be found also with the software storage.

5200.2. Summary of crate content

5200.2.1. Control Building:

C1: Timing
 1 CPU (control), 1 GPS, 2 Timing (out), ≈ 5 Timing (in)
C2: Frame Builder:
 3 CPU, 4 DOL, 1 VIC (to data distribution)
C3,4: Raw data archiving
 1 VIC, n CPU
C5: Data quality
 1 VIC, n CPU
C6: Trigger
 1 VIC, n CPU
C7: Online monitoring
 1 VIC, n CPU

5200.2.2. Main building:

C21: Laser control
 1 CPU(control), 1 VIC (to DAQ), 1 Timing, ≈ 5 ADC/DAC (slow)
 1 DOL -> input bench, 1 ADC (fast)
C22: Input bench control
 1 CPU(control), 1 VIC (to DAQ), 1 Timing, ≈ 5 ADC/DAC (slow)
 1DOL (to laser), 1 DOL (to mode cleaner), 1 DOL (to locking), 1 ADC (fast)
 [3 Camera, 1 CPU](VSB)
C23: Detection bench control
 1 CPU (control), 1 VIC (to DAQ), 1 Timing, ≈ 5 ADC/DAC (slow)
 1DOL (to laser), 1 DOL (to mode cleaner), 1 DOL (to locking), 1 ADC (fast),
 [1 CPU (signal processing), 1 VIC (to analog crate)](VSB),
 [3 Camera, 1 CPU](VSB)
C24: Signal detection
 1 VIC (to det. bench control), ≈ 8 ADC (fast) (about 32 channels)
C25: Signal detection
 Analogic electronic
C26: Signal detection
 Analogic electronic
C27: Alignment signals:
 1 CPU (control &signal processing), 1 VIC (to DAQ), 1 Timing,
 1DOL (to locking), ≈ 4 ADC (fast) (about 16 channels), ≈ 2 ADC/DAC (slow)
C28: Alignment:
 Analogic electronic
C29: Locking
 3 CPU (control &signal processing), 1 VIC (to DAQ), 1 Timing, 11 DOL
C30: Calibration
 1 CPU (control &signal processing), 1 GPS, 2 ADC/DAC (fast)
C31: Environement & building control
 1 CPU (control), dedicated network hardware
C32: Local readout
 1 CPU (control &signal processing), 1 VIC (to collect DAQ), 1 Timing, 1 GPS,

1 DOL (to frame builder), ≈ 5 ADC (fast)
C33,35,37,39,41,43: Suspension control (21 b.)
1 CPU (control), 1 Timing, ≈ 5 ADC/DAC (slow)
[1 CPU (signal processing), 1DOL (to locking), 1 DAC (fast),
1 VIC (to DAQ),](VSB),
[2 Cameras, 1 CPU](VSB, local position measurement)
[1 DSP , 2 ADC, 1 DAC] (2nd stage dumping)
C44,46,48,40,42,44: Suspension control
Analogic electronic

5200.2.3. Mode cleaner:

C61: Local readout
1 CPU (control &signal processing), 1 VIC (to collect DAQ), 1 Timing, 1 GPS,
1 DOL (to frame builder), ≈ 2 ADC (fast)
1 CPU (control), 1 Timing, ≈ 5 ADC/DAC (slow)
C62: Suspension control (21 b.)
[1 CPU (signal processing), 1DOL (to input bench), 1 DAC (fast),
1 VIC (to DAQ),](VSB),
[2 Cameras, 1 CPU](VSB, local position measurement)
[1 DSP , 2 ADC, 1 DAC] (2nd stage dumping)
C63: Suspension control
Analogic electronic

5200.2.4. End building:

C71: End mirror bench control (long. signals & alignment)
1 CPU (control & signal processing), 1 VIC (to DAQ), 1 Timing,
1DOL (to locking), ≈ 3 ADC (fast) (about 12 channels)
≈ 2 ADC/DAC (slow)
[1 Camera, 1 CPU](VSB)
C72: Signal detection
Analogic electronic
C73: Calibration
1 CPU (control &signal processing), 1 GPS, 2 ADC/DAC (fast)
C74: Environement & building control
1 CPU (control), dedicated network hardware
C75: Local readout
1 CPU (control &signal processing), 1 VIC (to collect DAQ), 1 Timing, 1 GPS,
1 DOL (to frame builder), ≈ 2 ADC (fast)
C76: Suspension control (21 b.)
1 CPU (control), 1 Timing, ≈ 5 ADC/DAC (slow)
[1 CPU (signal processing), 1DOL (to locking), 1 DAC (fast),
1 VIC (to DAQ),](VSB),
[2 Cameras, 1 CPU](VSB, local position measurement)
[1 DSP , 2 ADC, 1 DAC] (2nd stage dumping)
C77: Suspension control
Analogic electronic

(crate number are given for the north arm. Add 10 to get crate numbers for the west arm)

5200.3. Workstations Implementation

5200.3.1. Control Room & Computer Room

The Control room will contain a set two clusters of identical workstations. Even though they will be able to run any user interface they will be dedicated to one or two specific type of control. Two clusters will be use instead of one in order to do system maintenance without stoping Virgo. Here is the list of the forseen workstations:

- W1: Tube control & Tower control (on slow mon. net.)
- W2: Building control & Environnement control (on slow mon. net.)
- W3: Laser & input bench control
- W4: Detection bench control & Signal detection
- W5: Suspenssion control
- W6: Locking & Linear alignement
- W7: Name Server, Supervisor & Timing control
- W8: Non Linear Alignement & Beam Imaging
- W9: Local Readout & Frame Builder
- W10: Data distribution control & Data archiving
- W11: Trigger control & Data Monitoring

• W12: Data distribution: This workstation is located inside the computer room. It is a more powerfull workstation with lot of disk space and special hardware.

• W13: Network analyseur

The rack content is the following:

- R1 Cabling arrival
- R2 Network Hardware
- R3 Master timing (1 c.) Frame Buidler(1.c) Reconstruction & Trigger (1c.) Data monitoring (1 c.)
- R4 Data archiving (1crate + space for tape drives and disks)
- R5 Free (could hold CPU monitors)

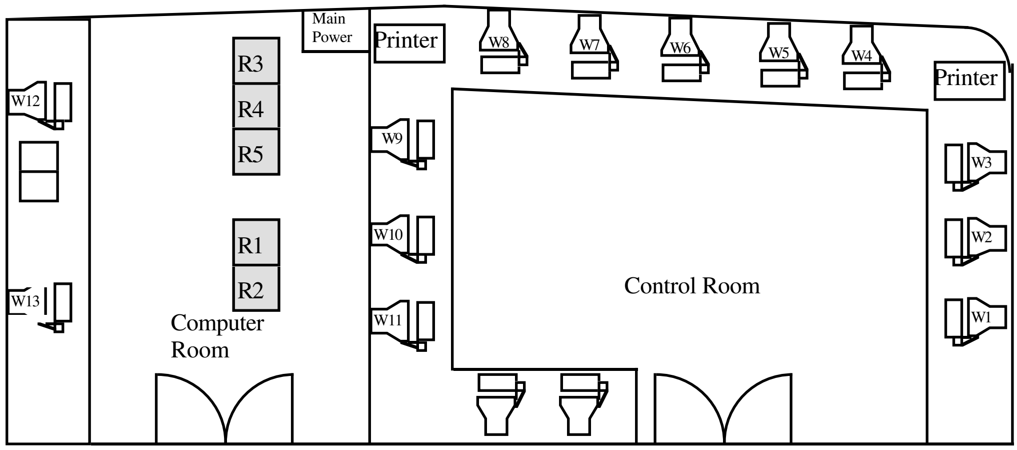

Figure 5200.2 Control Room and Computer room. Video monitor (for beam imaging or for room survey) will be install above the workstation

5200.3.2. Main Building

Few workstations (or X terminal) will be available in the Data acquisition room to locally work on the control. They will be on the same cluster as those in the Control Room.Here is the list:

- W21: Tower control & Tube control (on slow mon. net.)
- W22: (for W3,W6)
- W23: (for W4)
- W24: (for W6)
- W25: (for W8)

The rack content is the following:

- R11 Cabling arrival
- R12 Network Hardware
- R13 Detection Bench control (1.c) Signal detection (2 c.)
- R14 Detection Bench Suspension control (2 crates)
- R15 local readout(1c.), Calibration (1 c.), Env. monit & Build control (1c.)
- R16 Input North and Input West Suspension control (2x2 crates)
(These crates may need to be closer to these towers)
- R17 Recycling and Beam Split. Suspension control (2x2 crates)
- R18 Locking&alignment(1c.), alignement signals(1c.),
- R19 Detection Bench (2 crates)
- R20 Laser control (1 c.), Input bench control (1c.)

