"5000 Electronics and Software" revised November 6, 1995, received from Benoit Mours/VIRGO (mours@frcpn11.in2p3.fr).

Distributed by H. Yamamoto 2/23/96.

B. Barish
K. Blackburn
R. Bork
M. Coles
D. Coyne
A. Lazzarini
F. Raab
G. Sanders
D. H. Shoemaker
R. Vogt
R. Weiss
S. Whitcomb
H. Yamamoto
M. Zucker
Chronological File
Document Control Center

LIGO - ~~P950024~~ -00 - D
T950153-00-D

5000 ELECTRONICS AND SOFTWARE

(Rev. November 6, 1995)

Introduction

This chapter describes all the needed activities starting from the output of the local electronic up to the data ready to be analysed 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 and the hardware implementation 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 information 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 a 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 sent to a storage and distribution system which is the experiment 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.

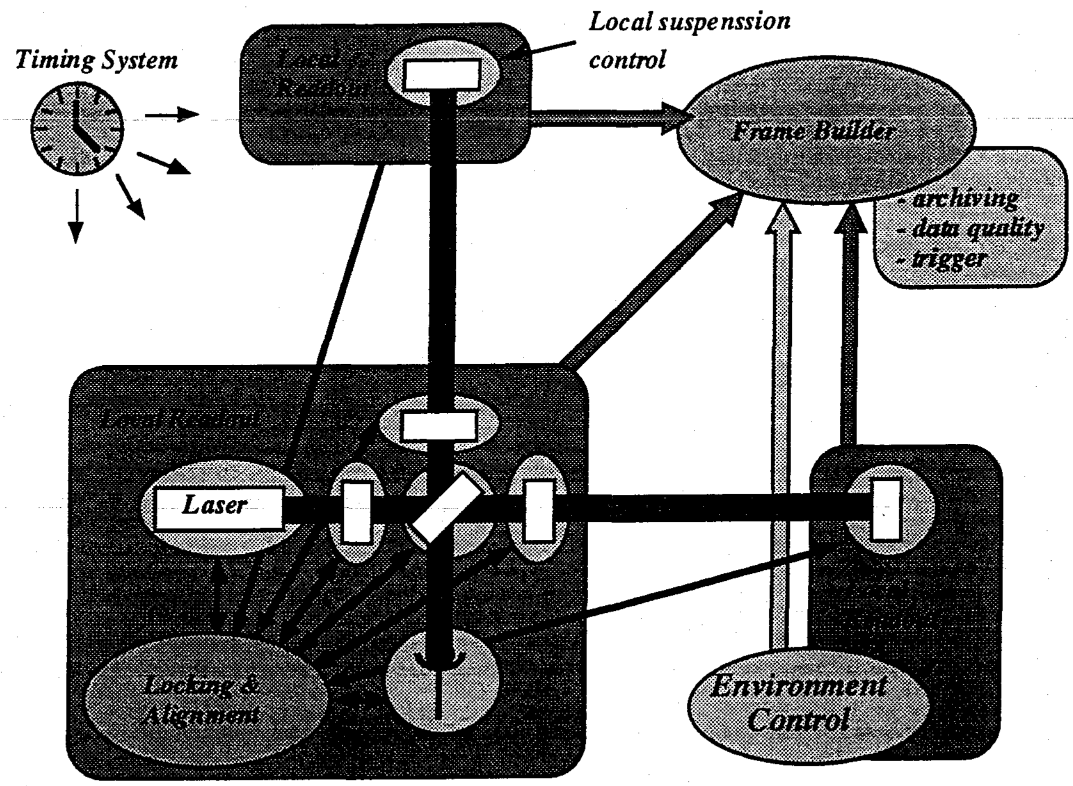

figure 5000.1 The online architecture

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/s,
- 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 sensors 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' sensors, processors and actuators connected with 'fast' links. To execute the various feedback loops within a constant time the system has to be built 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 loops 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 information can be exchanged. Higher level control processes transfer their information 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 sensors 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.

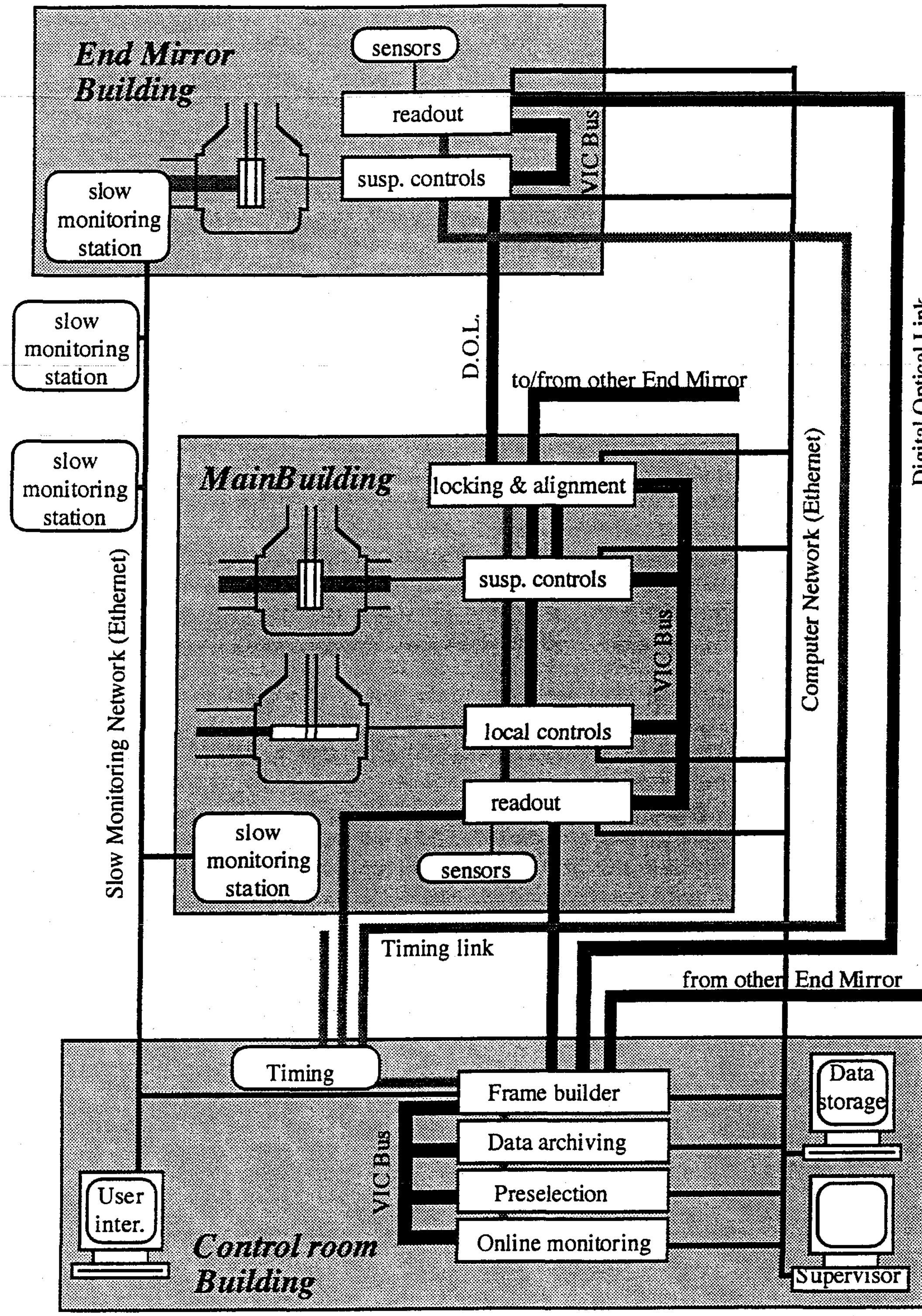

Figure 5000.2 The network architecture implemented between one end mirror building, the main building and the control room.

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 sensor 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 proccesses, only data for test, configuration parameters and status information should be exchanged using the computer network.

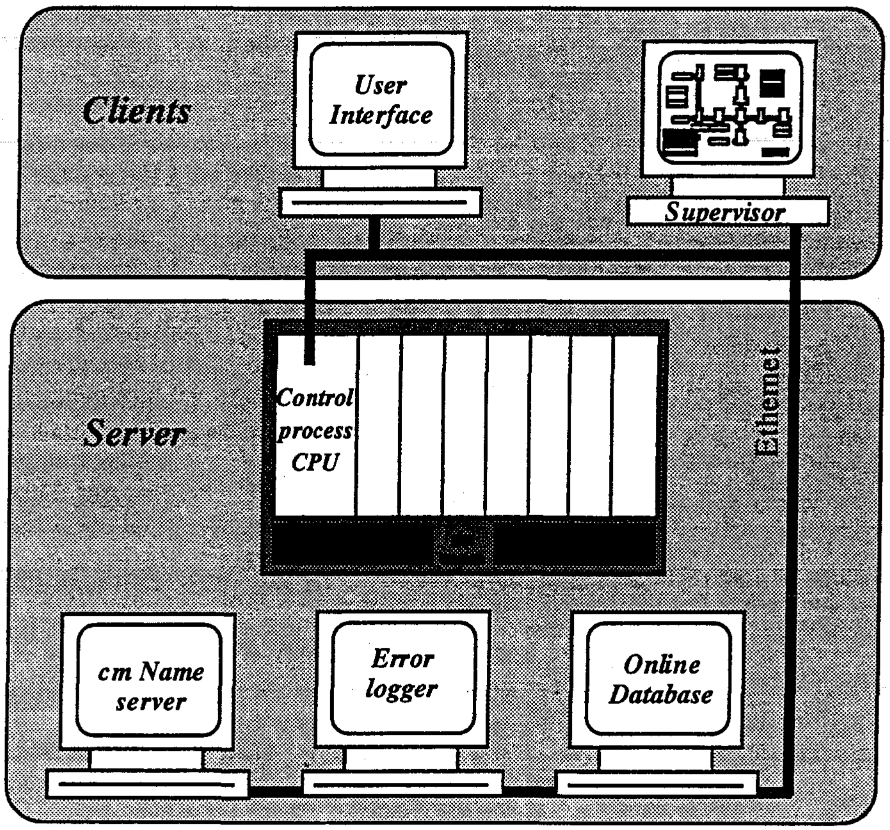

Figure 5000.3 client/server process organization

The user interface (a client of the control process) configures the server and/or monitor and displays the data provided by the server. Several user interface clients can be simultaneously used to monitor its information, 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 do not need to be real time.

The Cm library is used for ALL the communications between processes on the computer network. This package provides platform independent communications tools and access to the processes using logical names. 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 messages is given in table 5000.1. Figure 5000.4 and 5000.5 give 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 gives 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 5000.4):
• 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 tries 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 an intelligent configuration editor. The Online Data Base stores also all the information about privilege access to the server. This is done by using the machine name and the user account as selection parameters.

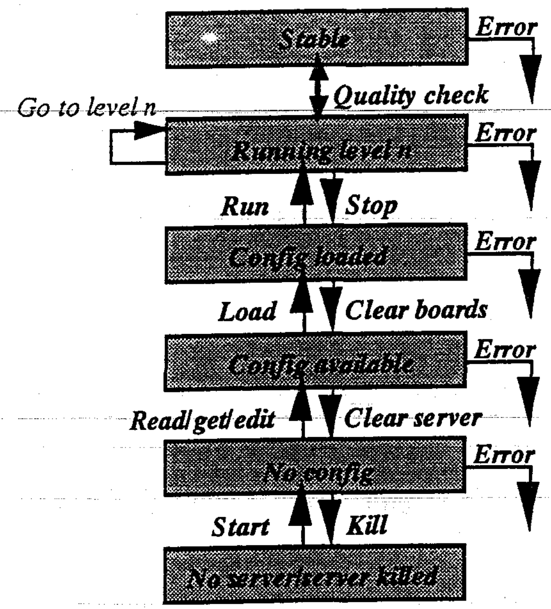

figure 5000.4 Server states. Changes from one state to another is triggered by one of the messages given in table 2.1 except for the error and for running/stable which is done using some quality check.

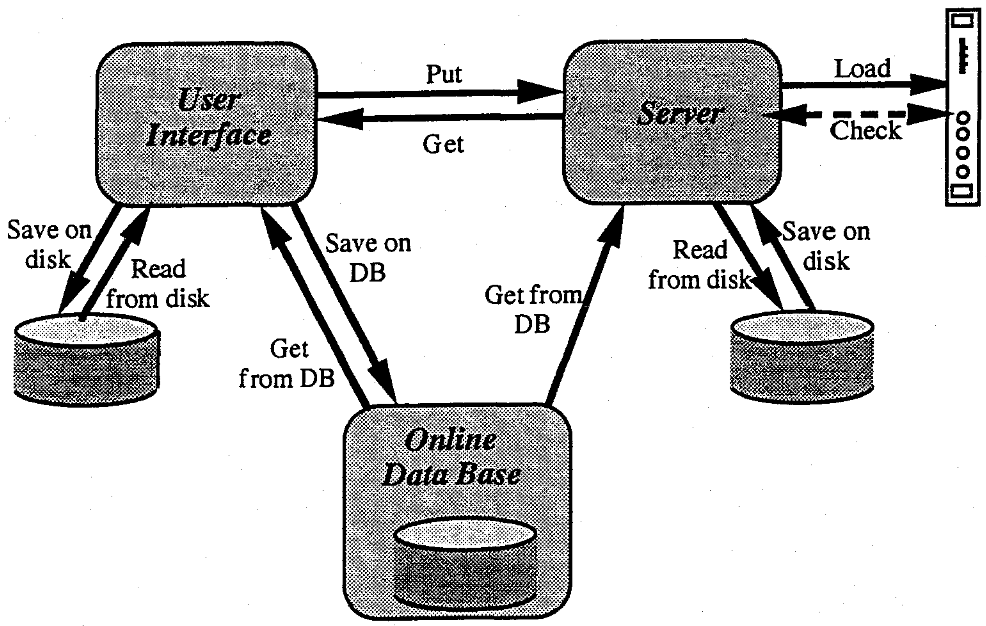

Figure 5000.5 The possible moves for the configuration files

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 runs the server. The first protection will be to limit the number of accounts on these processors. Then the server is driven by a client which runs on a different computer. Since all connections use Cm, the next protection will be the limitation (using domain 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. Therefore we can restrict the connections to a limited number of accounts run on a limited number of workstations. The server will get these information 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 highest 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 interferometer controls, 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.

