VLBA CORRELATOR: REVIEW OF BLOCK–II CORRELATOR CONTROL

T. J. Pearson

January 2, 1985

The following notes are based on discussion with D. Rogstad on December 7, 1984. Some of the text has been extracted from the comments in the Forth code.

OVERVIEW

Figure 1 (System Level Siting Diagram) shows the hardware layout of the Block–II Correlator. The station hardware and Correlator/Tensor hardware are interfaced to the VAX–11/780 by two DR70 (16-bit parallel) interfaces on a single Massbus. The Correlator Control Program (VCOROP) is written in Forth and communicates with the VMS operating system via the Forth kernel. All VMS instructions, system services, and library routines are available to the Forth programmer. The operator interface consists of an operator console (hardcopy terminal) which is used for starting up the Correlation process, and a workstation (graphics terminal with attached printer) which is used for controlling and monitoring the correlation of a single experiment. When two separate experiments (called "groups") are to be processed simultaneously, two workstations are used.

Figure 2 (Software Interfaces and Data Flow Diagram) shows the top-level software organization. All communication with the hardware is carried out in the main Control and Model Program, started up from the operator console. Each group is processed as a VMS subprocess of the main program, communicating with the main program via a "global section" (shared common). Correlation parameters are read from a Correlation Control Record (CCR file), set up in advance; for Mark–III this can be generated automatically from the station log files. Output data for this group are saved on disk in a Post-Correlation Record (PCR file).

For time sequencing the correlator control program uses the VAX/VMS Asynchronous System Trap (AST) mechanism, and an understanding of this mechanism is required for understanding the program operation. An AST is a software-simulated interrupt to a user-defined service-routine. An AST enables a process to be notified asynchronously with respect to its execution of the occurrence of a specific event. For example, a process may request an AST when an I/O operation is completed, or at a specific time of day. When the event occurs, the operating system interrupts the normal execution of the process and executes the user-specified routine. When that routine exits, the system resumes the process where it was interrupted.

MAIN PROCESS

Figure 3 shows the top-level structure of the main process (VCOROP) and one subprocess (GROUP1). When the program is started up, control is passed to the Forth command interpreter (NTRPRET0), which waits for a command from the operator. When a command is received, it is processed and the process waits for the next command (see Figure 4). Most commands merely set flags which are interpreted by the routines that execute in reponse to ASTs. These ASTs are established by the operator "INITIALIZE" command. The INITIALIZE command issues an I/O request to the station hardware asking it to return the current (tape) time, specifying that an AST is required on completion and specifying PROCESS as the AST routine. The station hardware returns the time on the next 1-sec tick (we shall assume that this "time read interval" [TRI] is always 1 sec in what follows, but it does not have to be). When it has finished its execution (described below), PROCESS sends a command buffer (filled with hardware requests by other routines) to the hardware and queues another read request. Thus the PROCESS routine is executed once every second on the second, synchronously with the master clock to which the tapes are synchronized. Typically PROCESS takes less than 0.1 sec; whatever happens, it mustn't take more than 1 sec!

A variety of other routines are executed in response to ASTs queued by PROCESS (Figure 5). These typically take longer than 1 sec to complete, and so they are interrupted by PROCESS. This requires PROCESS to run at a higher priority (AST level) than the other routines; PROCESS runs at "supervisor" level while the others run at "user" level.

PROCESS (T21): Supervisor level AST routine for executing the correlation control process. This AST is the I/O completion routine for the station time-read QIO request setup in PROCSTAT and actually queued in PROCTRAN. The station H/W does not return from this request until the next time-read interrupt, which occurs once each Time-Read-Interval (TRI). A TRI has been set to be the same interval as required to correlate 4×10^6 bits, or 1 second at the 4 Megabit/sec correlation rate.

TMODAST (T22): AST routine to call TONEMOD module for determining the phase parameters to run the tone-tracker within the Processor H/W. It is executed in MAIN process, and queued in the PROCESS AST when the appropriate TRI counters so indicate (typically every 37 sec).