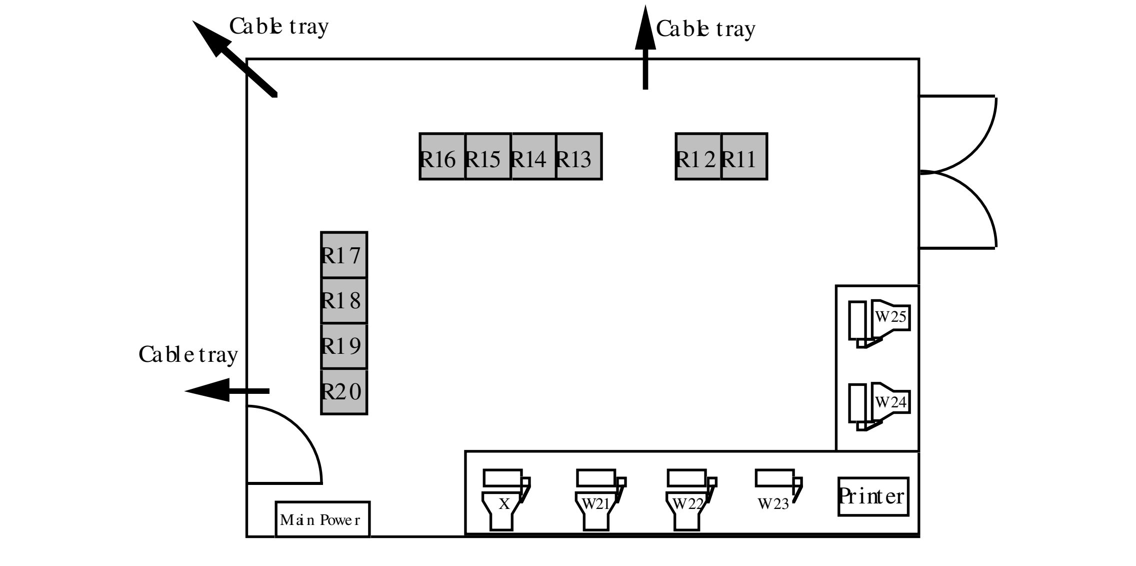

Figure 5200.3 Control Room and Computer room. Video monitor (for beam imaging or for room survey) will be install above the workstation

5200.3.3. Mode Cleaner or End Mirror Building

One general purpose workstation (W31, 32, 33) and one X Termnial will be available next to the tower. There will be also one rack with 2 crates for the suspension control and one for the local readout.

5200.4. Standardization: Hardware

For the hardware, we use as standard:

- the VME bus with A24D32 addressing mode for fast system
  or G64 for slow monitoring system like vacuum control
- an Ethernet network for the slow monitoring
- the VIC vertical bus between VME crate with the VIC8251 interface
- the same general purpose VME CPU
- Ethernet or FDDI for the general computer network.
- The Digital Optical link for fast crate to crate connections. These links are seen has a FIFO on each crate. The data rate on the 3 km optical fiber is 14 MBytes/s.

We will use the same type of VME crate and general purpose board (ADC,DAC,..)

BNC connector for cables.

The estimation of the overall number of standardized components is the following (including spears):

20 Worksations
40 VME crates
60 VME CPUs
40 DOL boards
.. fast ADC (> 20 kHz) + .. slow ADC (< 1 kHz/channel)
.. fast DAC (> 20 kHz) + .. slow DAC (< 1 kHz/channel)
30 Timing boards
8 GPS boards
32 Pumping stations

5300. Networks

The cabling necessary for the networks of VIRGO can be divided as follows:

• Optical Links for high speed transfers**:** This applies to the distribution of the time signal, to the servoloops control and to the acquisition.
• Slow Monitoring Network: This applies to the monitoring of all parameters collected from many crates in all buildings, towers, pumping stations, etc...
• Computer Network**:** This applies to the connection of all computers, workstations and some of the VME crates.
• Telephone: This applies to ordinary telephone lines in all rooms of all buildings.
• Video: This applies to video monitoring of remote places and of surroundings.

The cabling we propose conforms to the standard EIA/TIA 568 for cabling, and is generally referred to as "structured cabling". This means a hierarchical organization from a central point to peripheral points, with radial configuration of cables departing from each system and with a maximum length of 100 meters for each cable, reaching a "telematic outlet" placed wherever necessary.

Let us consider a schematic diagram of the VIRGO buildings, with the various relative distances.

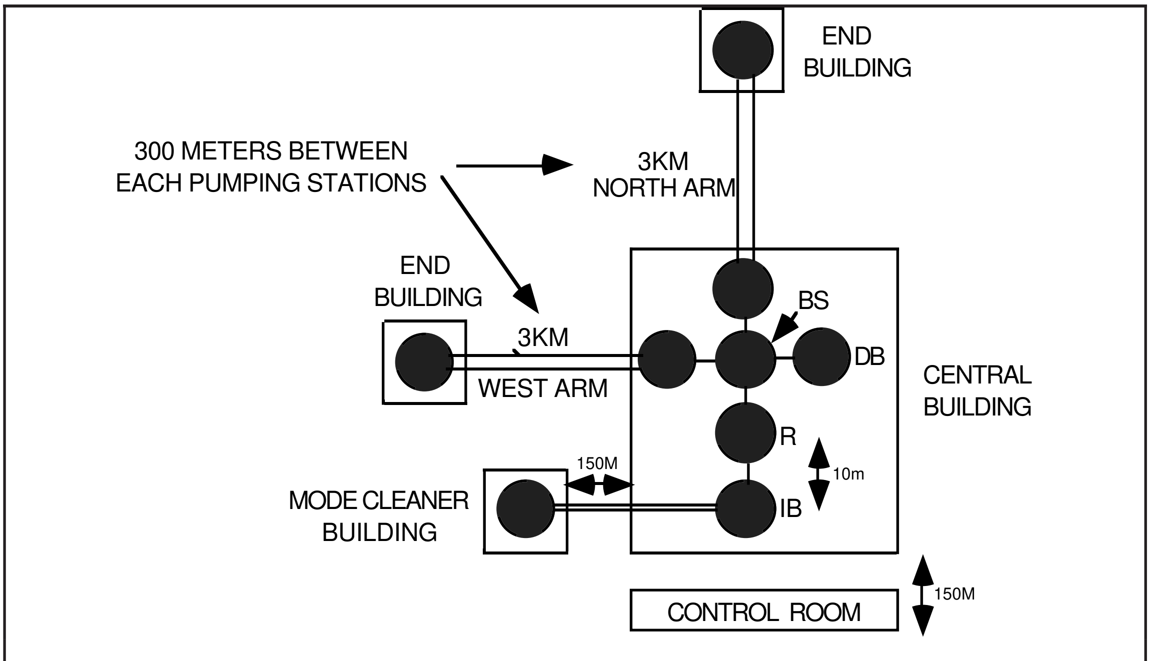

figure 5300.1 Virgo Buildings

It is necessary to use fibers from one building to another, and twisted pairs within the buildings, as far as possible; these latter can be STP (Shielded Twisted Pairs) or UTP (Unshielded Twisted Pairs). This kind of cabling accomodates both audio communications (telephone) and video (remote monitoring), as well as ETHERNET (10 base T) Serial (RS422) and FDDI (over copper).