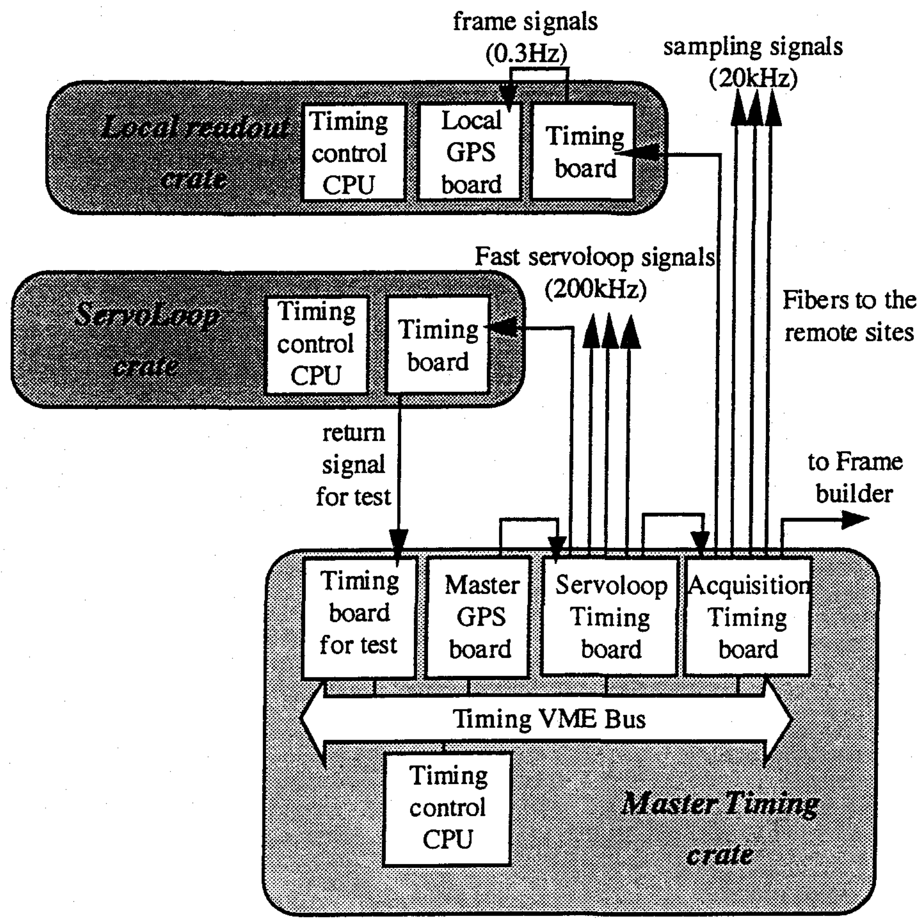

figure 5000.6 The timing system

Whereas the different sensors 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 clocks will be minimized. We forse three basic clocks:

• The fast servo loop clock (about 200 kHz). It will be used to oversample ADC used for precise measurements (signal detection or suspension damping)
• The sampling frequency (10 to 20 kHz). It will be used for most of the digital servo loop (locking, suspension control) and for the main data acquisition
• The seismic frequency (100 Hz). It will be used to record monitoring accelerometers.

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 loops. This shows 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.

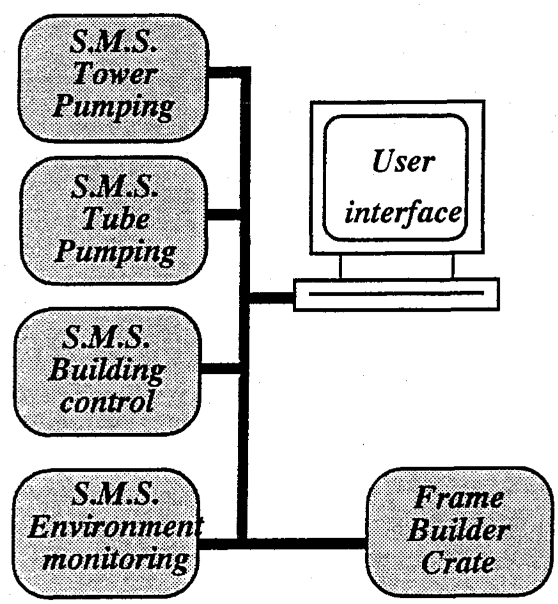

Figure 5000.7 Logical organization of the slow monitoring

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 sent 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
CPartChl 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 bandwidth, 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.

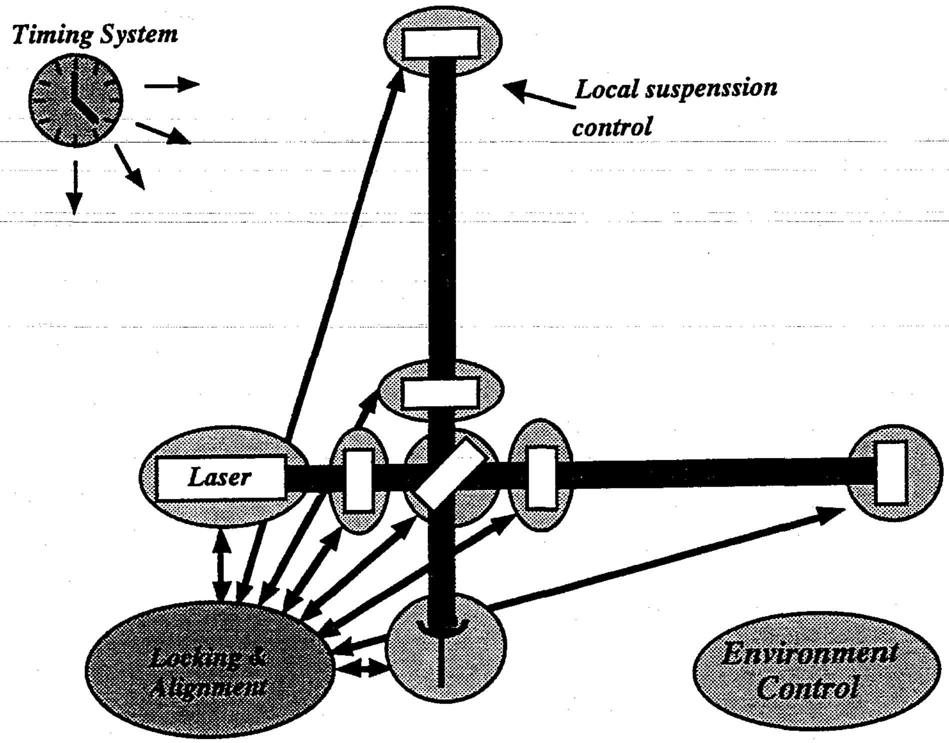

figure 5000.9 Global view of the controls.

Figure 5000.10 presents the chain of hardware components 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 transfer time between the components, given their relative distances). The status of each component is controlled by the supervisor using the computer network, while the real time data are exchange using dedicated hardware connections (DOL). The data exchanged 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 & alignment: the mirror absolute position from the local position, the force applied on the mirror and any relevant 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 . Strong connections 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.

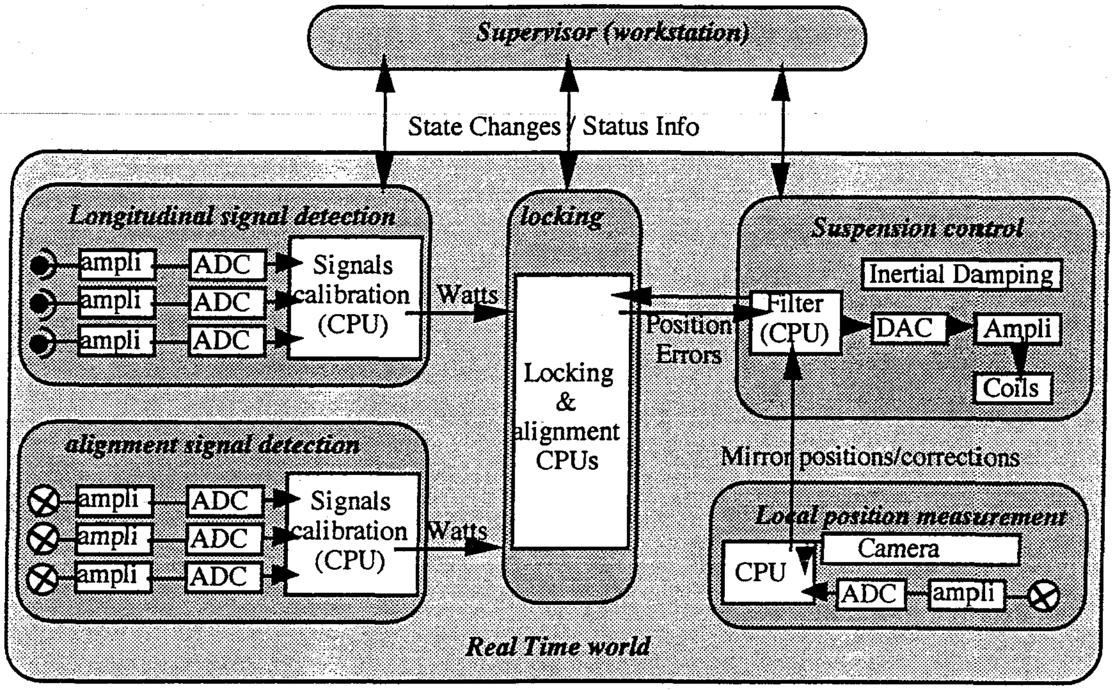

Figure 5000.10 Principle of the control implementation.

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

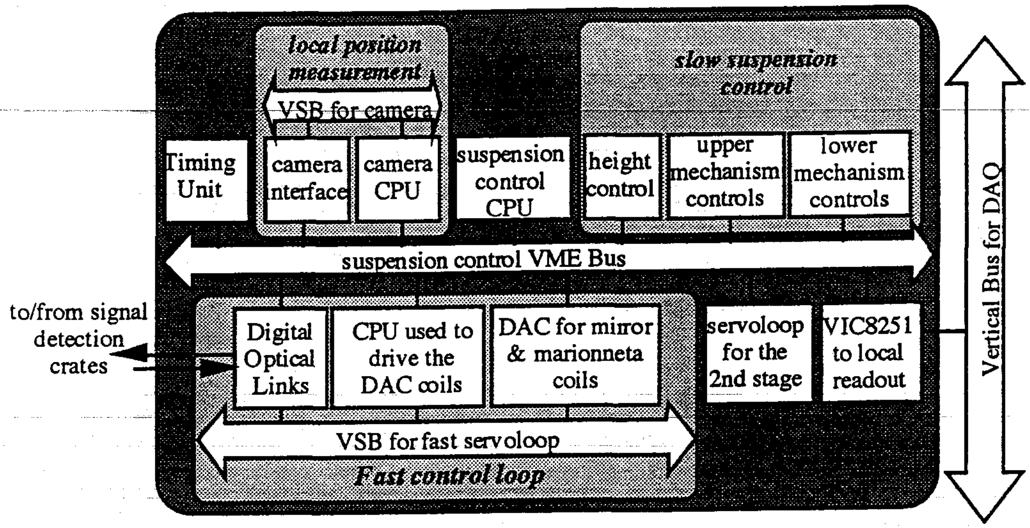

Figure 5000.11 VME crate for a suspension control (crates for the analogic electronic are not shown)

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 cameras.

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

Each suspension control provides status information 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 of reading 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 on the detection bench or on the input bench. Twin systems are foreseen 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 channels than the four real signals. The photodiode ADC's are sampled at fast sampling rate (typically 200 kHz) and an inboard CPU reduces 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 photodiodes have two outputs with two different dynamics corresponding to the two types 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.

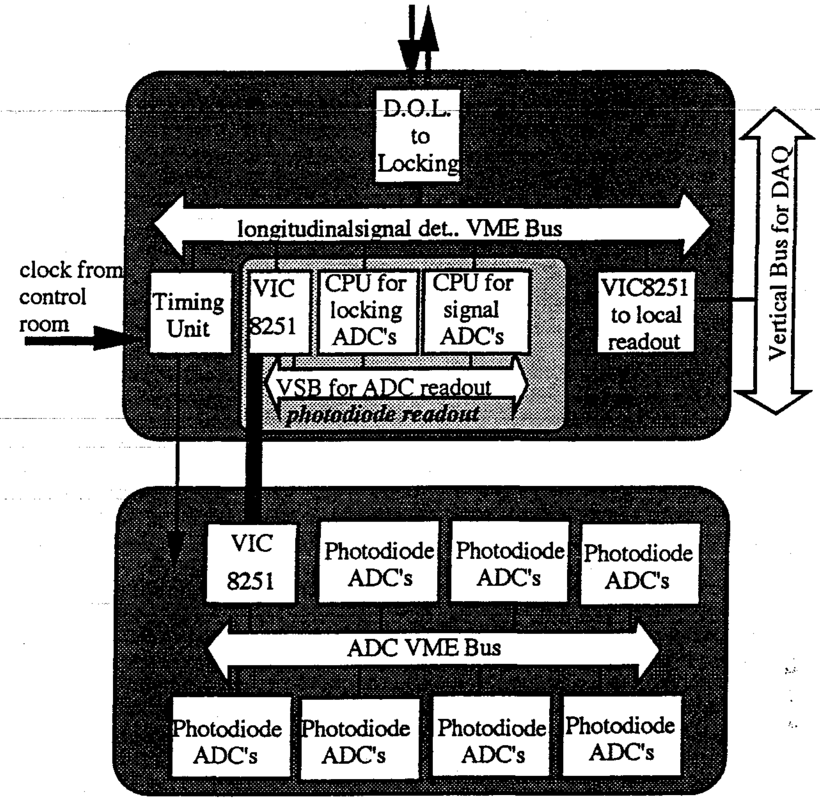

figure 5000.12 The signal detection crates

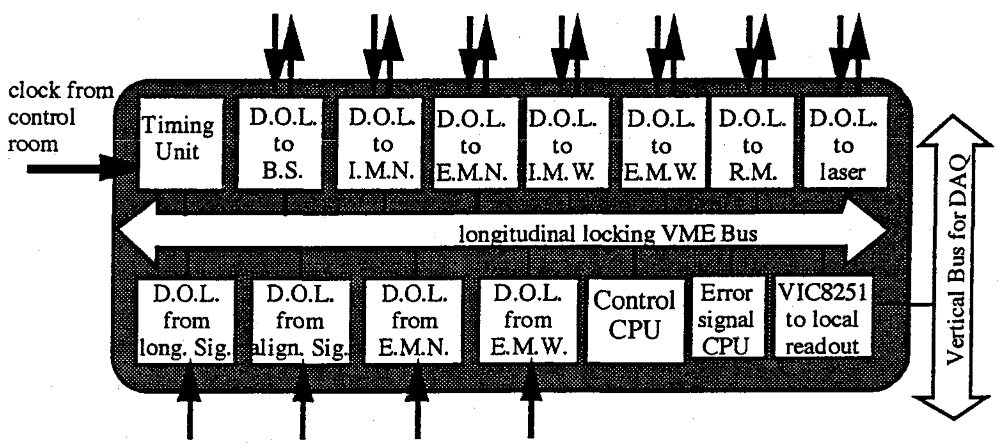

figure 5000.13 The Locking&alignment crates

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 laser running (power suply, prestabilization, ...). Fast information could be : exchanges with the input bench control and data sent to the data acquisition (local readout). Status information are collected using the slow monitoring network.