GMODAST (T23): AST routine to call GEOMMOD program for calculating the delay and phase parameters to run the Processor H/W. It is executed in MAIN process, and queued in the PROCESS AST when the appropriate TRI counters so indicate (typically every 31 sec). GEOMMOD determines the coefficients of a cubic interpolation which is evaluated by the hardware.

STNSTAST (T24): AST routine to check the station status error summary words for errors, write out any group 0 status or error log, and signal the appropriate group subprocess if status information must be sent. Executed in MAIN process (typically every 41 sec).

CORSTAST (T25): AST routine to check the correlator status error summary words for errors, write out any group 0 status or error log, and signal the appropriate group subprocess if status information must be sent. Executed in MAIN process (typically every 43 sec).

PRERRAST (T26): PRocessor-ERRor AST routine to check the I/O status words for all the I/O between the VAX and the processor stations and correlator hardware. This is run in the MAIN process in supervisor mode.

GROUP SUBPROCESS

Each group subprocess has a similar structure to the main process; it is normally waiting for commands from an operator (Forth interpreter), from which state it is interrupted to execute various AST routines set up by the PSTART command.

EVENTAST (T31): Event Flag test AST for initiating the various routines to be executed within each group when the corresponding event flags are set. If a given routine's flag is set, the routine is queued to execute as a USER mode AST. This is to insure that these operations will complete with more priority than operator initiated operations. Executed in a GROUP subprocess.

1. SAVE AND SET ORIGINS: Execute the word for saving the various background origin addresses on the RETURN stack. Using the AST parameter, set the GROUP origins.
2. CHECK IF MODEL STILL GOOD: Loop through the various geometric model times and test if the master clock time is still before the stop times of all the delay models being used. Set the model okay control flag for any channel model still running.
3. UPDATE MODEL TABLES: Check the end-of-scan control flag and the model initialize flag to determine if new scan parameters should be loaded or a model initialization should be performed. The T32 AST is queued, as needed, to perform these operations.
4. SEND MODELS TO PCR: Check the flags which indicate whether the tone models, geometric models or tensor configuration tables need to be written to the output file. This will occur at the update intervals (when the tone or geometric models are recalculated) or when any of the tensor configuration parameters change. If any of these tables need to be written to the output file, then queue the T33 AST which will actually output them.
5. CHECK STATION ERRORS: Check the flag which indicates if any errors were received on the last read from the station H/W. If there are station errors, then queue the T34 AST which will check their severity and take the appropriate actions depending on that severity.
6. CHECK CORRELATOR ERRORS: Check the flag which indicates if any errors were received on the last read from the correlator H/W and/or tensor board. If there are correlator or tensor board errors, then queue the T35 AST to check their severity and take the appropriate action depending on that severity.
7. PROCESS TENSOR DATA: Check the flag which indicates if there is any new tensor data to process. This will occur whenever the tensor H/W outputs to the computer. If there is tensor data to process, then queue the T36 AST which will actually do the processing. This may include such operations as writing the tensor data to the output file and updating the information on the display terminal.
8. PROCESS STATION DATA: Check the flag which indicates if there is any new station data to process. This will occur whenever the stations H/W outputs to the computer. If there is station data to process, then queue the T37 AST which will actually do the processing. This will include such operations as writing phase calibrator data to the output file and updating the information on the display terminal.
9. PROCESS CORRELATOR DATA: Check the flag which indicates if there is any new correlator data to process. This will occur whenever the correlator H/W or tensor board output to the computer. If there is correlator data to process, then queue the T38 AST which will actually do the processing. This will include such things as

writing data to the output file, updating the information on the display terminal, and fringe searching.

10. PROVIDE GROUP CONTROL: Check the flag which indicates if any control functions are to be performed for this user group. These include operations like IDLE, but executed from within the processor program due to some event rather than executed by the operator from the keyboard. Various messages from the main process are also included. If there are any of these type events, as indicated by bits in the GCTFLGO, then queue the T39 AST to handle them.
11. REQUEUE THE EVENTAST ROUTINE (T31 AST): If the group event AST is still ON, then requeue this AST to execute again at a real-time interval in the future (given by DELTIME). Otherwise (if the group event AST has been turned off), do not requeue the AST. This will cause no data to be output or processed until a START is again issued for the group (the group and the group event AST are turned ON) at which time this AST (EVENTAST) will again be queued.