Optical fibers will also be used to reach dedicated systems, e.g. for control of servoloops, timing and acquisition.

The central system will be placed in the Control room.

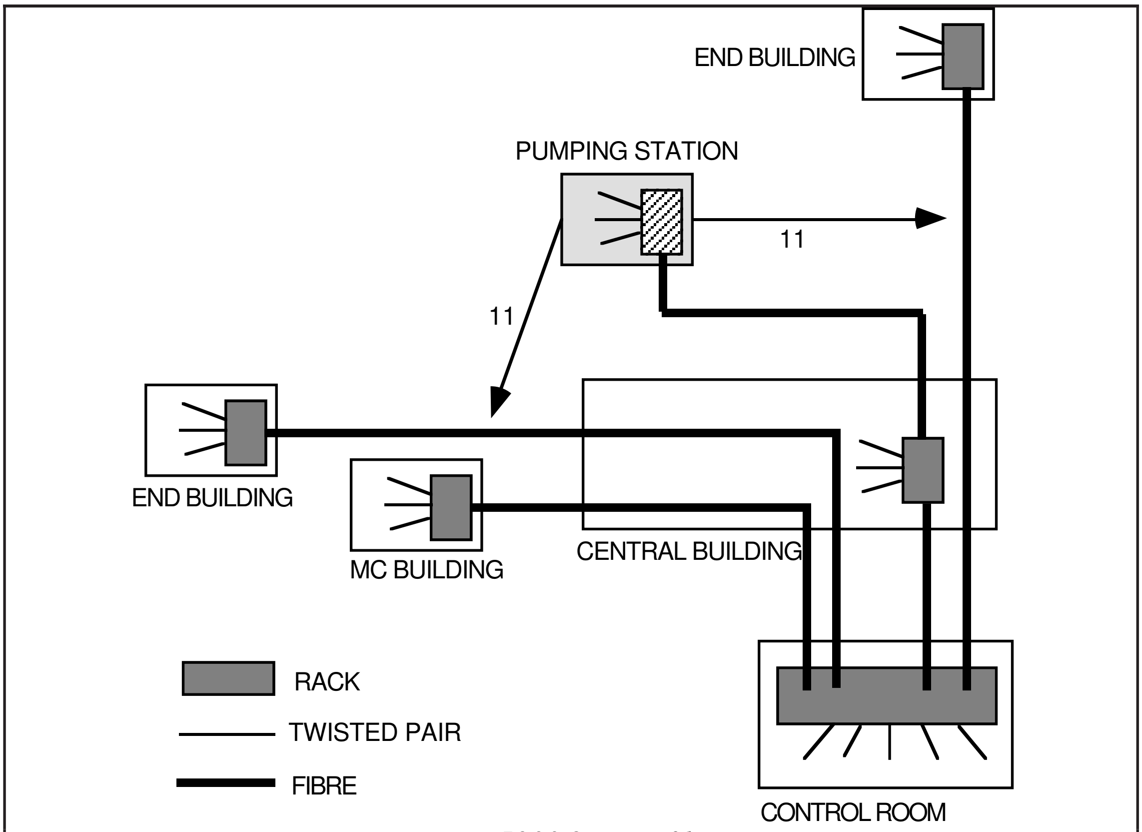

Figure 5300.2 Main fibers

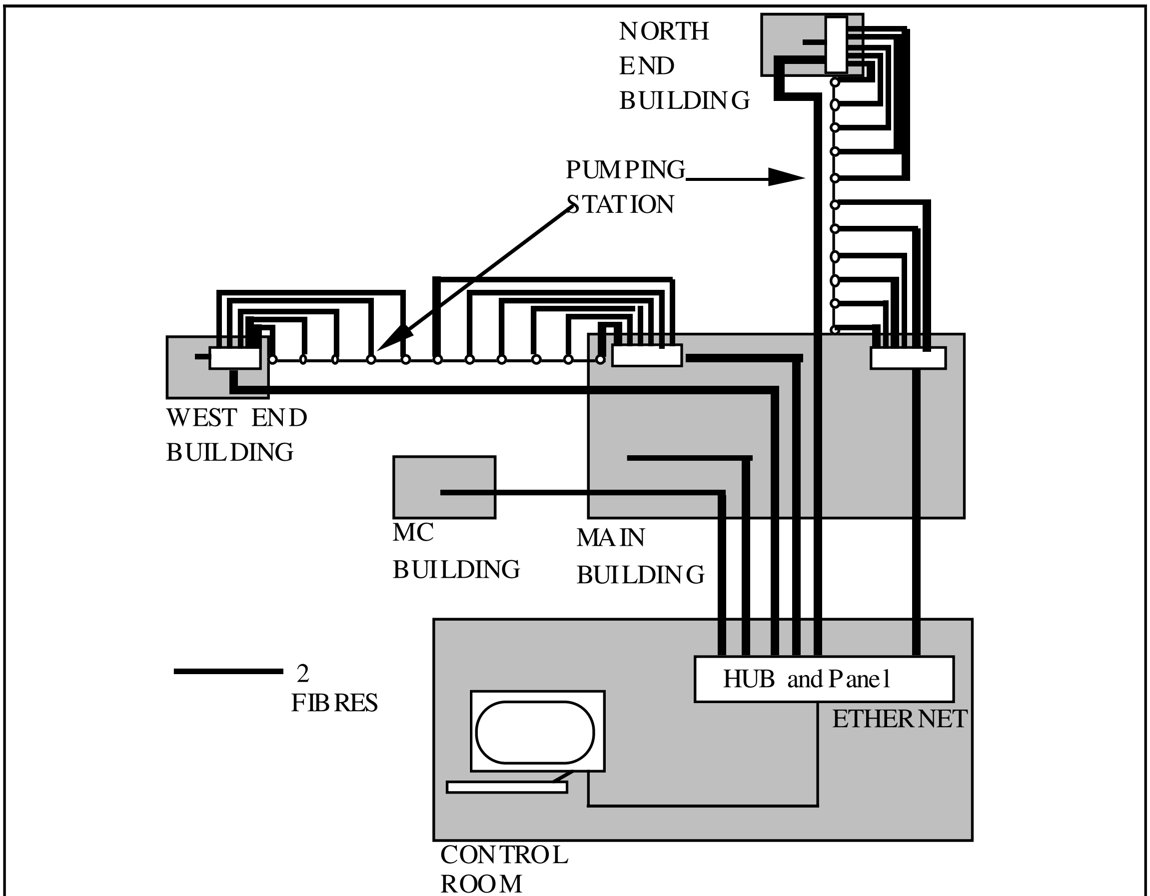

Figure 5300.3 The slow monitoring network.