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 servoloops 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 provides status information using the slow monitoring network.

5000.3. The control and monitoring of the interferometer environment

This 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 monitoring of the diffuse light could also be connected to the tube control station.

A G64 (or a VME) crate is foreseen for each control station. The electronic is controlled by the local CPU. It provides a set of vacuum measurements performed at regular time intervals. Summary information are sent on request to a global vacuum control, monitoring user interfaces, 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).

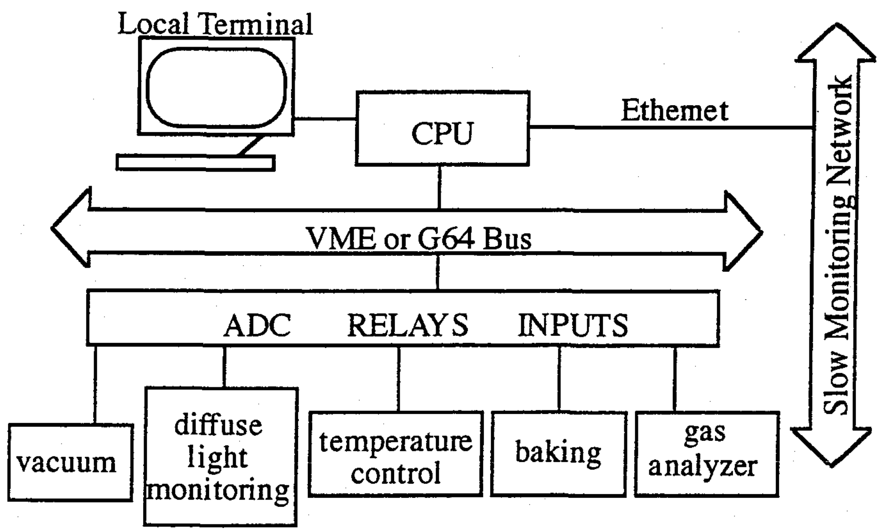

Figure 5000.14 A tube control station

5000.3.2. The Tower Control Station

It drives the pumping system and the tower clean room installation. It controls the backing of the tower, the tower temperature and the gas 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 set of measurements performed at regular time intervals.

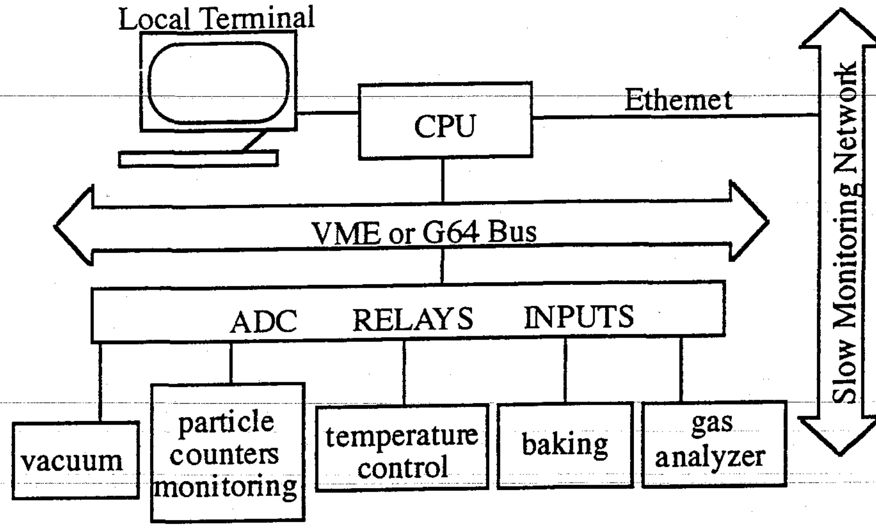

Figure 5000.15 A tower control station

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 provides 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 the beam profiles 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 sent 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.

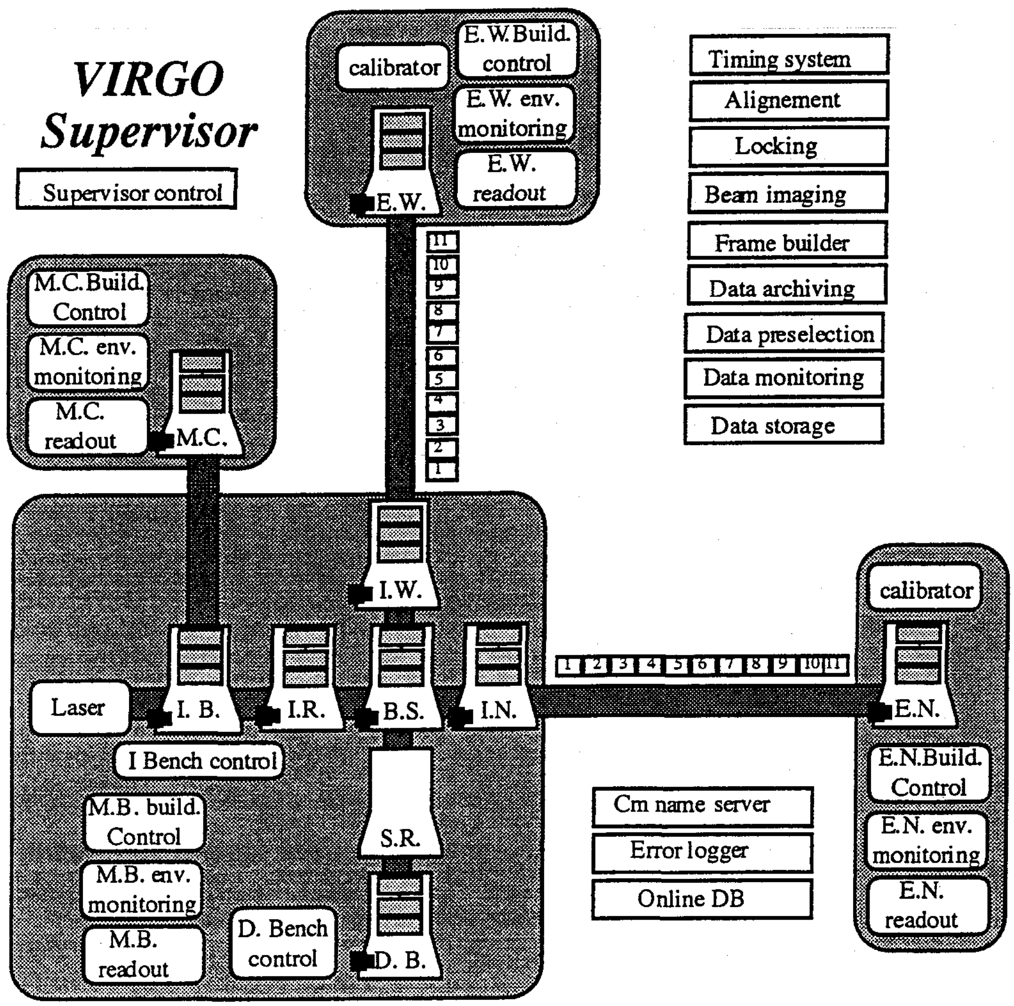

Figure 5100.1 The Supervisor

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 names,...)
- 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

This package manages task to task communications by providing a wide range of mechanisms required to manipulate transparently the network layer (TCPIP) and uniformly across the various platforms. A complete description is available within the Cm distribution kit. However, one may summarize the basic features it implements as follows:

Cm is meant to manage the task to task communications (sending or receiving messages) running on heterogenous machines (with different architectures or operating systems) without limitations on the number of active connections apart from those induced by the operating systems.

The set of tasks or applications that may participate this network define a Cm domain managed by one special application - the NameServer - in charge of the physical addressing scheme, allowing several independant such domains to coexist.

An application with which a connection is requested is referenced by a name, that must be unique within one Cm domain and that does not need to mention anyhow the machine on which it runs, nor the transport charectiristics (such as TCPIP parameters).

The central manager application NameServer is in charge of every mechanism for name registration, port number allocation, and physical addressing operations transparently for the user application.

The Cm Message part handles the formatting issues, insuring that a structured information frame is transmitted across the network (reaching heterogenous architectures) without loosing its integrity.

Elements of security management are introduced by the internal protocole, by transmitting the host name and user name of the message senders.

5100.3. The ErrorLogger

The error logger collects the information and error messages from all the processes. The messages are sent using the El library which uses Cm.
Each message contains 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 followed to fix the problem. The server goes one state down for severe error.
- fatal — the server is stopped 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.
Query operations can be requested by aplications giving selection criteria based on the date, the severity, the source, the message. The diagnostics and monitoring applications will be the normal users of the error logger.

5100.4. The Online DataBase

This package is meant to handle the online configuration database. It is based on one server (although several servers may coexist) responsible for the accesses to the stored objects (controling the protections, the selections and the manipulations) and of the effective storage medium for the objects. Access to the data base will be achieved from any application through the network. (DB is based on Cm)

Data items consist on named C objects and can be of most legal C type.

Management of items is centralized in a DB server (that also handles access control on objects)

Items (or objects) are accessed by their nam transparently over the network and selections of set-of-objects can be done with various criteria (name pattern, conditions on nvalue, ...)

An history of the contents evolution for every object is maintained by the system and recoveries of object values are possible on the basis of azbsolute or relative time and date specifications.

Objects are eventually stored into textual files (using a human readable format) that may even permit a non-network access to objects.

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 file 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. This directory and the software documentation could be accessed using WWW at: `http://lapphp.in2p3.fr/virgo/sdoc/doc.html`

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 has a single person in charge of the updates. The standard version of the software is found by using the 'pro' directory which contains logical links on the various packages. When several people are working on the same package, CVS is used as code manager.

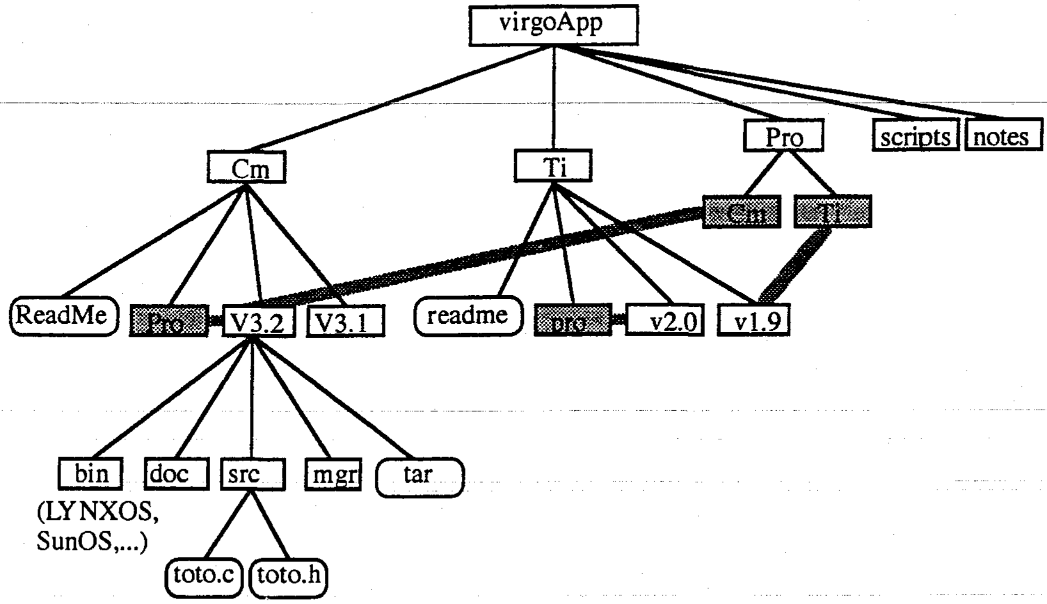

figure 5200.1 The file organization

Key letter	Package	Manager
Bc	Building control	F. Barone
Ca	Calibration	Annecy
Cm,Ci,CS,CSet	Communication	C. Arnault
Ds	Digital servoloop	F. Barone
Da	Data archiving	F. Barone
Db	Online database	C. Arnault
Dd	Data distribution	F. Barone
Do	Digital Optical Link	F. Bellachia
Dy	Data Display	X. Grave
El	Error logger	C. Arnault
Em	Environment monitoring	F. Barone
Fb	Frame Builder	F. Bellachia
Fr	Frame Lib	B. Mours
+(UA,UC,UH,UJ,UM,US,UV,c_H)		
Gx	Imaging system	C. Drezen
Hm	Historical monitoring	F. Barone
Ib	Input bench-control	Orsay
La	Laser control	Orsay
Li	Linear alignment	Frascati
Lr	Local readout	F. Bellachia
matlab	Matlab files for Virgo	B. Caron
Nl	Non-linear alignment	Frascati/Roma
Ob	Output bench-control	Annecy
Re	Data Reconstruction	Annecy
Scl	Control language	C. Arnault
Sd	Signal detection	Annecy
Siesta	Simulation	F. Marion
(+El,Fm,Fo,GR,IO,ME,MI,OP,PW,Th)		
Sm	Slow monitoring examples	Annecy
Su	Supervisor	Orsay
Ti	Timing system	F. Bellachia
Tu	Tube control	Orsay
To	Tower control	Orsay
Tr	Online trigger	Annecy
Xs	Siesta Graphical Interface	Annecy

Table 5200.1 Packages and key letters 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:	CALI

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' identified with a run number.

The data format on tape is either an ASCII file or a binary file. Data are written in ASCII format or in native binary format. Special macros (UJ_READ... and UJ_PUT...) are availables to read/write the various C data types.

A data file should start with a file header with the following data:
* for ASCII format:
  - the word BSCII (for version 1)
* for binary format:
  - byte 1 = the format version (current version number is 1)
  - byte 2 = sizeof(short)
  - byte 3 = sizeof(int)
  - byte 4 = sizeof(long)
  - byte 5 = sizeof(float)
  - byte 6 = sizeof(double)
  - bytes 7-8 = a short containing 0x1234 (to check the bytes swaping)
  - bytes 9-12 = a long word containting 0x01020304
  - bytes 13-16 = a float containg PI
  - bytes 17-24 = a double containg PI

The files is structured in records. One record content one C struct. For each record there is a short header with the following data:
* for ASCII format:
  - a string to give the type of the C struct
* for binary format: three 4 bytes 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 (function UJDicGetId). The correpondance with the structure name is given by the dictionnary (UJstructH and UJstructE) which are 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 using the function UJDicGetId.

