THE COMPUTER CONTROL FOR THE RF SYSTEM OF THE MILAN K800 CYCLOTRON

F. Zibra, A. Borini, A. Bosotti, M. Di Giacomo, D. Giove, C. Pagani, G. Varisco.
Istituto Nazionale di Fisica Nucleare and Universita' di Milano - L.A.S.A.
Via F.lli Cervi 201, 20090 Segrate (Milano), Italy.

ABSTRACT

The operation and setting of the RF system for the Milan Superconducting Cyclotron is controlled by a computer driven electronics[1,2].

The control architecture consists of a set of microcontroller based boards and of a master unit based on an Intel 80286 microprocessor. The boards may exchange data and commands at a speed of nearly 40 kbytes/sec.

The electronics and the applied software allow to drive each unit independently, while the master unit guarantees the coordination of the whole RF system and connects it to the cyclotron computer control, via the Ethernet network.

A particular care has been devoted to the development of the software architecture due to the complexity of the system.

In this paper the control architecture design is presented together with a discussion of the experience gained during its use.

RF CONSOLE

The RF Computer Control is basically composed by two main blocks, the RF Control Station (RFCS) and the Computer Control Cabinet (CCC).

All the distributed control modules are contained in the Computer Control Cabinet and are connected to the RF Control Station via Bitbus[5].

The RF Console is a subset of the main console[2,3,4] of the accelerator, being connected to the last via an Ethernet network. A movable console has been developed to be connected locally, close to the electronics cabinets, which have to be controlled. This allows to eventually set and drive the RF system, independently from the other components of the cyclotron, but still maintaining the same operator interface.

The RFCS consists of a minicomputer (INTEL 310 AP). Interface boards toward the Ethernet network (iSBC 186/51) and Bitbus (iSBX 344) are fitted inside the 310 AP. The main CPU of the minicomputer is a 8 MHz 80286, with 1 MByte of RAM. A RS-232 port has been dedicated to the monitoring of real time programs.

Operator inputs are handled via a touch panel, connected to the other RS-232 port, and four interaction modules, based on the 8044 microcontroller and connected via Bitbus.

Particularly, the operator chooses the RF parameter to be controlled on the touch panel (through a guided menu), handling it with either the knobs or the keyboard of the interaction modules. Each module display shows the actual parameter value. The RFCS architecture is shown in figure 1.

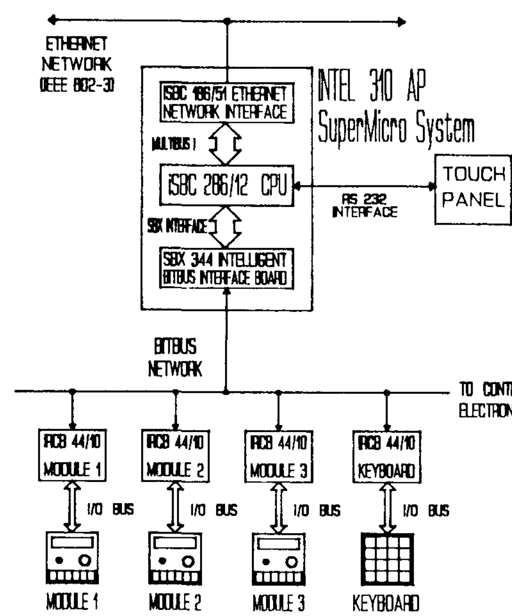

Fig. 1. Architecture of the RF Console.

The distributed microcontroller boards(iRCB 44/10 and iRCB 44/20) are contained in the CCC[1] and they are interfaced to the RFCS via a iSBX-344 board, which acts as the Bitbus network master. An especially designed interface bus (RF-IB) is used to link the microcontroller boards to the electronics racks to increase the control capability of each unit.

Three different topologies are used to connect the microcontrollers to the racks, as specified in the following.

Direct Connection

Direct connection, which is the simplest, is used when the subsystems, which have to be controlled, are common to the three cavities[1], like in the case of:

- BBC Power Amplifiers;
- Delay Lines;
- Control Interlocks.

The direct connection scheme is shown in Fig. 2, where RF-IB is the standard interconnection bus between the iRCB 44/10 and the control electronics subsystems.

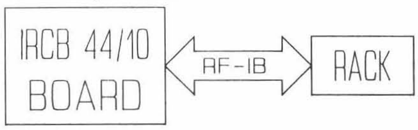

Fig. 2 - Direct Connection scheme.