We can give the number of fibres needed between the buildings:

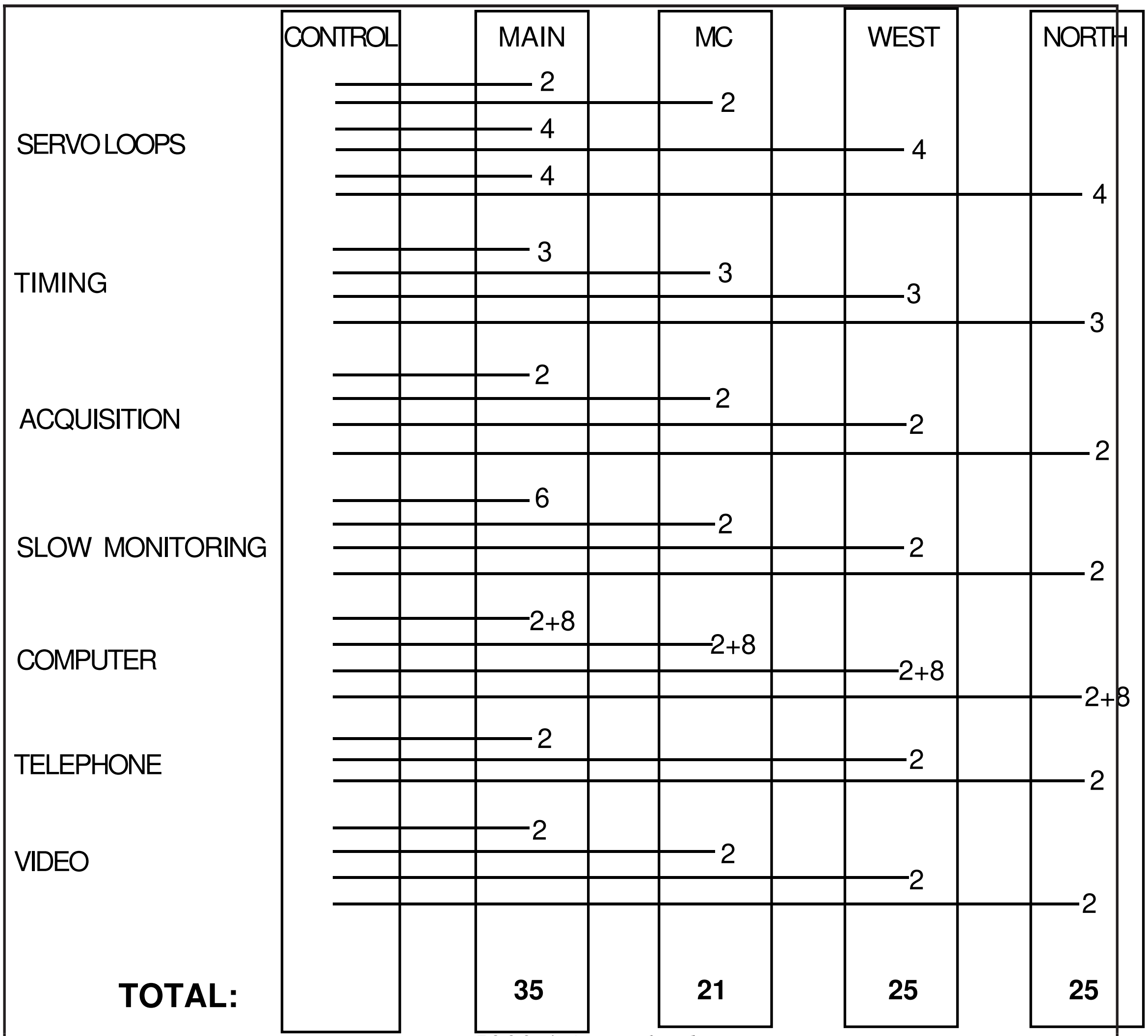

Figure 5300.4 Optical Fibers

We have 3 cables (one for reserve)with 18 fibres . The total length would be 21KM.

For the Slow Monitoring, we also need cable for the Pumping Stations. We would have cables with 4 Fibres (2 for reserve). A total of 15 KM will be necessary.

For Telephone, we will also need 15 KM of Twisted pairs cable to go into the Pumping Stations.

In the 5 buildings of VIRGO, we find a Fibre Distribution Panel which receives all the fibres and distributes them to the Hub for networking, or to the other systems as necessary. All this is done via a Patch Panel for both fibres and copper.

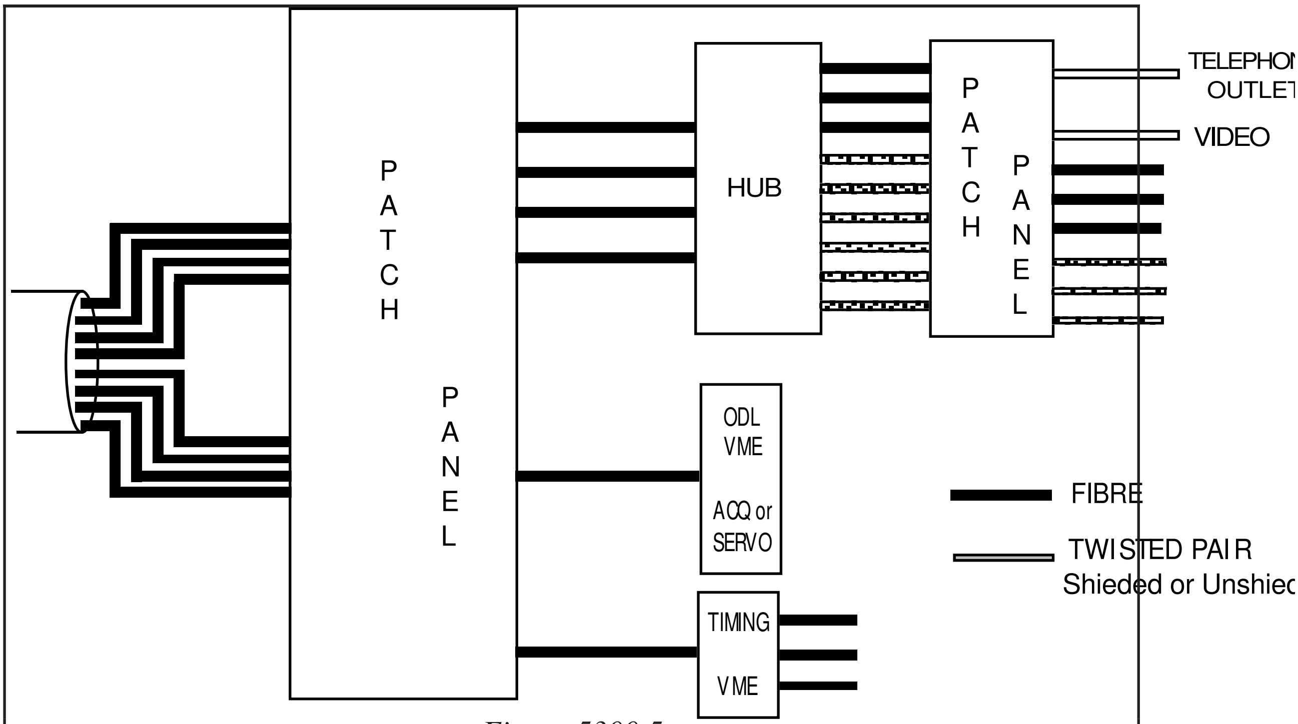

Figure 5300.5

In different buildings, each distribution system is a rack (Secondary); the number is dictated by the necessity to have only 100m cables from Sec to "outlets". Each rack accomodates a "patch panel" for configuring each outlet according to needs, and a "hub" with the intelligent part for computer network, or slow monitoring (repeaters, bridges, terminal servers...). The racks has to be powered via Uninterrupted Power Supplies, and uses about 1KW

The main system is also a rack, which contains a router for connections to INFNet, a module for connection to a telephone system (PBX) as well as a "hub" for communication via fibres with the secondary systems

Each "outlet" shall have 4 connectors of RJ45 type, all identical for phone, video, computers, and so on. The destination will be made on the patch panel in the SEC system, not in the outlets. The cable from the outlet to the equipment may change according to needs.