MODINIAST (T32): AST routine to perform the initialization operations on the Tone and Geometric models when any of their input parameters change. Model table status flags are set when table parameters are changed. These status flags are checked to see if the tables need initialization before the new values can be used. The MODINITL routine is called to perform the actual function. New scans are read from the correlation control record if the time is appropriate and the processor is in the automode.

MODOUTAST (T33): AST routine to output the Tone Models and Geometric Delay and Phase Models to the output file. It executes the routine MODTOPCR with the correct origins.

STNERRAST (T34): AST routine to process any severe status errors in the station hardware.

CORERRAST (T35): AST routine to process any severe status errors in the correlator hardware.

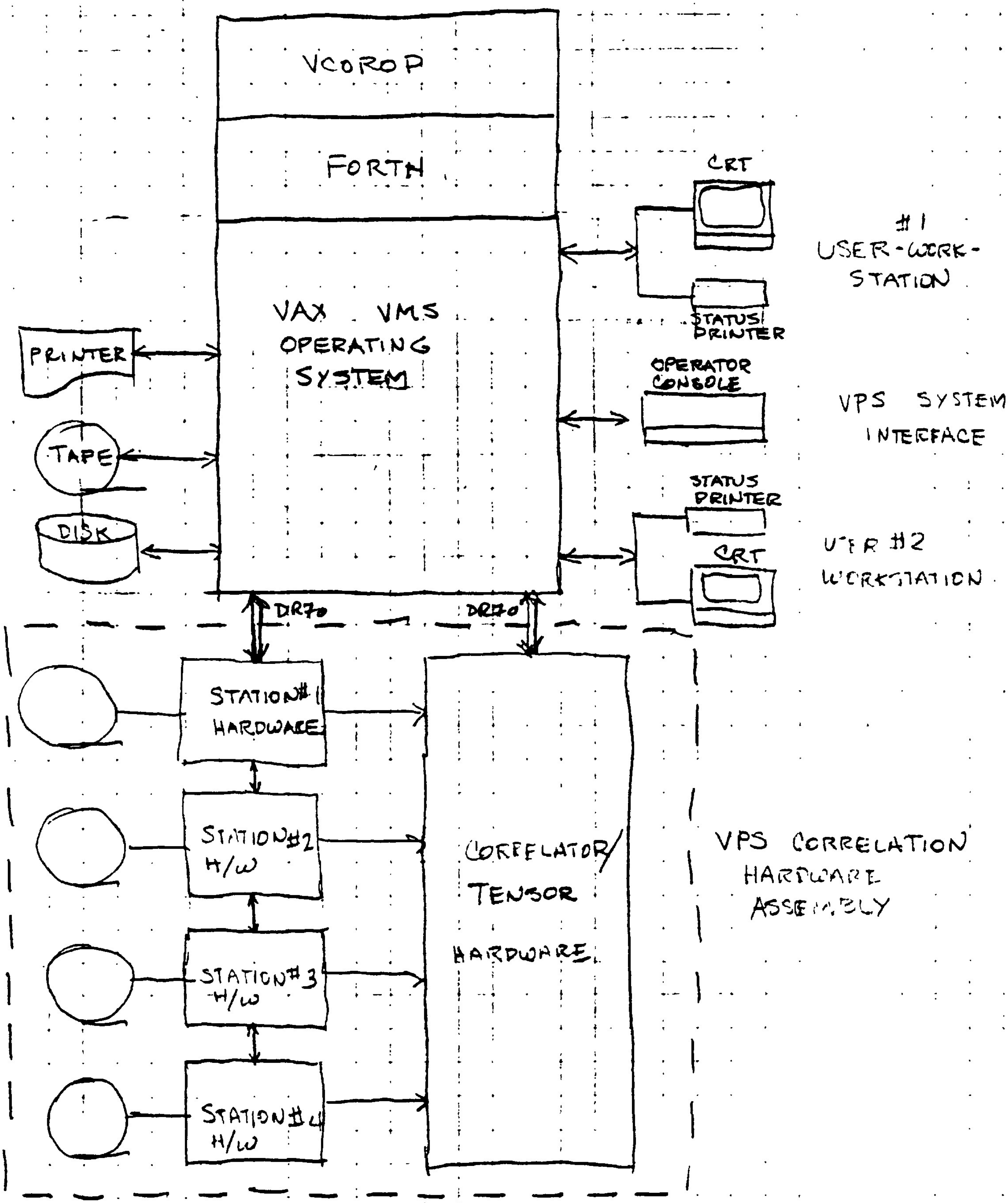
TSRDATAST (T36): AST routine for handling of the data that is dumped from the Tensor hardware.