The content of the C struct follows this header. Data are written in the same order as in the structure. Character string are preceded by their length coded as a int (4 bytes) in binary format. When a pointer is given, the size of the data is given by the latest interger word. The file starts by the detector description (if any) which is 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 exchange between client and server, we just send one binary file per frame with the same format as the one used to write on tape.

More information 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 sending a mail to VIRGO-L@FRCPN11.IN2P3.FR. VIRGOS-L is for software people and VIRGO-F for VIRGO 'Friends'. WWW servers are available in most of the labs. More information about the news system and old news could be found at:

`http://lapphp.in2p3.fr/virgo/news/news.html`

5200.2. Summary of the 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 (n from 1 to 6)
C5: Data quality
 1 VIC, n CPU (n around 3)
C6: Trigger
 1 VIC, n CPU (n around 3)
C7: Online monitoring
 1 VIC, n CPU (n around 3)

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),
 [2 CPU (signal processing), 2 VIC (to ADC crate)](VSB),
 [3 Camera, 1 CPU](VSB)
C24: Signal detection
 1 VIC (to det. bench control), ≈ 8 ADC (fast) (about 64 channels)
C25: Signal detection (locking channels)
 1 VIC (to det. bench control), ≈ 8 ADC (fast) (about 64 channels)
C26,27,28,29: Signal detection for detection bench
 Analogic electronic (8 diodes)
C30: 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)

C31: Alignment:
Analogic electronic
C32: Locking
5 CPU (control &signal processing), 1 VIC (to DAQ), 1 Timing, 11 DOL
C33: Calibration
1 CPU (control &signal processing), 1 GPS, 2 ADC/DAC (fast)
C34: Environement & building control
1 CPU (control), dedicated network hardware
C35: Local readout
1 CPU (control &signal processing), 1 VIC (to collect DAQ), 1 Timing, 1 GPS,
1 DOL (to frame builder), ≈ 5 ADC (fast)
C36,38,40,42,44,46: 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)
C37,39,41,43,45,47: 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: Environement & building control
1 CPU (control), dedicated network hardware
C74: Local readout
1 CPU (control &signal processing), 1 VIC (to collect DAQ), 1 Timing, 1 GPS,
1 DOL (to frame builder), ≈ 2 ADC (fast)
C75: 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)
C76: 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 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 used instead of one in order to do system maintenance without stopping VIRGO. Here is the list of the foreseen workstations:

- W1: Tube control & Tower control (on slow mon. net.)
- W2: Building control & Environment control (on slow mon. net.)
- W3: Laser & input bench control
- W4: Detection bench control & Signal detection
- W5: Suspension 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 powerful workstation with lot of disk space and special hardware.
- W13: Network analyser
- W14: Software distribution

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)

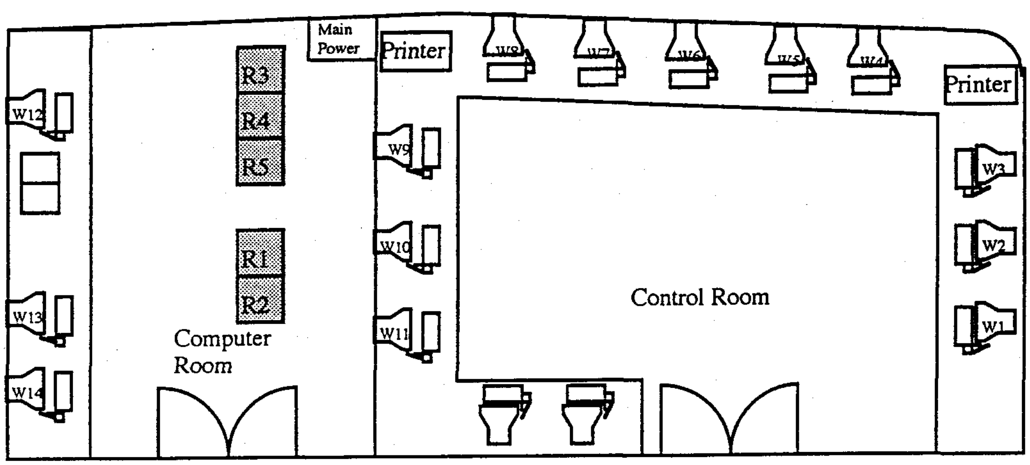

Figure 5200.2 Control Room and Computer room. Video monitor (for beam imaging or for room survey) will be installed above the workstation

5200.3.2. Main Building

A 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.)

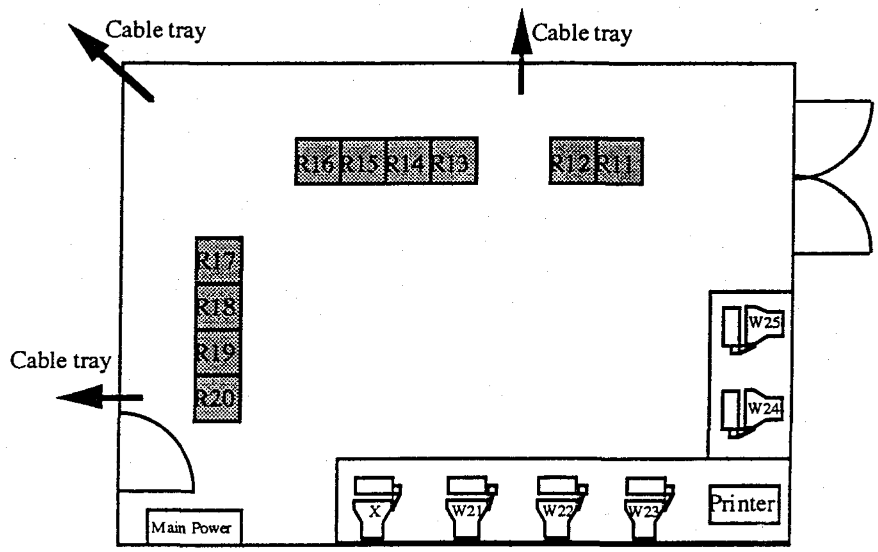

Figure 5200.3 Control Room and Computer room. Video monitor (for beam imaging or for room survey) will be installed above the workstation

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. The Video system

The requirement for the video system will evolve with time, from the survey of the mirror installation to the control of the smooth running of the interferometer. Therefore the video system should be easily reconfigurable. This means that we will have standardized cameras, monitors, cables and plugs.

We described here a reference solution where most of the monitors will be located in the control room. Since we have a large number of cameras, we will have choosers in the control room to decide which cameras are displayed. We will also use 'quadravision' to group four cameras in one monitor. It will be possible to forward the image of few selected monitors to the data acquisition room or near the optical benches to help for their tuning. It will be also always possible to insert a local monitor near the corresponding camera. A tape recorder will be available in the control room to keep track of some installation procedures or of the interferometer start up for instance.

5200.4.1 List of cameras

Human activity control:

The purpose of these cameras is to monitor the various rooms, the benches and mirrors installation and to survey their operations. The corresponding monitors will be installed near the Building control workstations. Colour cameras will be used to survey the halls to improve the visual resolution (cameras V1 to V8). During the mirror installation inside the tower, we will use the alignment cameras to monitor the operations. We should foresee a different set of lenses to increase their field. The list of cameras is the following:

V1	Clean Room	(colour)
V2,3	Gallery 1 ang 2	(colour)
V4,5	Hall 1 and 2	(colour)
V6	Data Acquisition Room	(colour)
V7	Mode Cleaner Hall	(colour)
V8	Laser room	(colour)
V9	Laser Bench	(black & white)
V10	outside Input Bench	(black & white)
V11	outside Detection Bench	(black & white)
V12	North End Mirror Hall	(black & white)
V13	West End Mirror Hall	(black & white)
V14	North End Mirror Gallery	(black & white)
V15	West End Mirror Gallery	(black & white)
V16	North End Mirror Bench	(black & white)
V17	West End Mirror Bench	(black & white)
V18,19	Central building views	(black & white)

To reduce the cost of the 3.2km transmission, the camera signals from an end mirror building will be grouped together with a "quadravision", with a monitor in the end building and another one in the control room.

Mirrors and Benches position measurements:

The position of each suspended object is monitored with two cameras, one with lens and one without lens (see alignment section of the FD). These are high precision cameras read by a VME board. The purpose of the video monitors is to provide a fast check for the auxiliary laser beams and for the camera operation and also to survey the mirror using the cameras equipped with lens. During the running of the interferometer, we will mostly display in the control room the cameras imaging a mirror (6 cameras).

V21	Input Bench Positioning	V31	(camera with lens)
V22	Mode Cleaner Mirror Positioning	V32	(camera with lens)
V23	Recycling Mirror Positioning	V33	(camera with lens)
V24	Beam Splitter Positioning	V34	(camera with lens)
V25	Detection Bench Positioning	V35	(camera with lens)
V26	Signal Recycling Positioning	V36	(camera with lens)
V27	Input North Positioning	V37	(camera with lens)
V28	Input West Positioning	V38	(camera with lens)
V29	End North Positioning	V39	(camera with lens)
V30	End West Positioning	V40	(camera with lens)

Beam Imaging:

A set of high precision cameras read by a VME board is installed to image the beam at various positions in the interferometer. They are used to align the benches and the interferometer and also to monitor the running conditions. The purpose of the video monitors is to have a fast check in addition to the digital images.

V41,V42,V43,V44	Laser & Input Bench
V45	Recycling mirror screen
V46	Mode Cleaner
V47	End of north arm bench
V48	End of west arm bench
V50,51,52,53	Detection Bench

Tower separating roof position measurements:

The position of suspension wire inside the conductance will be monitored by a camera. A mirror located near the conductance will provide a side view of the wire position. These cameras will be used to adjust and survey the suspension wire. Therefore we will use a set of black and white cameras. Their images will be displayed in the control room to survey separating roof displacement. Additional monitors could be installed near the tower .

V61	Recycling Mirror separating roof positioning
V62	Beam Splitter separating roof positioning
V63	Input North separating roof positioning
V64	Input West separating roof positioning
V65	End North separating roof positioning
V66	End West separating roof positioning

5200.4.2. Digital processing of the signals

In order to get a fast treatment for the position measurements we will have one processor for each couple of cameras looking at one mirror or bench (10 CPU). Then six more processors will be dedicated to the beam imaging. One for the cameras located in the laser and input bench, one at the end of the mode cleaner, one at the end of each arm, one for the recycling mirror screen and one for the cameras located on the detection bench. Of course we need one VME interface for each camera. Most of the image processing is done by the local VME processor. But the image presentation and the global non linear alignment analysis will be performed on various workstations and especially on the imaging workstation.

5200.4.3. Monitor installation in the Control room

Two colour monitors will be reserved in the control room to view the various halls (cameras V1 to V8). A chooser will select 2 of the 8 inputs. These monitors will be installed near the building control workstation. One of this input will be forwarded to a colour screen located in the data acquisition room.

The bench monitors and the end mirror building survey will be send to another chooser also located near the building control workstation. The separating roof positioning images will be send to the same chooser. For the end mirror, the separating roof positioning will replace the gallery camera in the quadravision during the permanent operations.

The signals coming from the alignment cameras (V21 to 28 and V31 to 36) will be send to a chooser (16 inputs, 2 outputs) located near the alignment workstation.

Four monitors will display the most useful information for the interferometer running: the signals coming from the input and output benches (V41 to 44 and V50 to 53), the four images of the transmitted beam (V45 to 49) grouped in one image and the images of the four mirror cavities (V37 to 40) also grouped in one image.

The figure 5200.4 summarises all the components of the system. Few additional monitors will be available for temporary installation during the installation of the interferometer.

5200.4.4. Cabling:

We foresee 4 cameras in the mode cleaner, 33 in the central building and 6 in each end mirror building. Taking into account few cables to send back images (1 to the mode cleaner building and 6 to the central building) and adding some spare cables we need the following cables starting from the control room (in fact from the computer room):

- 3 optical fibers to each end mirror building (4 cameras are grouped by a quadravision)
- 8 coaxial cables (150m of KX6) to the mode cleaner building
- 40 coaxial cables (150m of KX6) to the central building

The optical fibers are from the standard cabling. Dedicated patch panels for the coaxial cables will be installed in the data acquisition room, computer room and mode cleaner room near the network patch panels

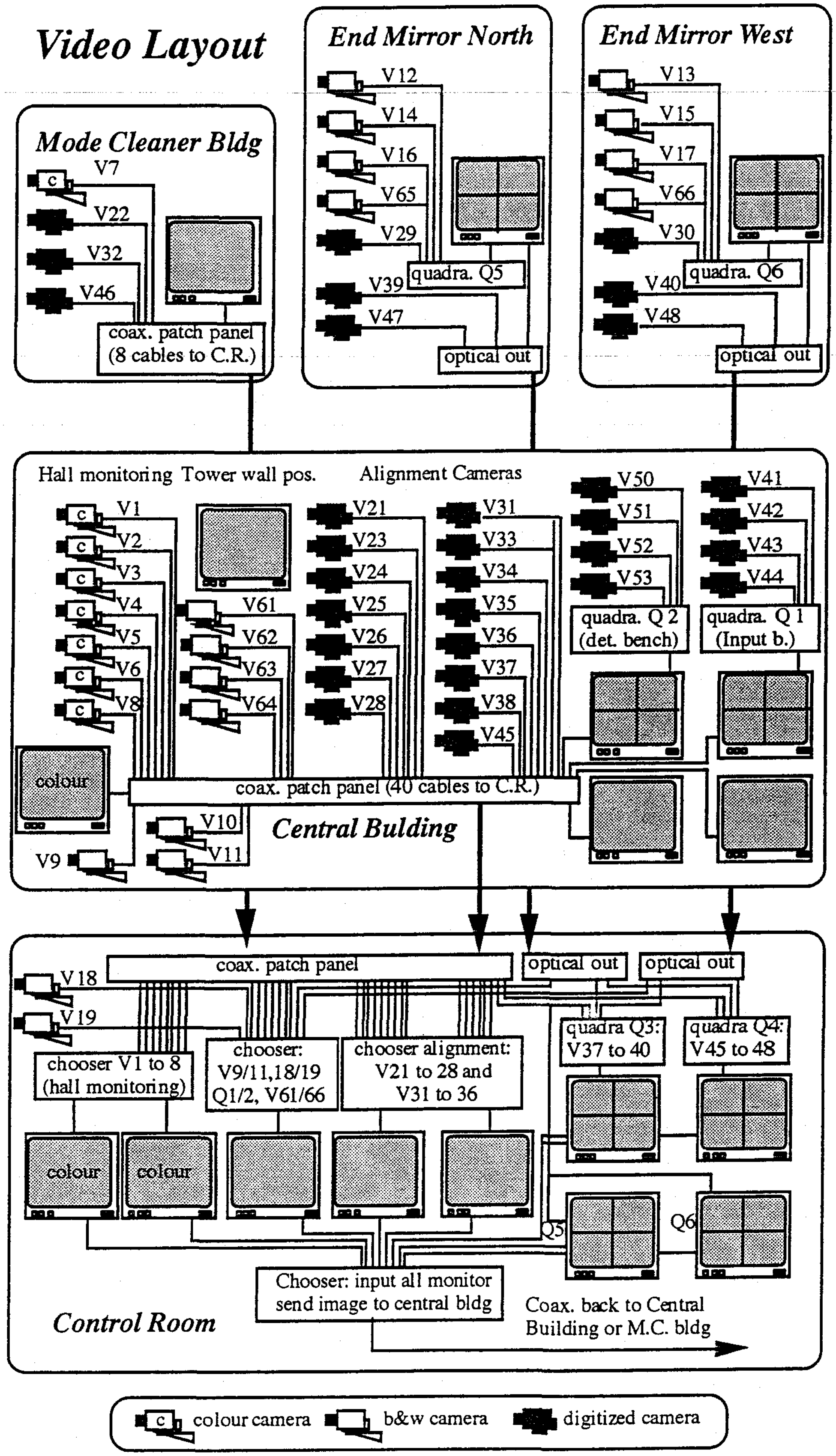

Figure 5200.4. Schematic view of the VIRGO Video system. The digital part of the system (VME interfaces, processors) is not presented.