A PTT line for Telecommunications to the INFN in PISA or, better, to CNUCE in PISA (this latter is on the backbone of the Italian Network, which is 2 Mbit/s) is needed.

5400. Data acquisition and Online processing

The configuration of the interferometer requires the implementation of a distributed readout system (figure 5400.1). In each building, a local readout system collects the available data and sent them via a FIFO memory connected to a digital optical link. In the control room building, the central data acquisition task collects the data from the digital optical link. It structures the data into frames which are written to tape and distributed to the online processing. All these tasks are synchronized with the timing system. For safety, the frame builder checks the reference time of each local system against the global time reference.

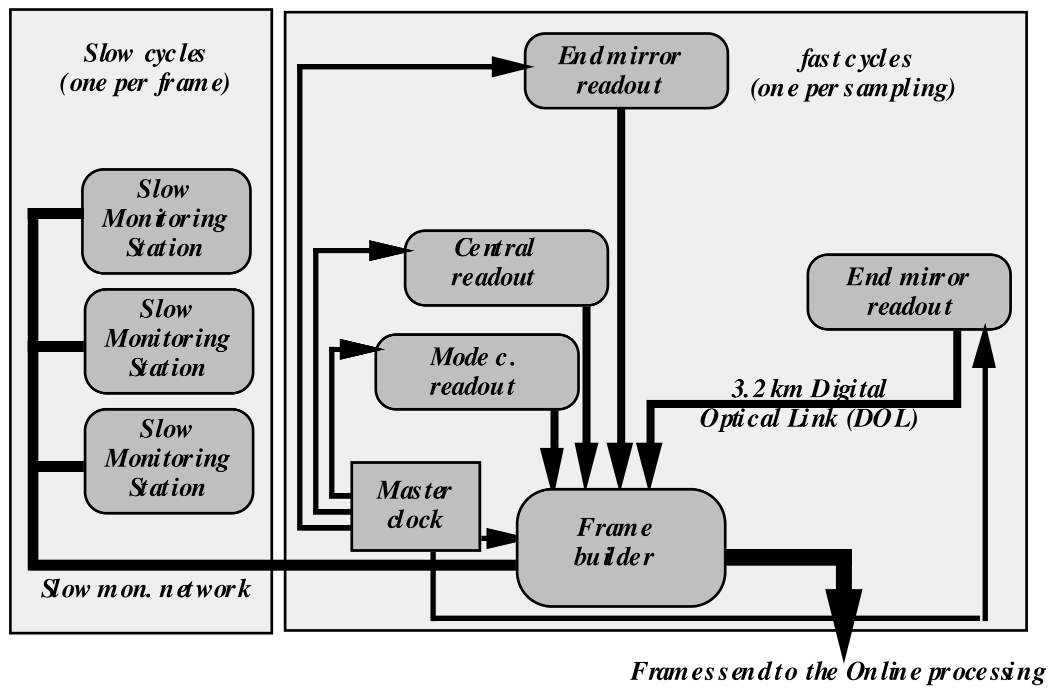

figure 5400.1 The distributed data acquisition system

5400.1. The local readout system.

On reception of the interrupts delivered by the central timing system (corresponding to one sampling of the dark fringe signal), the local readout system shown on figure 5400.2 reads the connected VME boards and builds a small data set. This data set, which is of fixed length, is made with the informations provided by the different captors, status registers, feedbacks and GPS timing module. Once assembled, it is written to a FIFO memory acting as an interface with the digital optical link connected to the central DAQ.

The figure 5400.3 gives a detailed description of the timing cycles performed on a local system after reception of the acquisition interrupt. While the expected sampling frequency is about 10kHz, the system is designed to run up to 20kHz in order to accommodate testing sequences with two times more data. The ADC readout frequency could be a fraction of the dark fringe sampling frequency for the signals which doesn't needed faster readout (alignment for instance). Some redundant control words are added to the data set to check the reliability of the transmission to the frame builder. A given measurement is identified by its position within the data set.

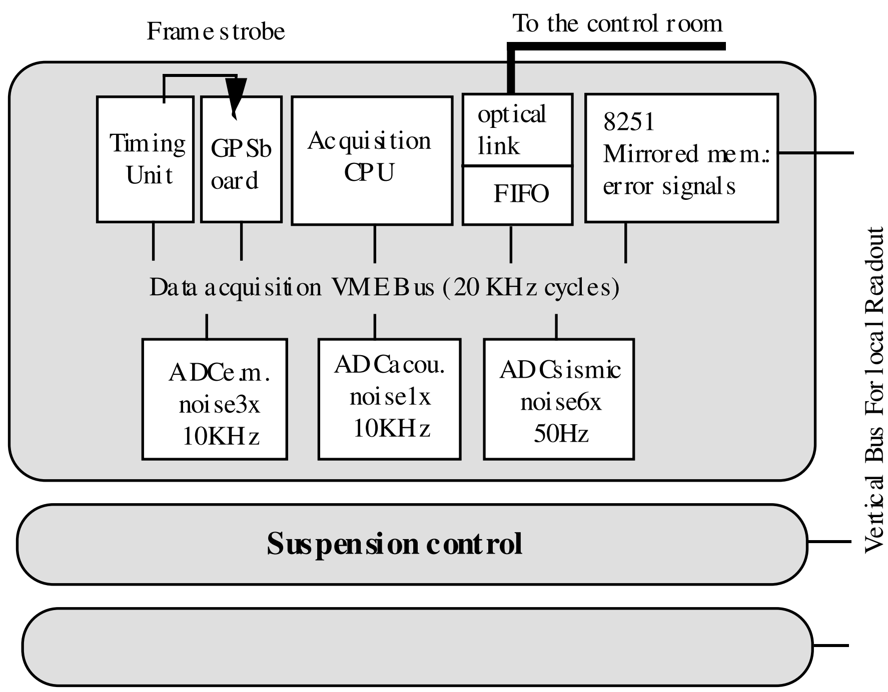

figure 5400.2 A local readout system

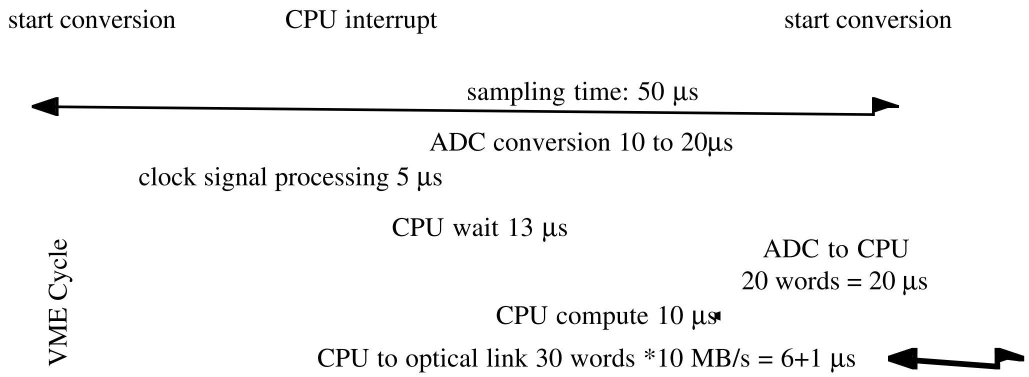

figure 5400.3 The timing sequence of a local readout cycle

5400.2. The frame builder

The architecture of the frame builder is shown on figure 5400.4 and its timing sequence on figure 5400.5. To accommodate for the readout time and the transmission time, there is one sampling cycle offset between the local readout timing and the frame builder timing.