Served Connection

A server (RF-IB Server) is used when three separate racks, with identical functions (one for each cavity), must be controlled. This is the case[1] of:

- Phase and Amplitude Loops;
- Fine Tuning System (Trimming Capacitor);
- Sliding Short and Coupler Moving Systems;
- Alarm Boards;
- Turn On Electronics.

The scheme with server is shown in Fig. 3.

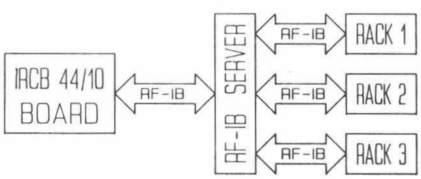

Fig. 3 - Served Connection scheme.

Hybrid Connection

This last scheme has been used when the microcontroller performs the control of an electronics rack (RF Mux) together with data acquisitions from instruments, which have a standard GP-IB interface like:

- ZPV Vector Analyzer;
- RF Synthesizer;
- RF URV Voltmeter[1].

In this case an interface board toward the IEEE 488 bus is mounted on the microcomputer (using an internal expansion bus) and the IEEE 488 bus is used together with the RF-IB. This kind of connection is shown in figure 4.

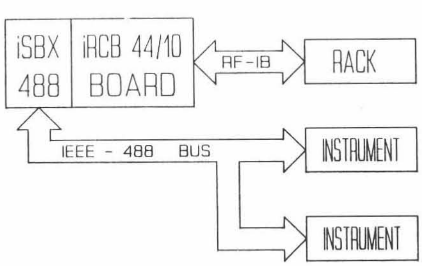

Fig. 4 - Hybrid Connection scheme.

BITBUS LEVEL CONTROL SOFTWARE

The distributed microcontroller boards, i.e. the computer control Bitbus level, are the devices that interacts with the RF control electronics.

The RF Bitbus network is composed by:

- one iSBX 344 master board;
- eight standard iRCB 44/10 (24 digital I/O points);
- two iRCB 44/20 (16 single ended analog channels);
- one iRCB 44/10, with a GP-IB interface;
- three custom design Bitbus boards (ABS), with a 16-bit DAC (Hybrid System DAC9377) and a 16-bit ADC (HS9516).

Since a great care must be taken to optimize the interactions between hardware and software, the Bitbus microcontroller firmware design has followed step by step the control electronics development[1].

Particularly the main software requirements may be so summarized:

- the structure of the application programs on each board must be the same, regardless the specific functions to be performed;
- extensive use of structured programming techniques allows easy modifications of single procedures, without affecting other parts;
- microcontroller communication frame assuring an excellent throughput between the nodes, regardless from the numbers of the used microcontrollers;
- fast response time (not more than 20 ms for the response to an external command).

The communication protocol actually implemented is based on a master (iSBX 344) which overviews all the communications between the other microcontroller boards and the 310 AP (in both directions). Particularly, the master has a task (poller) which cyclically controls the status of the slaves and allows the message ratings between them.

The software structure is substantially identical for all the microcontrollers and its flow chart scheme is presented in Fig. 5.

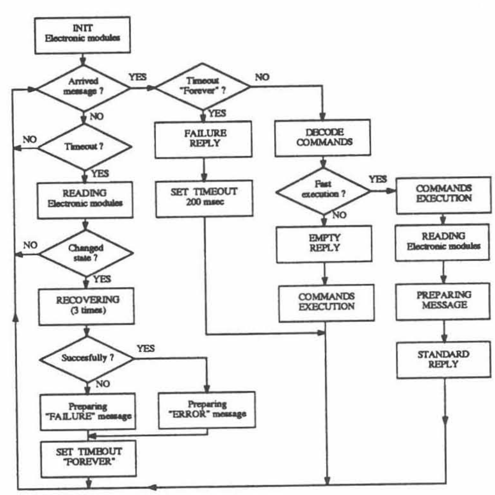

Fig. 5 - Software structure flow chart.

A subsystem controller makes the proper pre-settings when starting a power operation and then waits for an order. The controller remains in this idle state for a fixed time (actually 200 ms) or until the reception of an order.

After this time-out period has expired, if no orders from the master have arrived, the controller starts an overview on the device under control, to verify its status, according to the previous settings, makes corrections in case of fault and then begins again the idle state.