5200.5. 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 spares):

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.

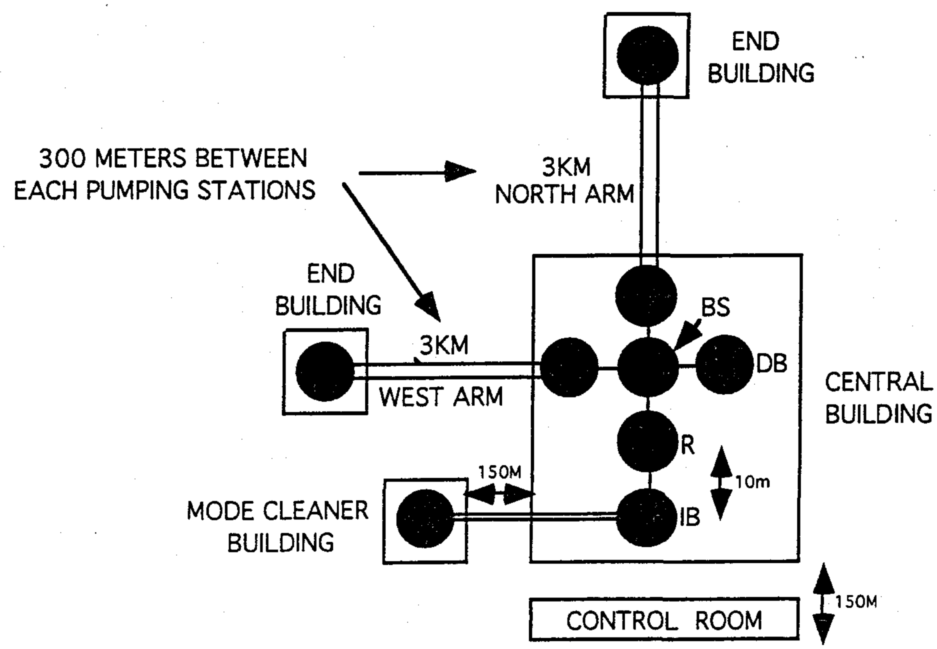

figure 5300.1 VIRGO Buildings

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.

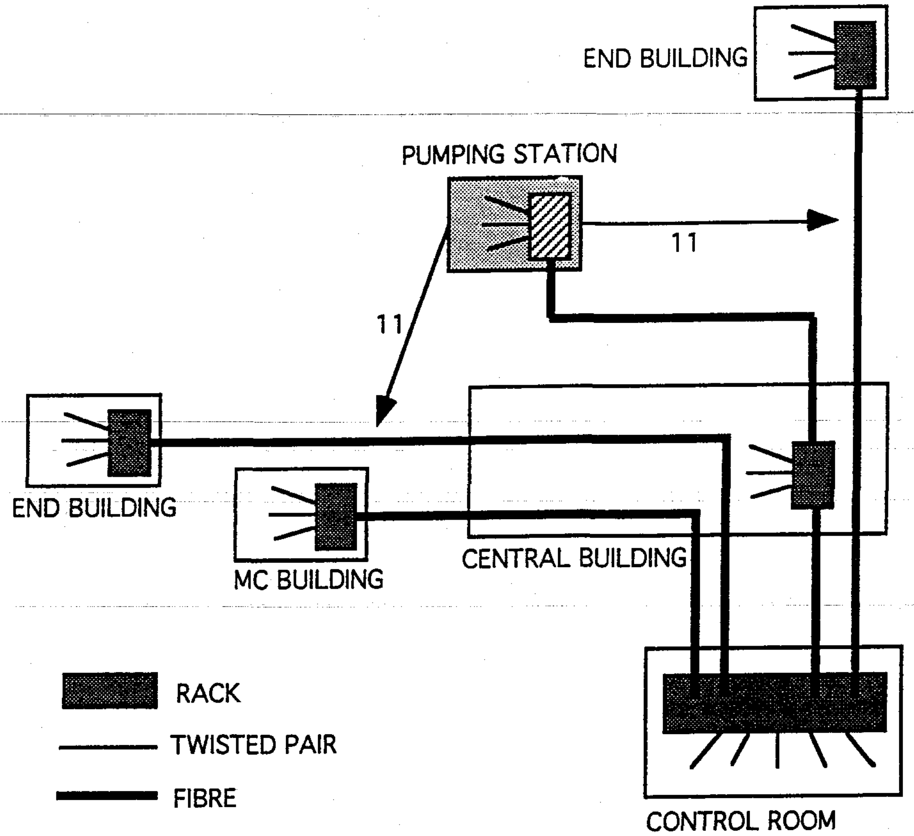

Figure 5300.2 Main fibers

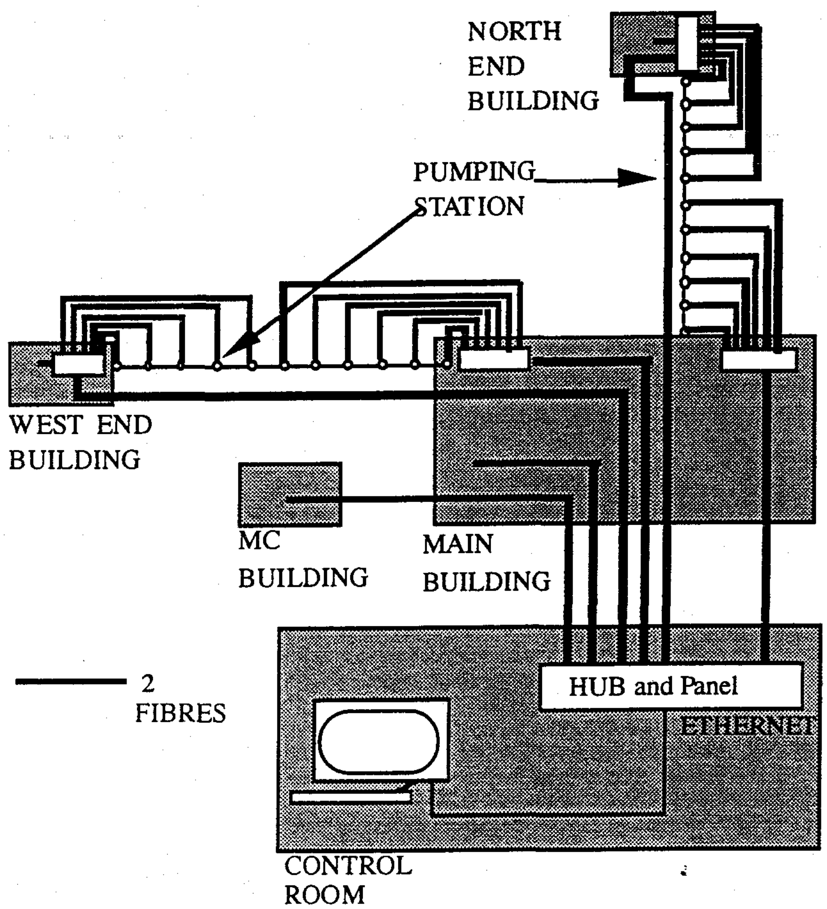

Figure 5300.3 The slow monitoring network.

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

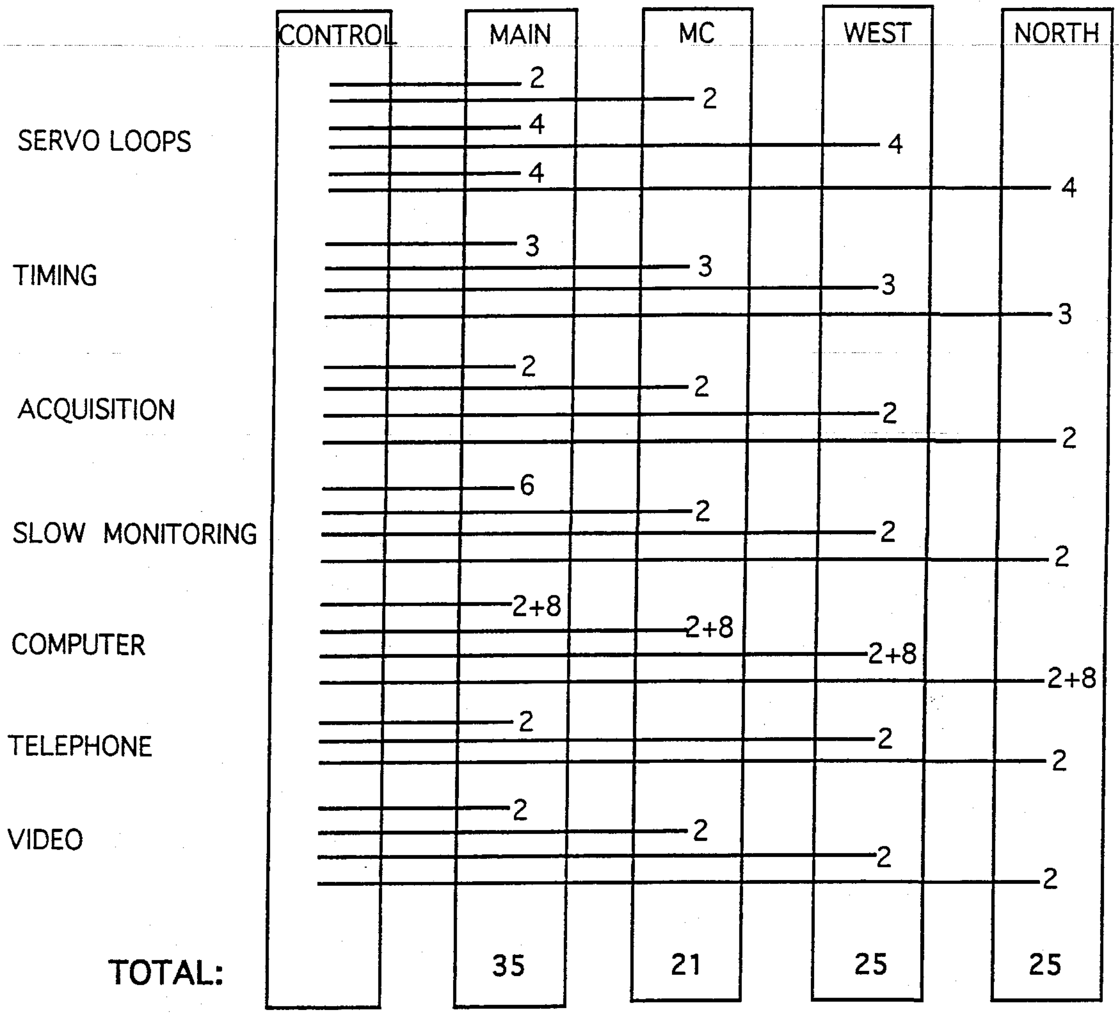

Figure 5300.4 Optical Fibers

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.

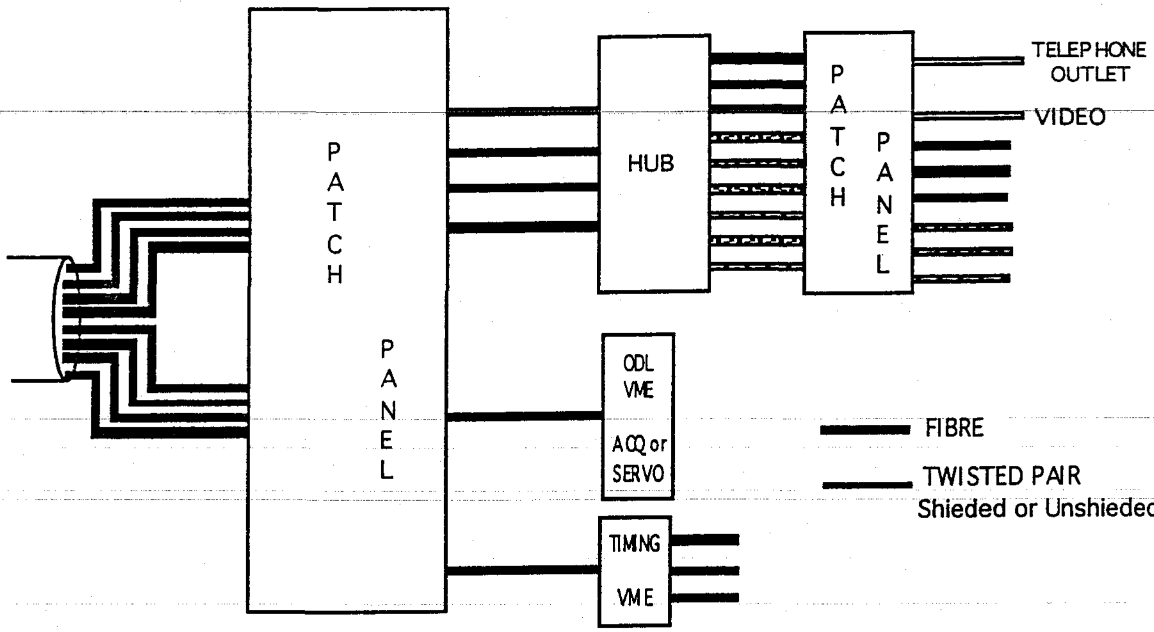

Figure 5300.5

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 sends 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.