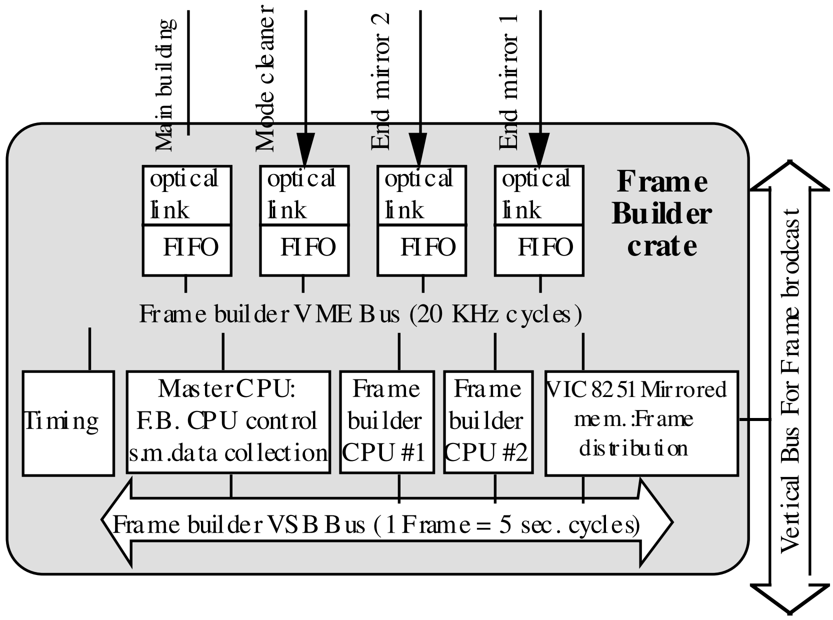

figure 5400.4 Block diagram for the Frame Builder.

Synchronized by the clocking sequence, one processor of the data acquisition system, frame builder 1, collects for one frame the data sets sitting in the different FIFO memories while another processor, frame builder 2, assembles together those from the previous frame to form and distribute ‘frames’ to the storage and online processing system through the VSB and vertical bus (see figure 5400.4). The slow monitoring data are collected in parallel by the master CPU and included in the frames even if the fast readout is not working.

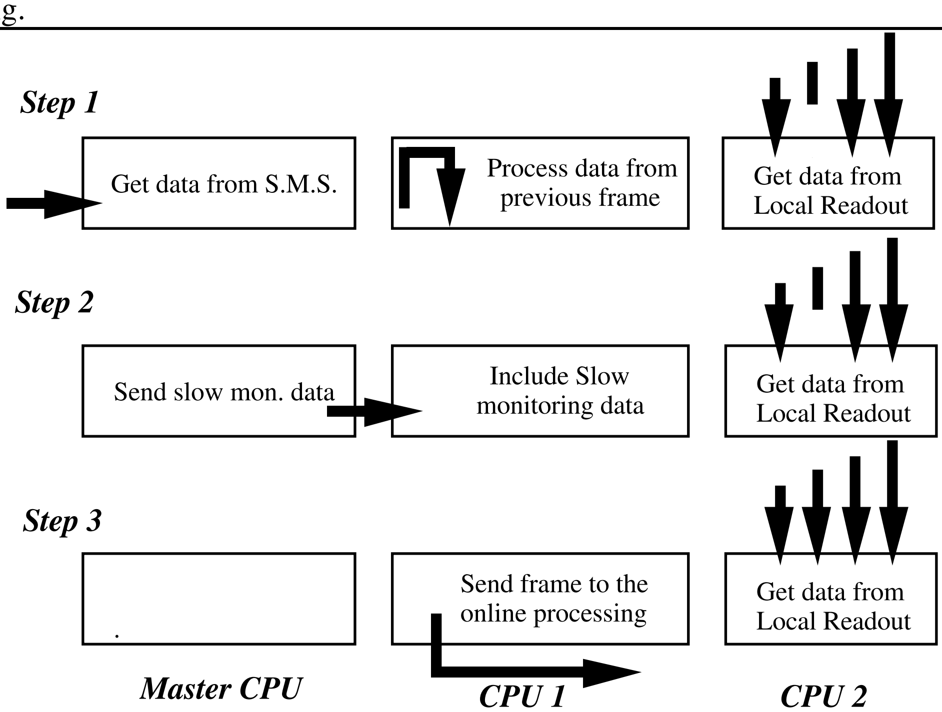

Figure 5400.5 The three steps performed by the frame builder for each frame. After these steps CPU 1 and CPU 2 switch their tasks.

The content of the various FIFO’s read by frame builder 1 is a stream of data sets collected according to the clocking sequence. To speed up their further processing, these sets are compressed and structured by frame builder 2. Then frame builder 1 and 2 switch their task for the next frame. The two frame builders CPU's are synchronized by the master frame builder CPU.

The table below gives a first estimate of the amount of data to be recorded. Only the parameters recorded at high rate are quoted. The slow monitoring stations are expected to provide a much smaller amount of data (40 SMS station x 100 words x 10 bytes = 40kBytes/frame).One distinguishes three types of parameters: the ones used to compute the signal (feedback, dark fringe), the ones which may be used to improve the signal/noise (seismic noise for instance) and the ones used for quality check.

Signal	# of chanels	rate	bits
Longitudinal photo. signals(5 signalsx3)	15	20KHz	16
longitudinal feedback (5 mirrors+laser)	6	20KHz	16
Alignment signal (2x4 quadrant diodes)	16	1KHz	16
Alignment data (5mirrors x 2corrections)	10	1KHz	16
Laser power&frequency	2	20KHz	16
Mode cleaner locking (signal + correction)	2	20KHz	16
Mode cleaner alignment	8	1KHz	16
Seismic noise (6 per tower)	6x9	100Hz	16
Acoustic noise (one per building)	4	20KHz	8
elec. mag. noise (one per building)	4	20KHz	8
Line voltage (one per building)	4	20KHz	8

This list leads to a total average rate of the order of 1.5 MBytes/sec. This number is certainly not final and will evolve before the start of data taking. It gives however an order of magnitude to dimension the required online performances. Few MBytes/sec. have to be added to estimate the maximum data rate to take into account data recorded for debugging like internal data coming from the second stage damping of the suspension. This explain our specification of 10MBytes/sec.

5400.3. The frame structure.

An frame is a unit of information containing all the information necessary for the understanding of the interferometers behavior over a finite time interval which integrates several sampling. It contains thus not only the sampling performed during the integrated time interval, but also those performed at a frequency smaller than the frame frequency.

To simplify its manipulation, an frame is organized as a set of C structures described by a header holding pointers to additional structures and values of parameters expected to be stable over the integrated time interval: the starting time of the frame, its duration, values produced by the slow monitoring. This header is followed by an arbitrary number of additional structures, each holding the values of a rapidly varying parameter like the main signal, the seismic noise. Each active element producing data at a rate higher than the frame rate is thus accumulated in a dedicated structure.

This frame structure is a standard which has to be conserved over the various stages of the analysis. Frame history, detector geometry, trigger results, monitoring data, reconstructed data, simulation results lead thus just to additional structures. It is always possible to add new structures or to drop old ones.

The input and output routines used for their creation and the specifications of their format have to be unique and kept as stable as possible. Their format has to be saved on tape to allow for automatic processing. More technical aspects will be described later in the software rules section.