In case of device fault the controller tries three times to recover the right settings. After that, a warning message is sent to the operator if the recovery has been successful, otherwise the controller sends a failure message, indicating to the operator which subsystem failed and the kind of fault into its message. In this case, the controller waits for the master message before starting again its work.

Moreover, the controller decodes the commands received from the master and establishes if the execution will be fast or not. Particularly, if the execution is fast, it executes the command and reads the electronics subsystem status before answering to the master with a reply. Otherwise the controller sends back an empty reply and then executes the command. The master itself will later perform the reading of the subsystem status to update the settings after the commands execution. The controller restarts its overview operation after the order has been carried out.

RF CONSOLE OPERATING SOFTWARE

The RF Computer Control can operate either from the accelerator main console or from a local console, for stand alone and maintenance use.

The development of a local console has been suggested by the fact that the complexity of the RF system may[6,7] requires a lot of time to get a detailed knowledge of the best settings for each beam to be accelerated. More precisely we can say that the operator uses the local console to start up the RF cavities, while, for cyclotron operation, the RF system is controlled by the main console.

The RF console operating software can be thought as constituted by three concentric levels:

- the iRMX 86 operating system;
- the operator interaction I/O devices and the interface software for the Ethernet and Bitbus networks;
- the control electronics applied software.

The operating software conceptual scheme is presented in Fig. 6.

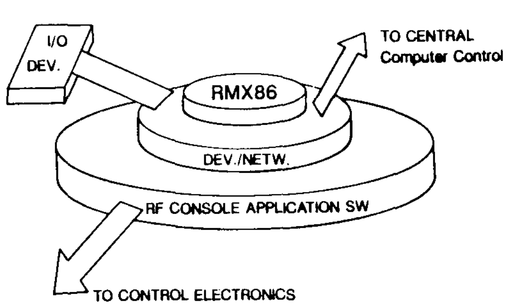

Fig. 6 - RF console operating software conceptual scheme.

CONCLUSIONS

The Bitbus level control software has been developed together with the RF system control electronics. The operating iRMX 86 high level software is now under development.

A program that simulates the RF console operating software has been implemented on a IBM personal computer to completely test the control electronics chains, before the completion of the iRMX 86 software, which requires a longer design time. The simulation software has the same logic structure as the operating one and it is written in the same language (PL/M-86). The main differences between simulation and real software are summarized in Fig. 7.

RF CONSOLE CONTROL SOFTWARE	SIMULATION SOFTWARE
REALTIME MULTITASKING MORE POWERFUL MORE COMPLEX WRITTEN IN PL/M 86	NO REALTIME NO MULTITASKING LESS POWERFUL LESS COMPLEX WRITTEN IN PL/M 86
IRMX 86 REALTIME EXECUTIVE	UDI INTEL UNIVERSAL DEVEL. INTERFACE DOS
INTEL BITBUS SOFTWARE DRIVER	INTEL SUPPLIED BITBUS SOFTWARE DRIVER
ISBX 344 INTELLIGENT BITBUS INTERFACE BOARD	IPCX 344 INTELLIGENT BITBUS INTERFACE BOARD
POLLING SOFTWARE	
Bitbus Interconn. ↓ To Control Electronics	Bitbus Interconn. ↓ To Control Electronics

Fig. 7 - RF Console SW vs. Simulation SW.

REFERENCES

1) A. Bosotti et al., New Developments in the Control Electronics for the Milan Superconducting Cyclotron RF System, paper presented at this conference.
2) A. Bosotti et al., Computer Control for the Milan K800 Cyclotron RF System, Proc. I EPAC, Rome 1988.
3) F. Aghion et al., The Operator Interface in the Control System for the Milan Superconducting Cyclotron, Proc. XI Int. Conf. on Cyclotr., Tokyo 1986, IONICS Pub.
4) F. Aghion et al., The Distributed Control System with Decentralized Access to an Optical Bus for the Milan Superconducting Cyclotron, Proc. XI Int. Conf. on Cyclotr., Tokyo 1986, IONICS Pub.
5) Multibus and Bitbus are trademark of INTEL Corp.
6) C. Pagani et al., Full Power Tests of the First RF Cavity for the Milan K800 Cyclotron, Proc. XI Int. Conf. on Cyclotr., Tokyo 1986, IONICS Pub.
7) C. Pagani, RF System of the Milan K800 Cyclotron, Proc. X Int. Conf. on Cyclotrons, East Lansing 1984, IEEE cat. N. 84CH 1966-3.