OVERVIEW OF STATION COMPUTER SOFTWARE

crs/June 1990

This is an overview of the resident software in the station computers of GMRT, dealing primarily with the software protocol and communication handling. In particular, we will not discuss here the software specific to an observation like the list of commands related to servo computer or the list of commands from Central Supervisory Computer (CSC) interpreted by the station computer.

HARDWARE

There will be 3 different types of processor boards available at each station - which will be called for brevity the Servo Control Computer (SCC), Station Supervisory Computer (SSC) and Monitor/Control Module (MCM). The MCM and SSC are being designed by A Ramakrishna and the SCC has been designed by the BARC team. The details of servo control and any safety measures are all handled by the SCC, which is commanded by the SSC through an asynchronous serial port (RS232C 4800 baud). The MCM is in fact a group of identical 8031-cards connected to the SSC on an RS 422 bus. The SSC, based on 80186, has three serial i/o channels - a synchronous (HDLC) line to the CSC through the optical fibre, RS 422 (bus mode) line to all the MCM at the station, and an RS 232 to the SCC.

SOFTWARE ON MCM

The Monitor/Control Module has to provide three basic functions: (a) Acquire data on A/D converters (upto 64 channels) which are an integral part of each MCM; (b) Output appropriate signals to various modules to provide necessary control functions as commanded by SSC; (c) Initiate Walsh-function pulses to provide phase-switching (THIS ROLE IS NOT YET WORKED OUT IN DETAIL).

All these functions are achieved by executing a program contained in EPROM and coordinating with the SSC for receiving appropriate commands. Whenever SSC wants to communicate to an MCM, it sends a special byte with 9th bit on, which generates an interrupt on all the MCMs connected to the SSC. The remaining bits contain an address unique to each MCM as defined by the dip-switch setting. If the address matches, the MCM will know that it can talk to SSC; all other MCMs will ignore any input from MCM until their turn comes. (MCM can still broadcast messages to all MCMs by using a special "address" with 9th bit on; this will cause all MCMs to receive data from MCM, but without altering the existing privilege of a unique MCM being able to transmit data to SSC.) Each "message" is initiated by two bytes - one special with 9th bit on, and the other contain the message length (number of bytes).

The amount of data that can be held by each MCM is limited by the capacity of 128 bytes of internal RAM (there is no other data memory). Out of this, 20 bytes are reserved for message received from SSC (maximum packet size), some bytes are reserved for various i/o and other functions; 3 bytes are reserved for desired control bits, and 3 bytes are reserved for status (which includes the control bits actually set, in contrast to what is required). Eight bytes (64 bits) signify the absolute channel numbers which need to be sampled. Although all 64 channels can be sampled by each MCM at a given time, in practice only a subset are selected at a time.

Data acquisition by MCM can be operated in three modes - obtain raw data (one byte for each channel), or obtain a 2-bit (3-level) status for each selected channel, or obtain a 4-level histogram (3 explicit levels, since total number of samples is known) for each selected channel over a specific duration. The maximum number of channels that can be selected at a given time depends on the mode: upto 64 for raw mode, upto 32 for 2-bit status, and upto 16 for the histogram-mode. In order to form the 2-bit status, two threshold comparison values are specified by SSC for each channel; similarly, in the case of histogram, 3 comparison thresholds are specified for each channel.

The typical interval between successive samples acquired by the microcontroller is about a millisecond - the interval does not vary much with the number of selected channels. In the raw or status modes, the numbers available in the MCM buffer correspond to the most recent value of the sample value OR the 2-bit status. Once initiated, the sampling will proceed until SSC commands otherwise. However, the histogram mode is somewhat different. Sampling will continue until the bytes saturate - until one of the levels is reached 255 counts; subsequent sampling will be suspended until SSC accepts the buffer to be transmitted and re-initiates sampling.

Note that the restriction on the maximum number of channels at MCM level does not impose any restriction on the total number that can be monitored. If MCM cannot handle more than 16 at a time, SSC can cycle through all the channels in groups of 16 until all required channels are monitored within each monitor cycle. (A "monitor cycle" is expected to be a few tens of seconds typically.)

SOFTWARE ON SSC

The Station Supervisory Computer interacts with other microprossers in the station as well as to the central electronics building through the HDLC link. The software in this computer consists of two parts: (a) a real-time kernel which gets loaded at power-on and will get activated whenever an interrupt is raised by one of the serial ports or a timer; and (b) a foreground process which can be considered as a subprogram of the supervisory kernel. While the supervisory kernel is taking care of all communication and protocols, the foreground process is responsible for the control/monitor operations desired at the station. The following is a brief description of the various interrupts envisaged at the station:

(a) Timer interrupt - typically, once a second. This is used for scheduling various tasks, perform housekeeping activities like marking a packet for transmission at the next opportunity, etc. exchange commands/data with the servo-control-computer through the serial port and store the received data in the appropriate buffer and set a flag for the foreground process to know about it; interpret the commands from the CSC if the relevant flag indicates that a kernel function was requested from the CSC.