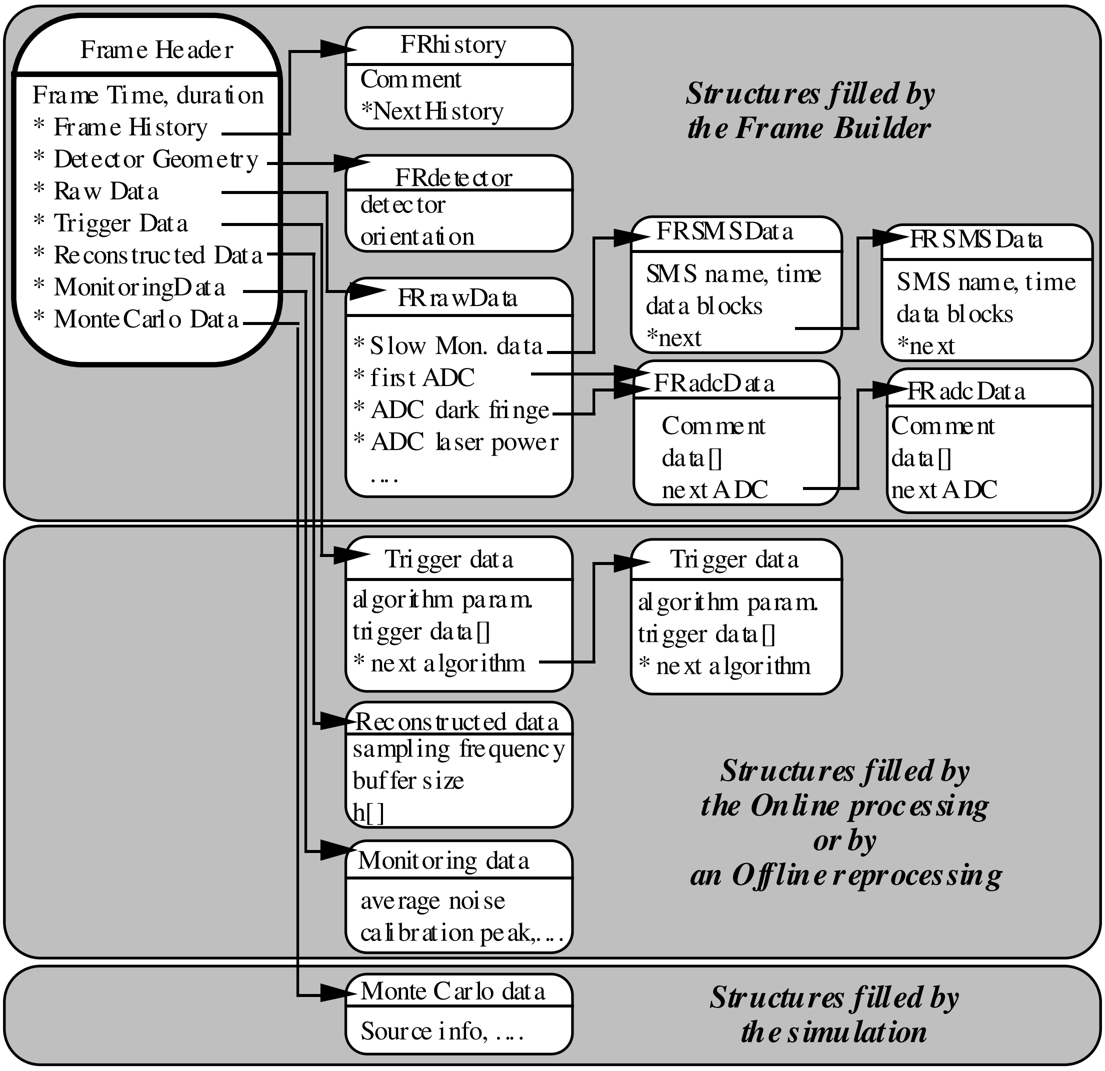

Figure 5400.6 The frame structure

5400.4. The Data Display.

This is a tools to look at the content of Frames (from disk files or from the Data Distribution) and provides tools to plot the data in various way. It could be used online to survey data or off line to play with simulated data or reconstructed data.

5400.5. Online processing

In parrallel to the raw data archiving, Tthe online processing has mainly three tasks (see figure 5400.7) described in the following sections. All these task are run in CPUs connected in parallel to the Frame builder vertical bus.

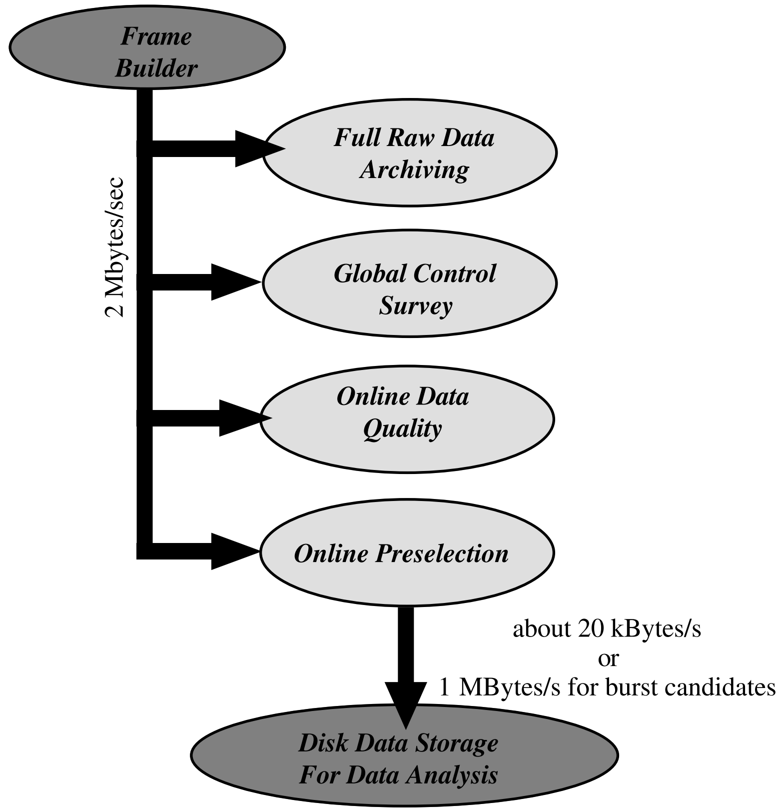

figure 5400.7The principle of the online processing

5400.5.1. Global Control Survey

The purpose of the Global Control Survey is to complement the real time component of the Real time Global Control.

Its main role is to detects unwanted trends in the various servo-loops and, more generally, to react on faulty behaviors thwarting the locking of the interferometer. It should be in a position to perform all correlations deemed necessary to fulfill its task. Its decisions are reached on the basis of the overall response of the interferometer, taking into account as much as possible, all the information available at this level. Its other function is to permit the development onlline of Global Control softwares.

5400.5.2. Data monitoring and quality

This online task will survey permanently the data quality produced by the interferometer. It uses the signal induced by the calibrators, the noise level and run data quality algorithms. It should provide a fast feedback in case of problems. It is a real time data quality check. The corresponding informations are stored in the Data storage system to allow further data selection according to the data quality requirements of the offline analysis. It may be run has part of the Trigger system.

5400.6.3. Data Filtering or frame trigger

Let’s remember the data rate of 1MBytes/sec in continuous mode. This means about 1 DAT tape (4 GBytes) every hour and translates into about 9000 tapes/year. Such an amount of data is not a problem for the online writing or for the storage, but it may very quickly overflow any data analysis. Therefore it may be wise to reduce their amount for the detailed data analysis. A typical goal would be a reduction of more than one order of magnitude. The final reduction factor will in fact be driven by our data analysis capability and by the data production rate. Remember that this rate is also function of our operational efficiency.

Prior to any kind of physics analysis one has to convert raw data (i.e. ADC counts provided by the main feedback and the remaining dark fringe signal) to physical quantities and to compute h values. This is the first task of the trigger. This computation requires the knowledge of the 'transfer functions' of the various controls and of the calibration constants. These quantities are available from a special database accessible from any Virgo laboratory. This reconstruction should be the same as the offline reconstruction.