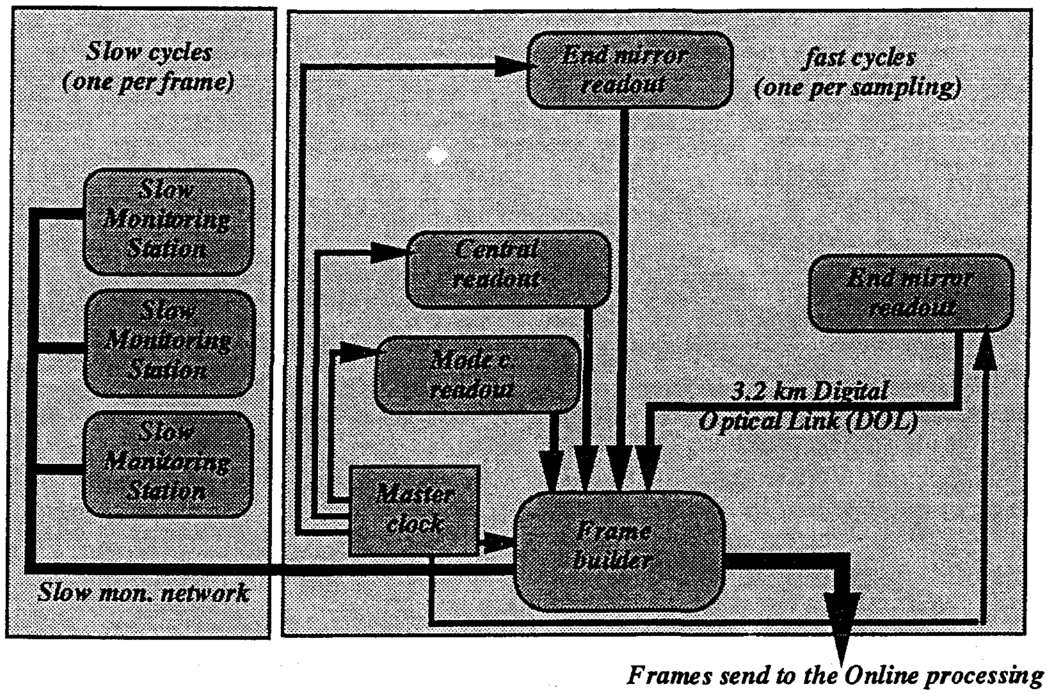

figure 5400.1 The distributed data acquisition system

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 information provided by the different sensors, 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 twice more data. The ADC readout frequency could be a fraction of the dark fringe sampling frequency for the signals which do not need fast 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.

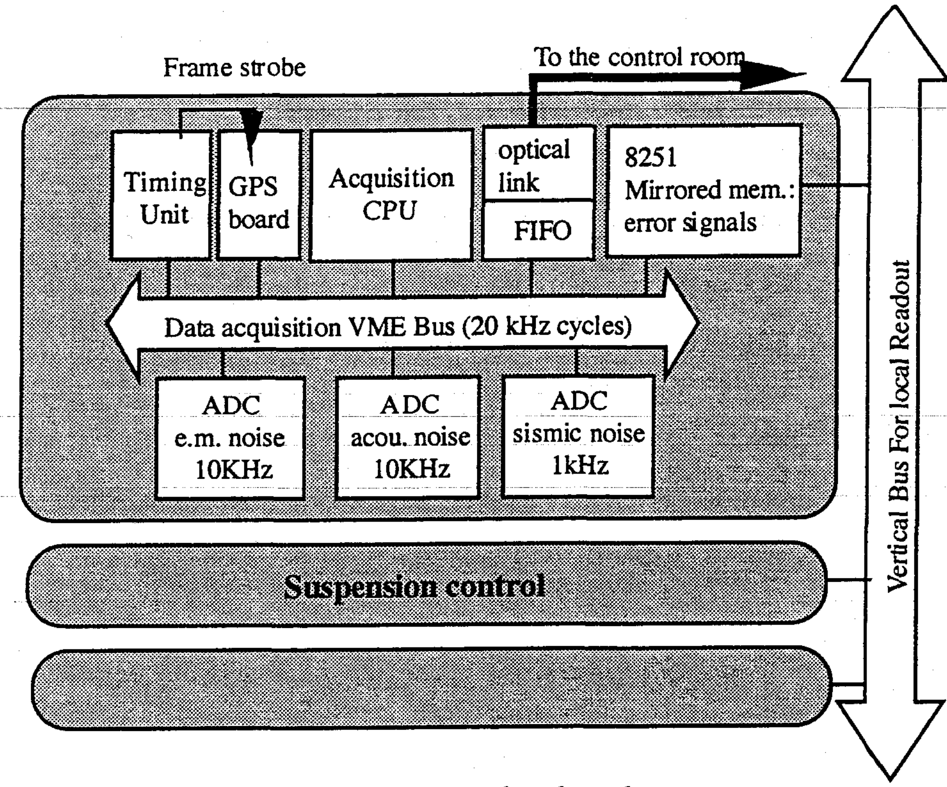

figure 5400.2 A local readout system

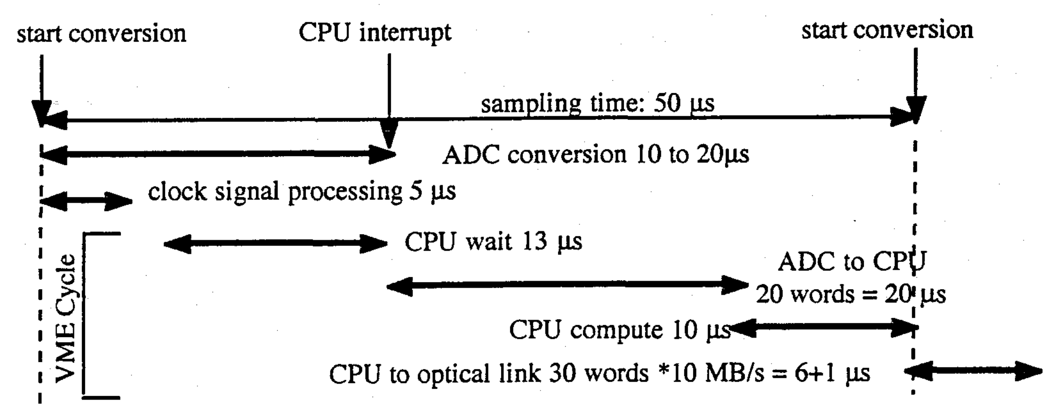

figure 5400.3 The timing sequence of a local readout cycle

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.

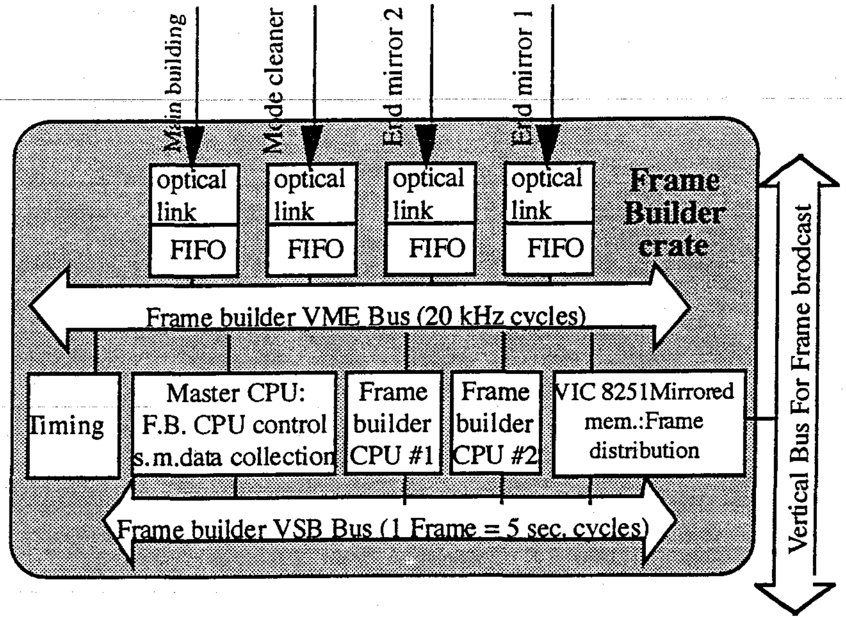

figure 5400.4 Block diagram for the Frame Builder.

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.

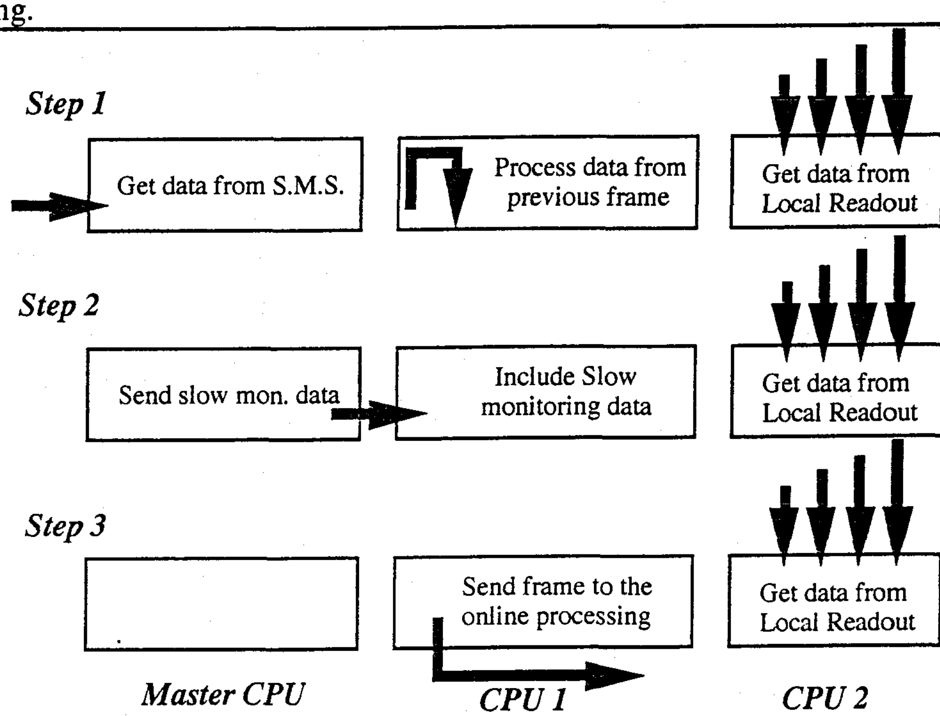

Figure 5400.5 The three steps performed by the frame builder for each frame. After these steps CPU 1 and CPU 2 switch their tasks.

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 channels	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/s. 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. A few MBytes/s. 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 explains our specification of 10MBytes/sec.

5400.3. The frame structure.

A frame is a unit of information containing all the information necessary for the understanding of the interferometer behavior over a finite time interval which integrates several samplings. 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.

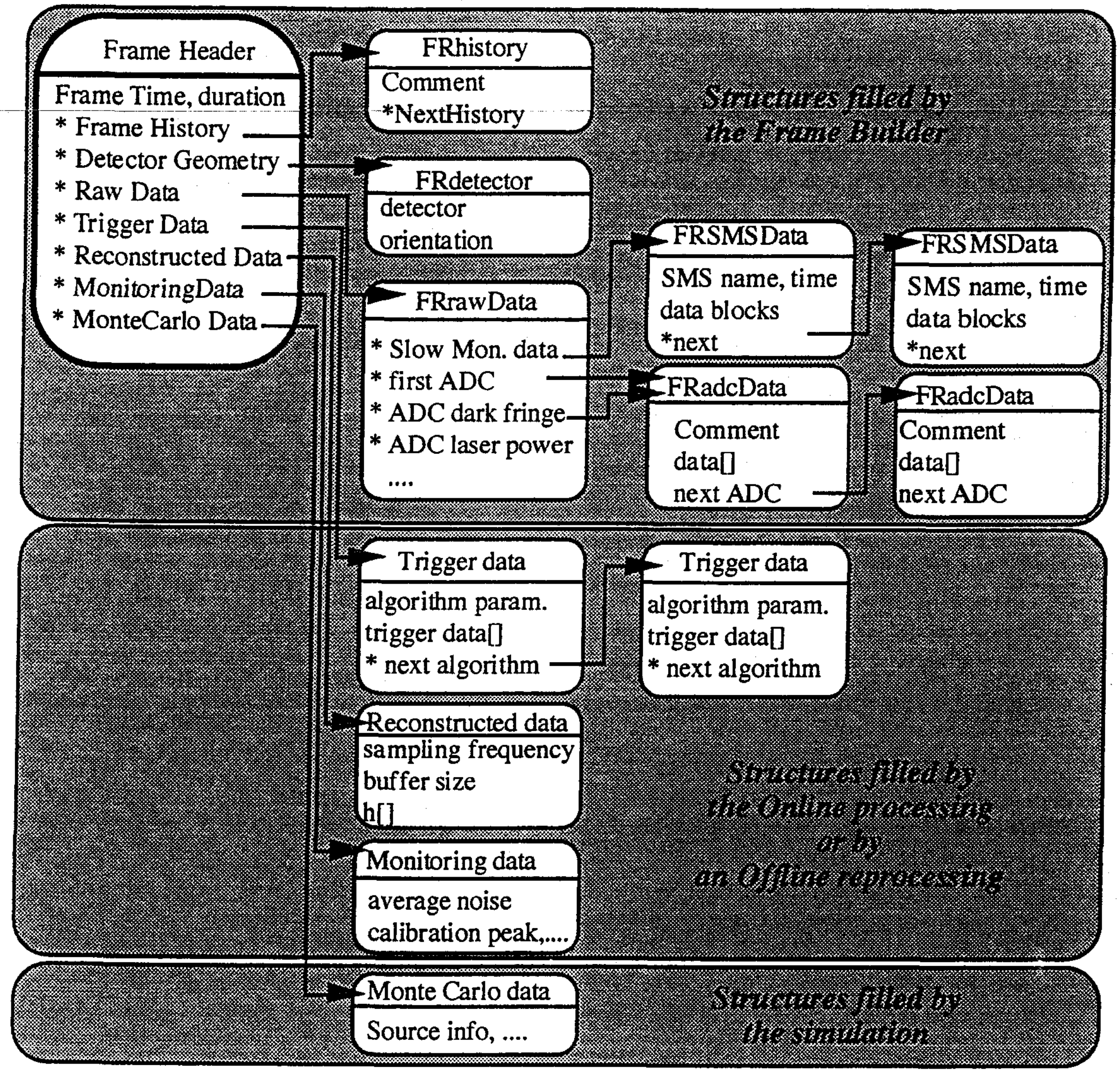

Figure 5400.6 The frame structure

5400.4. The Data Display.

This is a tool 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 offline to play with simulated data or reconstructed data.