STNDATAST (T37): AST routine for handling of the data that is dumped from the Station hardware.

CORDATAST (T38): AST routine for handling of the data that is dumped from the Correlator hardware.

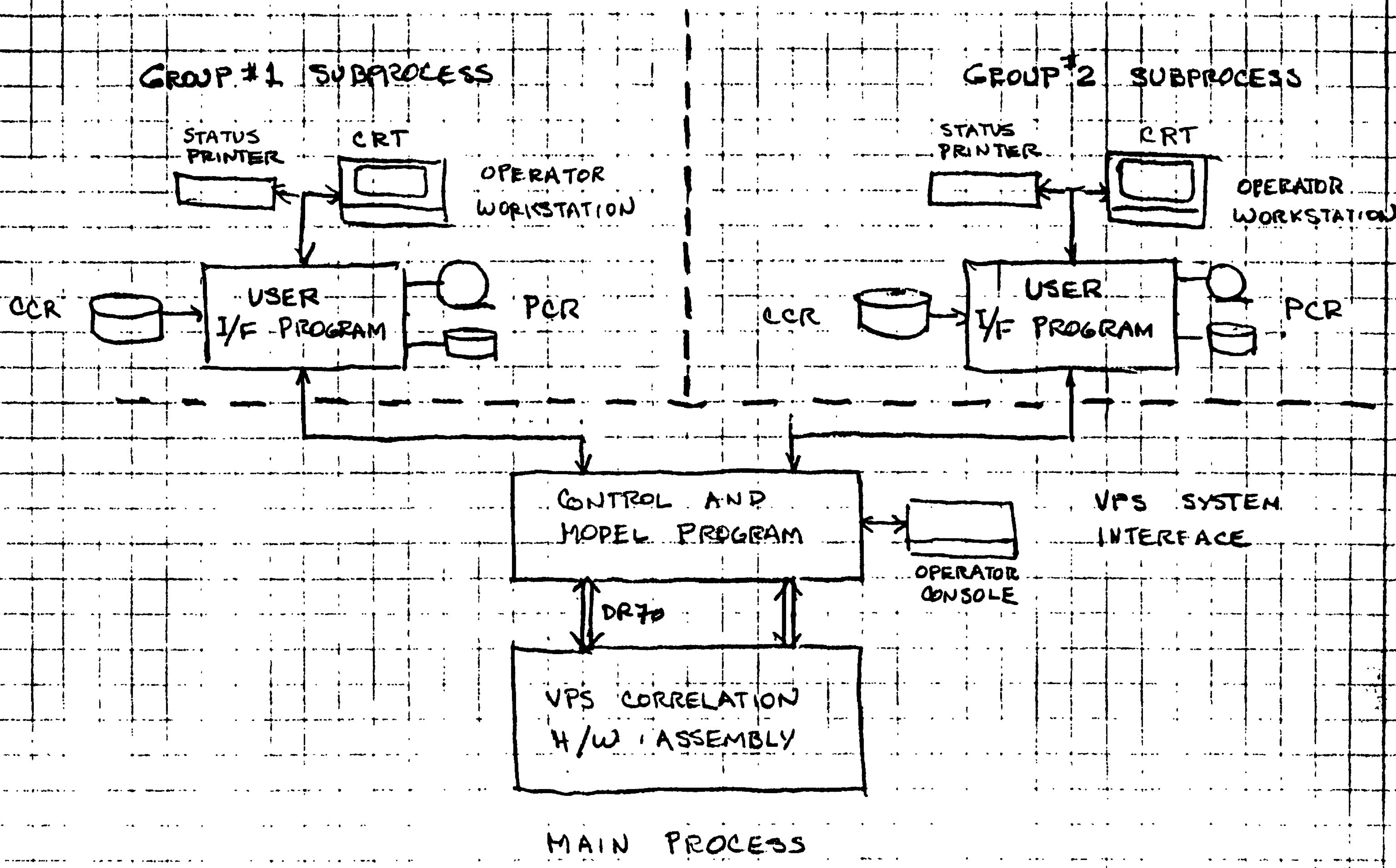
GCNTRLAST (T39): AST routine for control of the user group state from inside of the AST routines, in contrast to operations coming from the operator keyboard.