Then trigger algorithms looking for burst events are run. Triggered frames are defined within 'time windows' during which binary coalescence or burst candidates may occur. These candidates will be selected using simple and robust search algorithms to minimize a possible bias in the further offline analysis: selection efficiency has to supersede rejection efficiency. Random triggers are recorded to monitor the trigger algorithm efficiency. To cover a large type of astrophysical events, several trigger algorithms will be run in parallel.

The informations kept after trigger selection are written to Data Summary Tapes (DST). They are the followings:

For each frame:

- reconstructed h value resampled at lower frequency (1 or 2 kHz) : This represents 4 to 8 kBytes per second. It allows long term pulsars searches.
- slow monitoring records (less than 5 kBytes per second) to follow permanently the interferometers environment.

For 'triggered frames':

- all the raw data required to perform a full analysis of the signal candidates.

The triggered frames are send to the data storage system using a standard network.

Of course, false signals will be selected. But this filtering is not the final data analysis and a lot of work is still needed to extract and proof the existence of the gravitational waves from the DST tapes. The achieved data reduction should provide the conditions for a fast analysis and a fast data dispatching over the collaboration. Most of the data analysis will thus be performed on this filtered sample. But remember that it will always be possible to start again from the original raw data sample.

The filtering algorithms will be designed and tuned using simulated and real data. They will certainly change and improve with some learning experience. This is why we are prepared to reprocess the original raw data. Dedicated computers will be installed on the site for this purpose.

5500. Data archiving storage and distribution

5500.1. The Building Control Station

It controls the parameters describing the status of the buildings, like temperature, pressure, humidity and similar parameters. It uses a distributed I/O acquisition protocol (ETN) which is interfaced to a standard VME environment. In case of need there is the possibility of commanding some actuators in the building. For each building it is foreseen a VME crate, controlled by a local CPU, running the operating system LynxOS (building control server) and linking each server to the slow monitoring network (Ethernet). A user interface running on one of the control room workstation will provide the possibility of managing the whole building control network, setting the configurations of each server and, if necessary, displaying the history of the single parameters.

5500.2. The Environment Monitoring Stations

It monitors all the environment parameters, that are the parameters not directly connected with the interferometer, but necessary for the correlation analysis. Typical parameters which must be monitored are:

• electromagnetic noise	sampled at 10 kHz or faster
• acoustic noise	sampled at 10 kHz or faster
• seismic noise	sampled at 1 kHz
• cosmic ray noise	
• all power supplies	
• weather	pressure, wind, temperature, etc.
• human activity	count number of people in or close to the buildings
• status information	

Due to the fact that some of these variables have to be sampled at quite high frequency and due to the need of a precise timing (electromagnetic noise and acoustic noise) part of the environment monitoring system will be integrated within the fast VME data acquisition system (local readout). The other parameters are sent directly to the frame builder by a standard slow monitoring system, that consists of a dedicated VME crate, controlled by a local CPU, running the operating system LynxOS (environment monitoring server) and linking each server to the slow monitoring network (Ethernet). The sensors are defined and chosen within the environment monitoring structure with direct acquisition by the system or internal sampling and data transfer to the environment monitoring system via a standard RS232.

An user interface running on one of the control room workstations will provide the possibility of managing the whole environment monitoring network, setting the configuration of each server and of the sensors. Such user interface has one graphical page for each type of parameter. The history of each environment monitoring parameter can be displaied on this workstation using the Historical Monitoring software in connection with the Data Distribution System.

5500.3. Raw data archiving

The first task of the on line processing is that of archiving all the frames produced by the frame builder in order to have a complete archive containing all the information acquired by the interferometer, which allows a future full reprocessing of the original data if necessary. The raw data archiving gets the data from a VME CPU reading the vertical bus and writing them on tapes. For security reasons, two copies of the raw data are made and stored in two different locations.

The architecture of the raw data archiving will be defined at the latest possible time, because of the fast technological evolution which may render obsolete the present time solution. Anyway, according to the frame data rate and the typical output on tape drives, the present time technical solution is that of a parallel staging of the data on disk before copying them on tapes. In this case several CPU are needed which work in parallel.

5500.4. Data Distribution

The data distribution system collects the data produced by the on line processing and by the frame builder and store them on disk. At the same time, acting as server, it distributes all the requested data to the authorized Virgo user interfaces. These operations have different priorities: the highest priority is assigned to the data collection from the frame builder and from the on-line processing, while the lowest is assigned to the requests of data from users.

To avoid any interference between these two different operations the system must use a dedicated Ethernet line for Data Collection and another Ethernet line for serving the Data Requests. Cm protocol is used for both the operations

The structure of the stored data is organized according to a tree structure, as shown in the figure 5500.1.

```
     EXP Name                                               EXP Name
       |                                                       |
  ----------------------------------             ---------------
  |             |            |                   |          |
 RUN   N# T#      RUN   N# T#   RUN N# T#           RUN N# T#   RUN N# T#
  |
 -----------------------------------------------------------
 |                |                |                |
 FRAME N# T# Q#   FRAME N# T# Q#   FRAME N# T# Q#   FRAME N# T# Q#

with N# = number, T# = time, Q# = quality factor
```

figure 5500.1

In this way, the stored data have the same structure defined by the frame builder and by the on line processing (frames). Therefore the user interface access the Data Distribution System getting blocks of data (frames) exactly as stored. In its requests it must specify the experiment name (EXP#), the run RUN#) (number or starting time) and the frames (FRAME#) (numbers or starting time/ending time).

On request the Data Distribution System will provide also specific parameters within specified intervals of time (or number of frames). Because this kind of request would mean a cross access to many frames with a considerable loss of time for some parameters, the Data Distribution System also stores some parameters contained in the frames according to a table structure (as in a classic database), which surely guarantees a faster and more efficient data access.

The access to the Data Distribution System is restricted only to Virgo labs and only for reading and getting data. The access to the stored data for changing their structure, for deleting and moving files and for all the relevant operations of management of the system itself, will be restricted to authorized people.

5500.5. Historical Monitoring

The historical monitoring is a user interface running on any workstation, which, acting as a client to the Data Distribution System, can retrieve and present the data stored in the Data Distribution System (i.e. making histograms, etc). The Historical Monitoring works with the parameters stored as function of time by the Data Distribution system, while the Data Display works with full frame.

5600. Simulation: SIESTA

With the increase in size and complexity of the gravitational wave experiment like the Virgo project, it becomes very important to be able to predict the behavior of the detector before its construction, in order to optimize all the components. Such predictions could be done in first approximation by analytical computations, but if one looks for a detailed description or for a study of a correlation between effects like mechanical motion, thermal noise and light propagation, the only realistic approach becomes the numerical approach. Therefore, the first goal of the simulation is to be a tool for the detailed design of the Virgo detector.

The natural next step is to use the simulation as a tool for the understanding of the behavior of Virgo as a detector by comparing its results to simulated data. This will be important when the detector starts working, and even before, when individual components are tested. Simulated data may also be used as inputs to exercise the data acquisition system.

Finally, the last task of the simulation is to allow the study of search algorithms on well known data . The level of description we are aiming at will allow us to study real and fake signals in a quantitative way, as it is done for big experiment in High Energy Physics. We think this will be a very important point when real data are analyzed.

To meet all this requirements one uses two complementary approaches. First one uses a set of small simulation programs, adapted to the study of well defined problems. Then one develops a general purpose program (SIESTA) which can ideally study all the problems related with correlations. Such a program is usually run with more emphasis on a specific issue. More informations about SIESTA could be found in the Siesta directory with documentation and examples

5700. Table of content