5400.5. Online processing

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

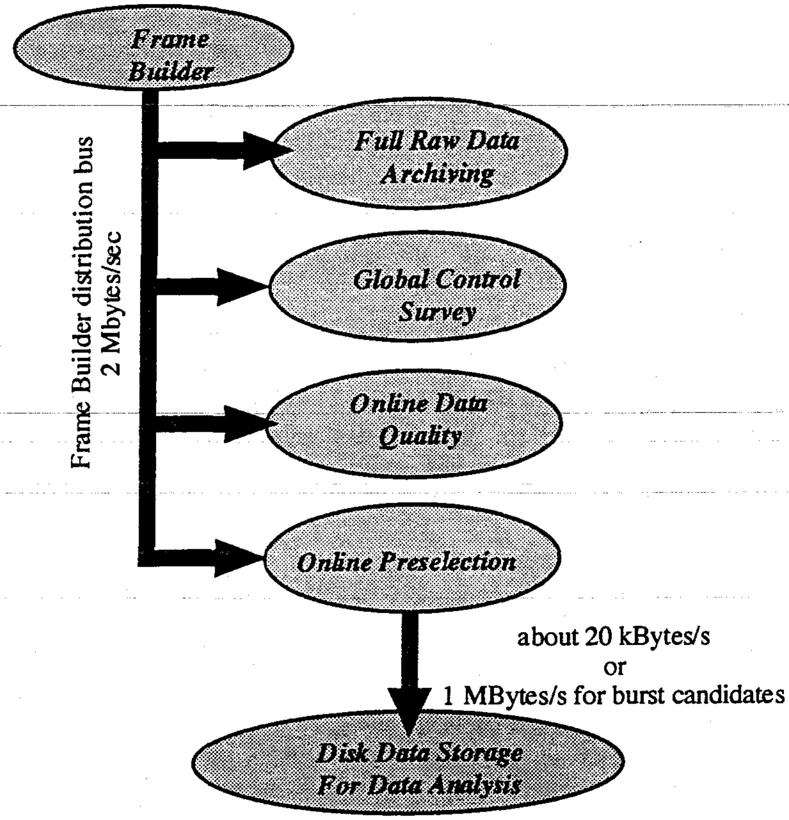

figure 5400.7 The principle of the online processing

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 trend in the various servo-loops and, more generally, to react on faulty behaviors affecting 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 online 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 information 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 us remember the typical 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 the 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 information 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 sent 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. Buildings Control System

The Building Control System controls the appliances installed in the buildings (i.e. power supply systems, air conditioning systems, etc.) and the main actuators in the buildings. At the same time, it acquires all the variables describing the status of the buildings (i.e. temperature, pressure, humidity, etc.),

A distributed I/O control protocol system (ETN system - Enhanced Tecnint Network) is used both for acquisition and control. This system is interfaced to a standard VME environment. A PC system option permits the control of those appliances for which the control software is directly provided by the factories for MS-DOS operating system.

A Building Control Station is foreseen for each building. Each station (Building Control Server) consists of a dedicated VME crate, controlled by a local CPU, running the operating system LynxOS and linking each server to the Slow Monitoring Network (or a PC with both operating systems MS-DOS and LynxOS). The architecture of each server is open for making it easy to change its structure on the basis of the needs during the construction phase and the operational phase (Fig.5500.1.1).

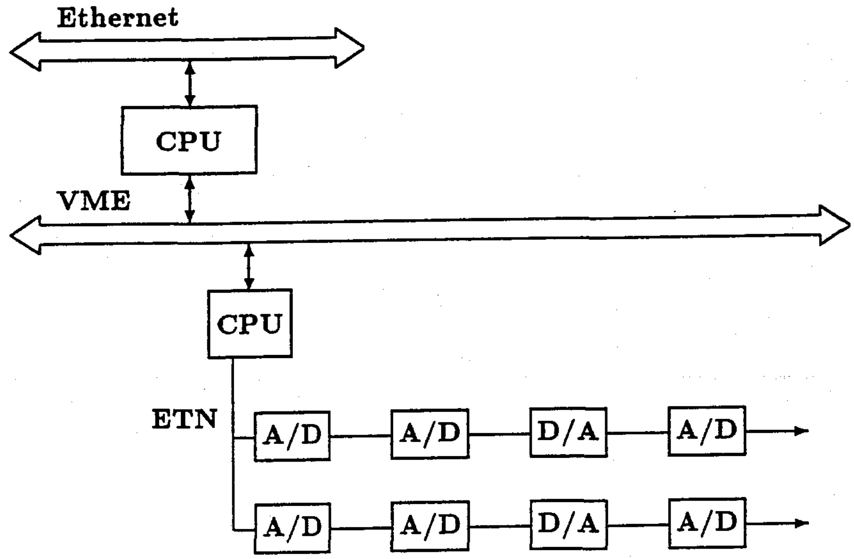

Figure 5500.1.1 Building Control Station Hardware Structure

The architecture of the Buildings Control System is shown in Fig.5500.1.2. A user interface running on one of the control room workstations provides the possibility of managing the whole Buildings Control Network, setting the configuration of each server and of the probes (stored in the On Line Database System), checking the errors (stored in the Error Logger System), sending control commands, acquiring all the variables describing the status of the buildings and, if necessary, displaying their history.

5500.2 Environment Monitoring System

The Environment Monitoring System monitors all the environment variables, which are not directly related to the status of interferometer, but necessary for the correlation analysis with the interferometer output signals.

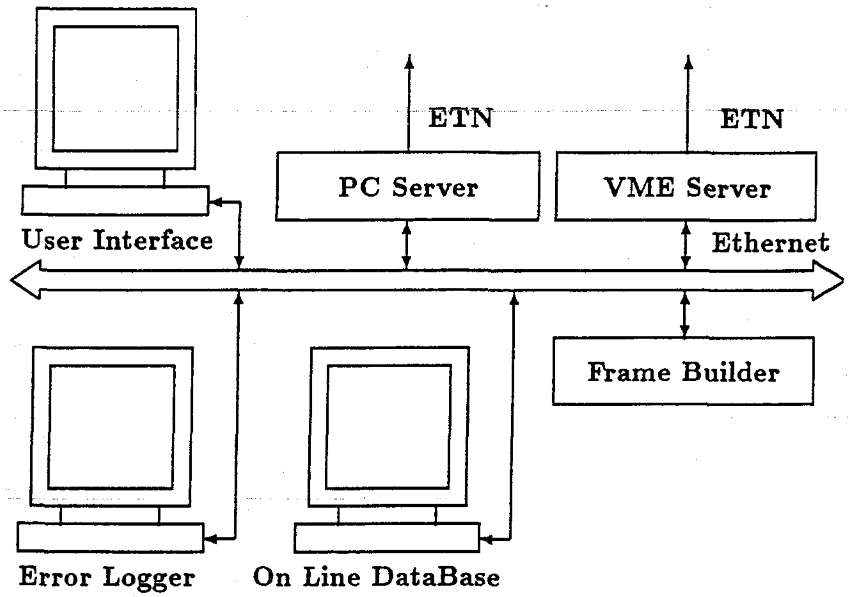

Figure 5500.1.2 Architecture of the Buildings Control System

A first study on the type and the number of variables which need to be monitored has led to the Table 5500.2.I. More precise indications on the actual number and final position of the probes is going to be obtained by direct measurements on site with the interferometer working. For this reason each Environment Monitoring Station (one for each building) is characterized by an open architecture which makes it easy to add other probes, if necessary, or simply to change the type of probe used.

Parameter	CB	MCB	NB	WB	NT	WT	Freq	Timing
Electromagnetic Noise	8	2	2	2	-	-	20 kHz	yes
Acoustic Noise	8	2	2	2	-	-	20 kHz	yes
Power Supply Noise	5	2	2	2	-	-	20 kHz	yes
Seismic Noise (trid.)	8	2	3	3	-	-	1 kHz	yes
Cosmic Ray Noise	1	1	1	1	-	-	impuls	yes
Env. Temperature	8	2	3	3	-	-	1 Hz	no
Env. Pressure	8	2	3	3	-	-	1 Hz	no
Env. Humidity	8	2	3	3	-	-	1 Hz	no
Wind Speed	1	-	1	1	-	-	1 Hz	no
Wind Direction	1	-	1	1	-	-	1 Hz	no
Rain	1	-	1	1	-	-	1 Hz	no

CB = Central Building MCB = Mode Cleaner Building
NB = Nord End Building WB = West End Building
NT = Nord Tunnel WT = West Tunnel

Table 5500.2.1 Environment Monitoring Parameters Table

Due to the different influence of the environement variables on the sensitivity curve of the interferometer, the sampling frequency is not the same for all. In fact, some of them need to be sampled at the same frequency of the output signal of the interferometer (20 kHz) with precise timing (i.e. electromagnetic noise and acoustic noise) while for others it is sufficient a lower frequency sampling rate with no precise timing definition. On this basis it is possible to divide the environment monitoring acquisition in two different logical channels:

• **ch.1** - The acquisition of fast variables which need precise timing is integrated within a fast VME data acquisition system (local readout), which makes it easy to handle the data and to transfer them to the Frame Builder. At the same time these variables are acquired for checks and for spectral analysis by the Environment Monitoring Stations, which record and send their main characteristics to the Frame Builder.

• **ch.2** - The acquisition of slow variables is made by the Environment Monitoring Stations (Environment Monitoring Servers), which send them directly to the Frame Builder by means of the Slow Monitoring Network (Ethernet). Each station consists of a dedicated VME crate, controlled by a local CPU, running the operating system LynxOS and linking each server to the Slow Monitoring Network, in which ADC boards and RS232 boards, necessary for those probes which internally sample the variables and transfer the data via a standard protocol RS232, are housed (see Fig.5500.2.1).

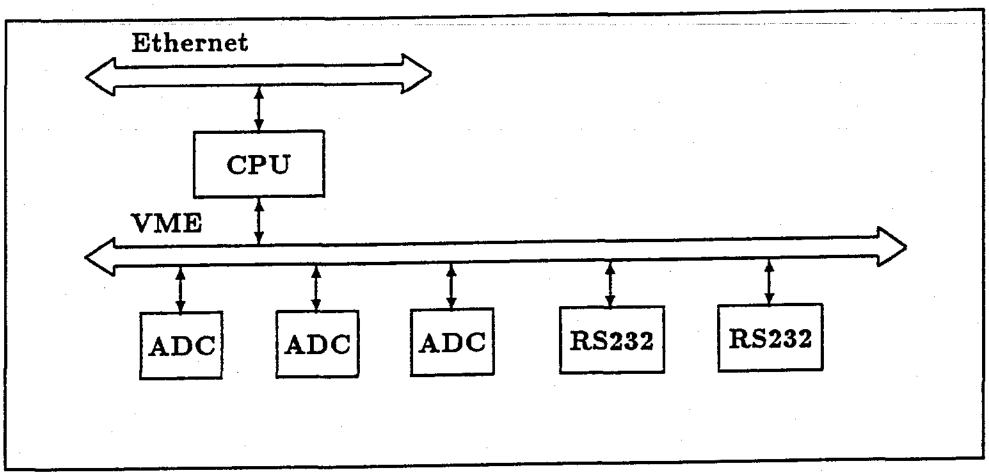

Figure 5500.2.1 Environment Monitoring Station Hardware Structure

The architecture of the Environment Monitoring System is shown in Fig.5500.2.2. A user interface running on one of the control room workstations provides the possibility of managing the whole Environment Monitoring Network, setting the configuration of each server, of all the fast and slow acquisition boards and of the probes (stored in the On Line Database System), and checking the errors (stored in the Error Logger System). Such user interface has one graphical page for each type of variable. The history of each environment monitoring parameter can be displayed on this workstation using the Historical Monitoring software in connection with the Data Distribution System.

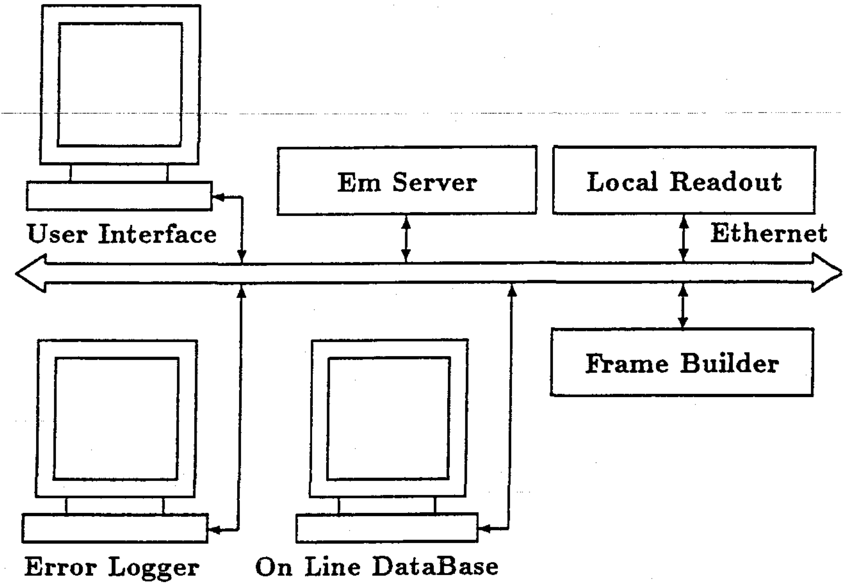

Figure 5500.2.2 Architecture of the Environment Monitoring System

5500.3 Raw Data Archiving System

The Raw Data Archiving System archives all the frames produced by the Frame Builder, making it possible any eventual full retrieval and reprocessing of the original data. For safety reasons, it makes two copies of the archive, which are stored in two different locations.

The major technical specification for the Raw Data Archive is given by the need of storing a maximum continuous data flow equal to 10 MByte/sec, although an expected average data flow equal to 2 MByte/sec is foreseen (see 5400.2).

Such large amount of data have necessarily to be mantained off-line. Although an off-line archive can use both optical disks and magnetic tapes, actually the only economical affordable solution is that of using magnetic tapes like standard DAT tapes, which have a storage capacity of 4 GByte. Therefore, assuming a continuous data flow equal to 2 MBtye/sec about 16000 DAT tapes/year/copy are necessary.

Although the data flow specification can be easily satisfied by using a VIC bus for data transfer between the Frame Builder System and the Raw Data Archiving System, the data storage specification requires the implementation of a two stage storage procedure, because the maximum writing speed on DAT tapes is only about 500 kByte/sec. A modular solution to this problem is given by a parallel staging of the data on disks (which can sustain a read/write data flow larger than 10 MByte/sec and can have a storage capacity equal to that of the DAT tapes) and then copying them on DAT tapes. This solution makes it easy also to reconfigure the system on the basis of the actual data flow.