Figure 1



SYSTEM LEVEL SITING DIAGRAM

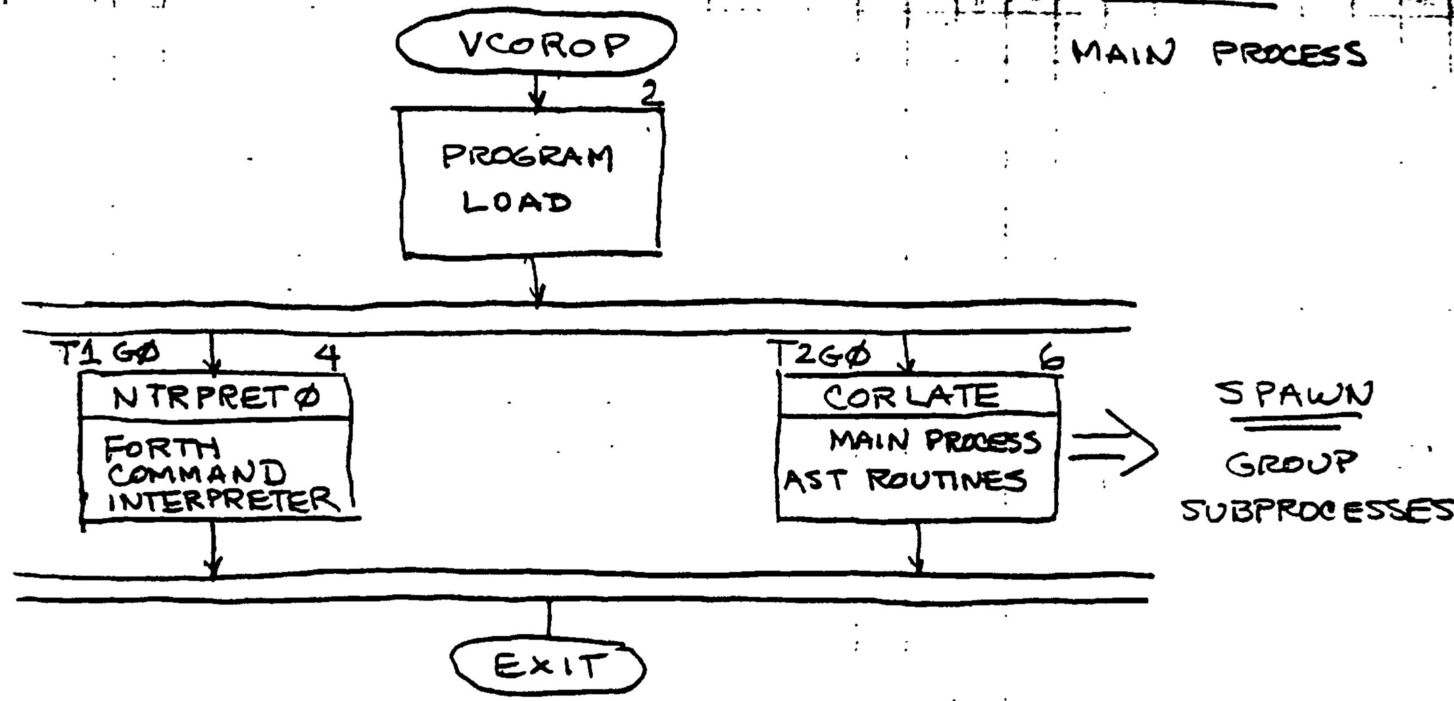
Figure 2