(b) HDLC - This occurs whenever the central computer sends a packet of information meant for the specific SSC. If the address decoded by the HDLC chip matches the station address, a DMA is initiated and an interrupt is raised. The interrupt service routine consists in (a) checking for parity error, if any, in the received packet - and marking a repeat request during next transmission; (b) interpreting if the CSC is switched to listen to this station and if so, transmit any pending packet to the CSC; (c) interpreting if the received packet is meant for the supervisor kernel or for the foreground process and set the appropriate flag indicating the same.

(c) Servo Serial Port (RS 232) - this interrupt is raised if any alarm condition occurs in the servo system.

(d) Keyboard interrupt (?) - This is mostly for diagnostic purposes, and is raised by every character typed on a local keyboard connected temporarily at the station.

Depending on the size of the kernel, the absolute start address of the foreground process will be frozen to a pre-determined value contained in the EPROM. The supervisor kernel maintains a look-up-table which is originally copied from EPROM into RAM, which associates each process with a unique identification (4-character name) and start- and end- addresses of the code located in EPROM or RAM. When the CSC requests a process to be spawned, the code is first copied into RAM starting with the pre-determined start address for the foreground process, and then control is passed to the appropriate start address. Some of the commands from CSC in this connection relate to accepting fresh code for a new process to be downloaded by the CSC; stop the current process; spawn a new process identified by its name by the CSC, but moved from memory locations indicated by the look-up-table maintained by the kernel.

When CSC is in listen-mode, one and only one packet is transmitted by SSC. If no information is pending to be transmitted, a mere acknowledgment is transmitted. It is expected that the maximum packet size is informed each time by the CSC -- e.g. the command could be "SEND" or "SEND cnnn", where c is a code and nnn is a 3-digit number indicating the maximum packet size in bytes which the CSC is prepared to receive; when cnnn is not specified, a default value of 80 bytes is assumed (not necessarily equal to the previous packet size!)

The single-character code mentioned above indicates if the logical packet is complete in the present physical packet or not. By convention, multiple logical packets will be allowed in a physical packet if, and only if, all the logical packets are entirely represented within the same physical packet. A special procedure is employed to split a logical packet between multiple physical packets. In this case, all relevant physical packets will begin with "Bnnn" where nnn is a 3-digit block number. For the first block, this field will be "B001" followed by ":mmmmm", where mmmmm is a 5-digit number indicate the size of the logical packet (bytes). The last physical block representing the logical record will have "Ennn" instead of having "B". The case of record-splitting is especially important when downloading code to SSC.

Each information to be sent by SSC is encoded into a separate logical packet, which includes the following fields:

    size;label;argument(s)

where "size" is in one of the forms "nnnn", "B001:mmmmm", "Bnnn" or "Ennn". "Label" is an 8-character name beginning always with a non-numeric, non-blank character (followed by any nonblank character/s) with trailing blanks if necessary. The number and type of arguments depend on the label. Examples of label+argument field are:

COORD	179.00 110.00	(az-el coordinates in degrees)
HA	-29.00	(hour-angle in degrees)
MONaaa	Oooo	(monitor-address contents in octal)
REPLY	ssss	(reply to query contained in packet ssss received from CSC)
OBSPROG	name	(name of foreground process, if any)
ALRM.WH		(Alarm - wind high) - no argument
ALRM.AMH		(Alarm - azimuth motor current high)
REJECT	ssss	(Incompatible/unintelligible command indicated by packet ssss of CSC)
REPEAT		(Repeat previous packet - parity err)
BEGIN	proc_name	(proc_name has been activated)

A specific request leading to REJECT as reply corresponds to seeking a monitor-parameter without prior information that the value is to be monitored. Request for any monitor point can only be made after placing the relevant address in the monitor-list, e.g., by using one of the following FICTITIOUS commands:

```
     SET.MON1+aaa                       (add new address aaa to list)
     SET.MON1-aaa                       (remove address aaa from list)
     SET.MON aaa bbb:ccc ddd ...        (choose addresses aaa, bbb-to-ccc ...)
```

Then any request for a monitored parameter must have been implied in one of these commands for the CSC to accept.

A system maximum packet size of 512 bytes is imposed for all communication between SSC and CSC. The typical buffer allocated for transmit/receive generally is, however, 1-2kB.

A word about the packet sequence number. Note that this is allocated only by the CSC and not by SSC. If a station wants to enquire about some info it sought earlier, it will re-send the whole packet and not some arbitrary reference number it maintains. Thus, a packet originating from CSC and received by SSC has the following format:

```
     size;pkt_seq_no.;label;argument(s)
```

The pkt_seq_no is set to 0000 if no acknowledgment is sought explicitly by CSC - This provision is made for utilising broadcast-mode for special purposes, e.g., setting time (current IST/LST contained in the packet), or sending some emergency/close-down command (e.g., park all antennas).

Note that the list of all commands has still not been compiled. At the moment, we only have a list commands for Servo Computer agreed between BARC and us. The commands listed above are only illustrative and do not follow the actual convention to be embedded in the packets exchanged between SSC and CSC.