Hence, assuming a data flow equal to 2 MByte/sec, the Raw Data Archiving System can be composed of a dedicated VME crate controlled by a master CPU, running the operating system LynxOS and linking the Raw Data Archiving System to the network (Ethernet). In this crate a VIC bus and 5 CPUs, each provided with a 4 GByte disk and a DAT unit with autoloader, are housed (Fig.5500.3.1). This configuration has also the advantage of giving enough time to rewind the full DAT tapes and to load another one on the same unit.

The whole procedure is shown in the scheme of Fig.5500.3.2, that shows the temporal course of the storage procedure. A full cycle lasts 10 ksec (about 3 hours), that is new data are written on the same disk after 10 ksec. This solution permits to have higher data flows (up to 10 MByte/sec) for short times.

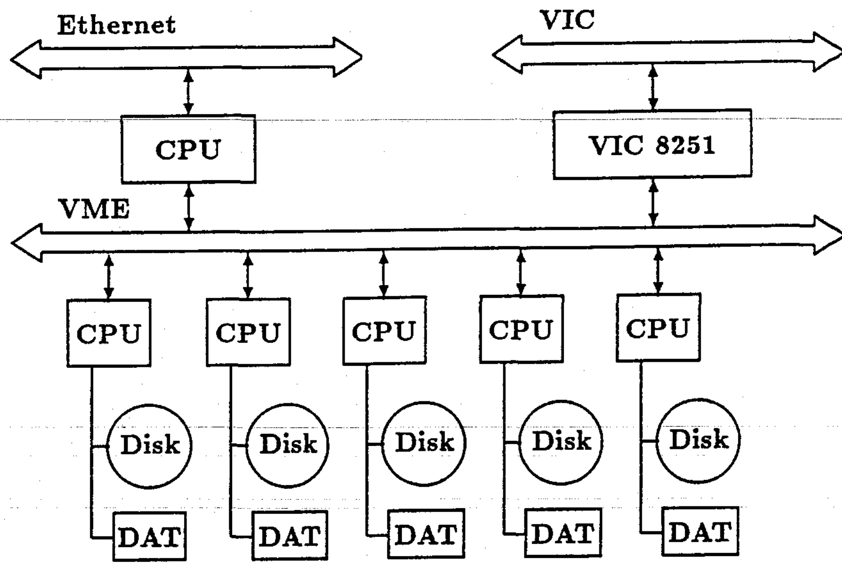

Figure 5500.3.1 Data Archiving System Hardware Structure

The architecture of the Raw Data Archiving System is shown in Fig.5500.3.3. A user interface running on one of the control room workstations makes it possible to manage the Data Archiving System, setting its configuration (stored in the On Line Database System), checking and changing its status, checking the errors (stored in the Error Logger System).

The second copy of the archive is obtained by simply duplicating the system. This solution has the great advantage that the two systems work in shadow mode, that is data are not lost even if one of the two systems is in fault.

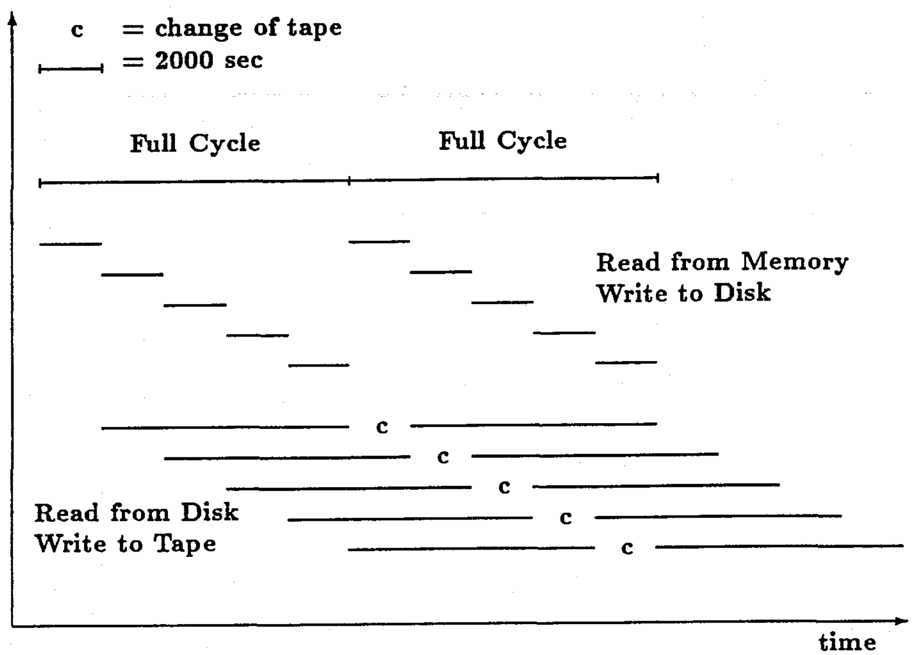

Figure 5500.3.2 Time Course of the Storage Procedure

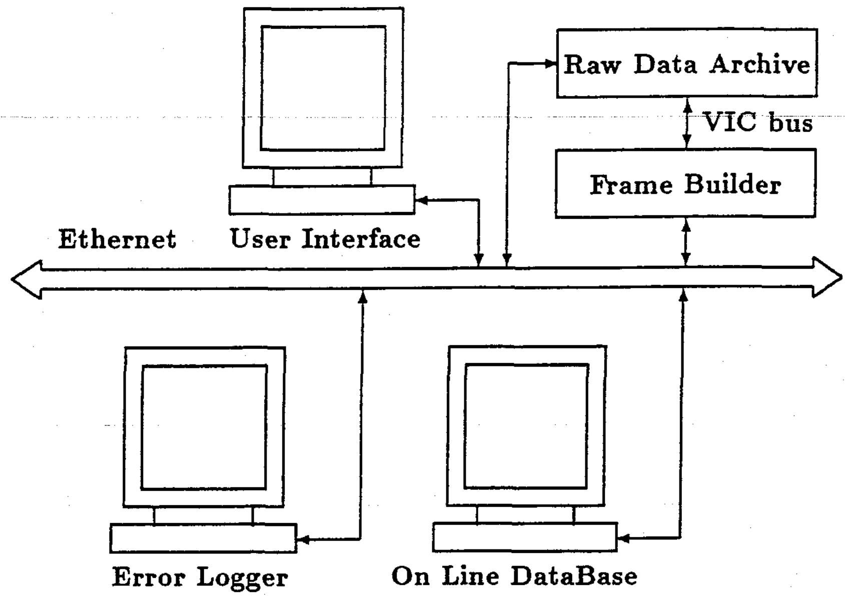

Figure 5500.3.3 Architecture of the Data Archiving System

5500.4 Data Distribution

The Data Distribution System collects the data produced by the On Line Processing System and stores them on disks (on-line data distribution) and on DAT tapes (off-line data distribution - DST tapes, see 5400.5.3). These data contain all the useful information for the off-line data analysis, such as the reconstructed h and the frames selected by the parallel real-time data analysis algorithms running on the On Line Processing System.

The data handled by the Data Distribution System are of the following types:

• reconstructed h values (with acquisition time and a quality coefficient) (maximum continuos data flow equal to 100 kByte/sec at 20 kHz sampling rate).

• slow monitoring records for monitoring the interferometer environment (maximum continuos data flow equal to 5 kByte/sec).

• frames selected by the real-time data analysis algorithms, which contain all the raw data required to perform a full analysis of the signal candidates (maximum data flow equal to 2 MByte/sec).

• raw data, retrieved from the archive DAT tapes, to be reprocessed, or data retrieved from DST tapes.

The data storage on DST tapes is divided into three different channels, which contain the following information:

• **ch.1** - all the frames with reconstructed h data and slow monitoring records at the sampling rate (20 kHz) for performing a full data analysis for all the sources on the whole band of the interferometer. The storage of these data requires a maximum amount of 800 DAT tapes/year (corresponding to a data flow equal to 3200 GByte/year).

• **ch.2** - all the frames with slow monitoring records and reconstructed h data resampled at lower frequency (2 KHz) for performing a specific research on continuos sources (i.e. pulsar search). The storage of these data requires a maximum amount of 80 DAT tapes/year (corresponding to a data flow equal to 320 GByte/year).

• **ch.3** - frames selected by the parallel real-time data analysis algorithms running on the On Line Processing System for performing a specific research on bursts or coalescing binaries, which contain all the necessary raw data. The specification on the maximum data flow is about 2 MByte/sec, but these data have not be stored in continuos mode. Assuming 10 days of these data, then the quantity of data to store is equal to about 1750 GByte (450 DAT tapes/year).

Requested copies will be dispatched to all the groups of the collaboration for performing an off-line data analysis. The lists of the contents of the data stored in these three channels are kept always on line on the Data Distribution System.

Only part of the information received by the On Line Processing System is stored on disks, because of the limited storage capacity of such devices. The storage time of each channel will be decided according to the needs of data analysis. We foresee to have 500 GByte on line. Therefore, the Data Distribution System will be made of at least 80 disks (9 GByte each).

All this information is organized in the following way. The frame directory and the slow monitoring variables are stored as in a classic database system and for this a commercial database is used for a full compatibility with other scientific databases, while the frames selected by the On Line Processing System are stored according to a tree structure, as shown in Fig.5500.4.1.

```
          EXP Name
             |
      _______|________________________
      |                |              |
     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 5500.4.1 Structure of the Stored Frames

At the same time the Data Distribution System, acting as server, distributes all the requested data to the authorized Virgo user interfaces. To avoid any interference between the data collection from the On Line Processing and the Data Distribution, the system uses separate standard network lines.

The Data Distribution System must have the necessary CPU power for handling the required number of disks, the storage on tapes and allow the users access to the on-line data. This specifications can be obtained by using an open architecture multiprocessor system, with multiple buses for I/O and multiprotocol (Ethernet, FDDI, VME, PCI, etc.). The architecture we can envisage now is shown in Fig.5500.4.2., that is a computer system which can accomodate multiple CPUs and memory boards, PCI peripherals buses for the I/O which supports several SCSI buses to which we can attach disks and tapes (with loaders) and a VME bus for connection with the On Line Procesing System (via VIC bus) and a DAT reading unit. This structure has the advantage that can be easily expanded according to future needs. In fact, further disks may be added over the time and likely also other CPUs and memory.

The architecture of the Data Distribution System is shown in Fig.5500.4.3. A main user interface running on one of the control room workstations provides the possibility of managing the Data distribution network, setting the configuration (stored in the On Line Database System), checking the errors (stored in the Error Logger System), accessing the stored data for changing their structure, for deleting and moving files and for all the relevant operations of management of the system itself. Standard user interfaces access the stored data in read only mode. In particular, the history of each stored quantity can be displayed on these workstations using the Historical Monitoring software.

Finally, the Data Distribution System must give the possibility of retrieving and eventually reprocessing the stored data (raw data, from the archive DAT tapes, or data retrieved from DST tapes). The operation of loading the requested DAT cassettes to the Data Distribution System reading unit can be performed automatically using a robotic system (at least for DST tapes).

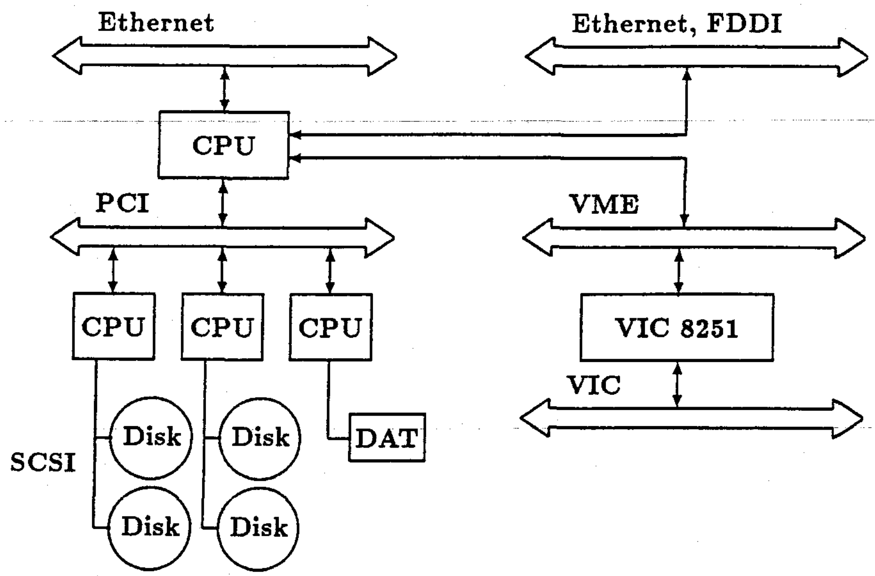

Figure 5500.4.2 Data Distribution System Hardware Structure

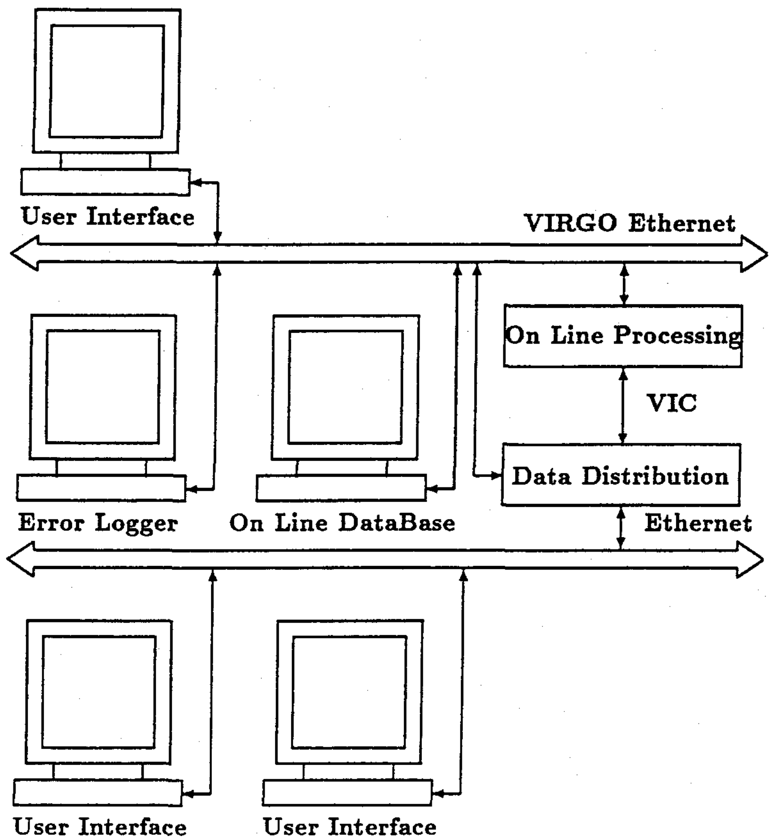

Figure 5500.4.3 Architecture of the Data Distribution System

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 Data Display shows the data archived according to a frame structure.

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 information about SIESTA could be found in the Siesta directory with documentation and examples