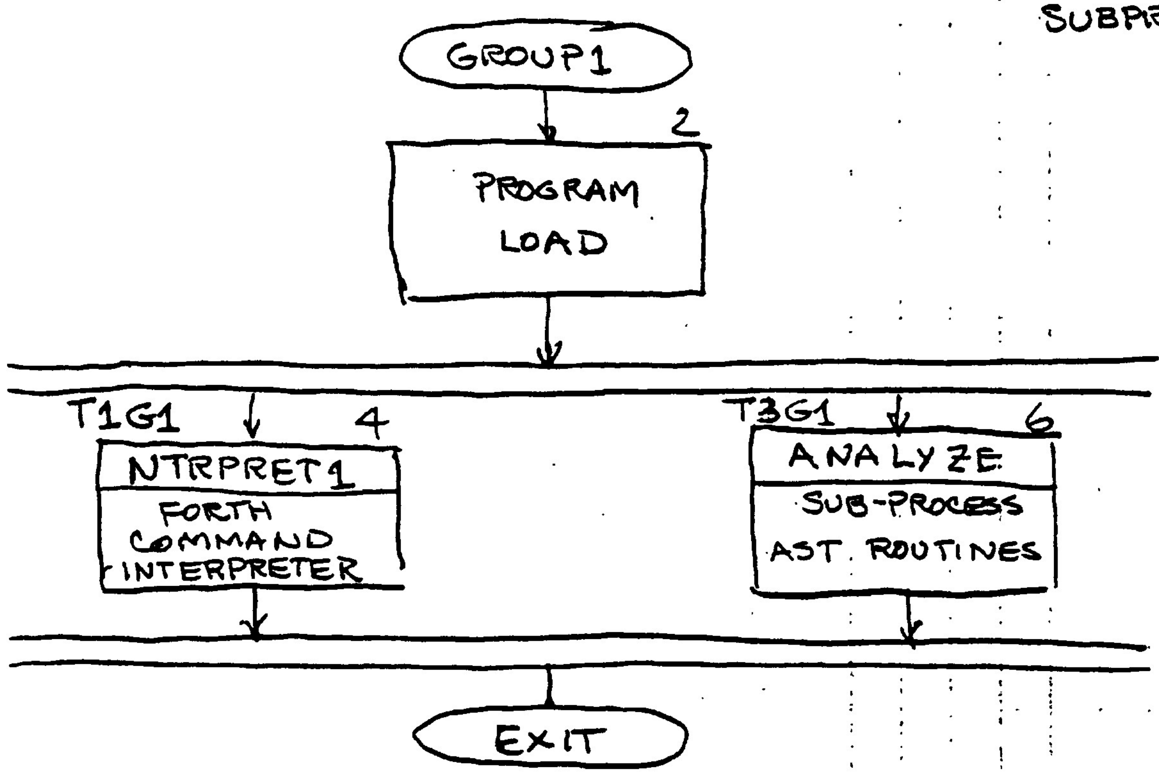
SOFTWARE INTERFACES AND DATA FLOW DIAGRAM

Figure 3

MAIN PROCESS



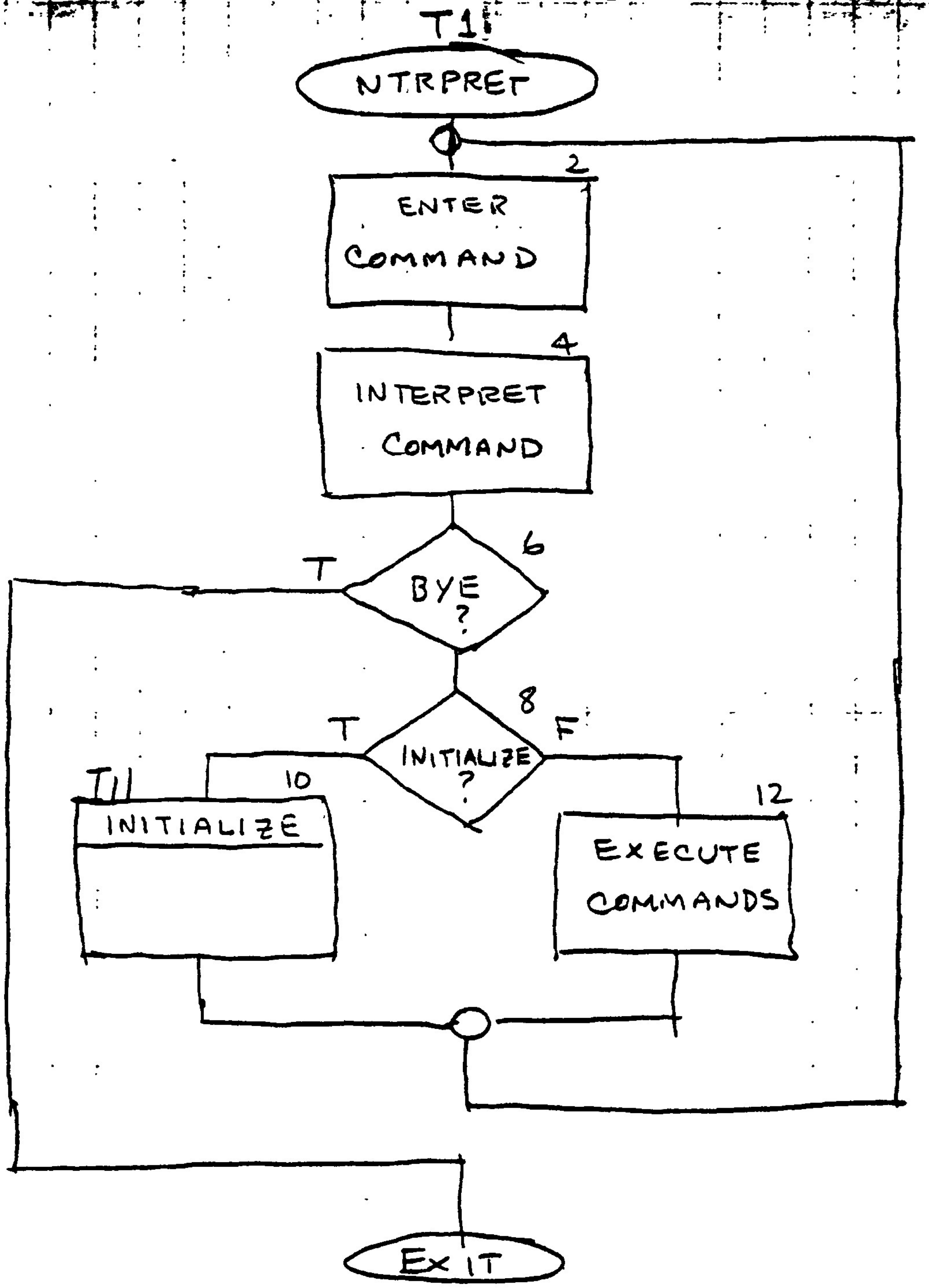
SUBPROCESS #1



SUBPROCESS #?

o
o
o
o
o

Figure 4



T11
NTRPRET
2
ENTER COMMAND
4
INTERPRET COMMAND
6
T
BYE ?
8
T
F
INITIALIZE ?
T11
10
INITIALIZE
12
EXECUTE COMMANDS
EXIT

TRI = time real interval = 1^s (usually)