DSS 13 Microprocessor Antenna Controller

R. M. Gosline
Radio Frequency and Microwave Subsystems Section

This article describes a microprocessor-based antenna monitor and control system with multiple CPUs. The system was developed as part of the unattended station project for DSS 13 and was enhanced for use by the SETI project. The operational features, hardware, and software designs are described, and a discussion is provided of the major problems encountered.

I. Introduction

The original work on the DSS 13 antenna controller was performed as part of the unattended station development [1]. The objectives were twofold: (1) to replace the existing antenna controller subsystem (consisting of a Modcomp II minicomputer and a custom-designed remote data collection unit) with standard microprocessor controllers similar to those of the other subsystems at DSS 13 through the use of modules recommended by the Computation and Control Module (CCM) committee; and (2) to generate new software using a high-level programming language and a top-down structured design. Use of only commercially available modules simplified the requirements for documentation and sparing.

Additional improvements were made to the system to support the SETI (Search for Extraterrestrial Intelligence) project. The improvements consisted of two additional pointing modes for random curve fitting, control of a noise diode for the Noise-Adding Radiometer (NAR), and the capability to lock either servo axis in high speed.

II. System Description

The DSS 13 26-meter antenna is an azimuth–elevation type with a hydraulic servo system. Each axis has a high and low speed capability. The maximum tracking rate in low speed is 0.2 degree/second in azimuth and 0.05 degree/second in elevation. In high speed, the maximum rate is 2 degrees/second for each channel. The encoder resolution is 20 binary bits, corresponding to 0.34 millidegree. The antenna was constructed in 1958 at DSS 12 and was moved to its present location at DSS 13 in 1961. Thus it is old and rather fragile, requiring precise control of acceleration and deceleration. The maximum values are shown in Table 1.

Operation of the antenna in an unattended mode requires careful monitoring of many sensors to prevent damage to the system in the event of a failure. Depending on the situation, an out-of-limit condition will cause one or more of the following actions:

(1) An operator warning message.

(2) Reduction of drive voltage.

(3) STOP (controlled deceleration and brake activation) and shutdown of pumps and electronics.

(4) Move to the stow position (zenith) and shutdown.

A list of all analog sensors is shown in Table 2. All digital sensors are tabulated in Table 3. For data reporting purposes, some angle values are treated as sensor data and are tabulated in Table 4.

Unattended or remote control requires the use of high-level commands that are designated station-level commands. The commands may be issued from the NOCC, the station controller, or the local console.

A. Station-Level Commands

INIT	Turns on electronics and hydraulic pumps.
CNF	Indicates the start of configuration or predict data. The various types of configuration commands are shown in the next section.
OFF	Performs a STOP; then shuts off hydraulics and electronics.
OPR	Operates; moves antenna to follow predict data.
STOW	Moves to the stow position and performs a STOP.
STOP	Decelerates and applies brakes.
STR	Requests status report (general health or selected data). The various types of status requests are shown in Subsection C.
CLEAR	Resets lockout after a fault.
HELP	Displays one of the help menus.
CLOK	Synchronizes the APS clock to the station clock.
CLDO	Clears direct offsets.
CLPO	Clears position offsets.
CLRO	Clears rate offsets.
NDON	Turns noise diode on.
NDOF	Turns noise diode off.

B. CNF or Configuration Data Commands (All Preceded by CNF)

YEAR ⟨year⟩	The current year must be entered once after loading the program because the value is not available from the station clock.
MODE ⟨IDLE, AZEL, SID, 3DAY, 3RAD, 3AZL⟩	Selects the desired tracking mode. There are five tracking modes and an idle mode.
IDLE	This mode bypasses any position update calculation holding the antenna at the current point. The mode is used mainly for testing and maintenance.
AZEL	The AZEL mode allows pointing to a given azimuth–elevation angle pair. The purpose of this mode is to point to objects that are fixed relative to the local tangent plane (the collimation tower or synchronous satellites).
SID	A sidereal rate mode is used for tracking objects at sidereal rate from right ascension–declination coordinates. This mode is used for objects such as radio stars.
3DAY	This mode is used for tracking objects such as deep-space spacecraft or planets that are close to sidereal rate. A second-order curve fit (parabolic) is generated from three sets of values of right ascension and declination for 0 hours GMT on 3 successive days.
3RAD	This mode is similar to 3DAY but is more general in that a day–time parameter is specified for each point. The time interval may range from 2 seconds to years. Provision has been made to accept a fourth point which will cause points 2, 3, and 4 to shift to points 1, 2, and 3 at the original time of point 2. Thus, a continuous curve in right ascension–declination coordinates can be followed. The fourth point, if used, must be sent in the time interval between the time of points 1 and 2 and must have a time greater than the time of point 3. Normally, the antenna tracks between points 1 and 2 but may continue beyond point 2 if a fourth point is not used.
3AZL	This mode is identical to 3RAD except that the azimuth–elevation coordinate system is used.
NAME ⟨reference name⟩	Allows the user to label a configuration set with a reference name.
Xn ⟨az or ra angle⟩ [n = 1, 2, 3, 4]	Specifies an angle predict for azimuth or right ascension. The value is interpreted according to the MODE. Right ascension values may be formatted as decimal degrees or as hours, minutes, and seconds.
Yn ⟨el or dec angle⟩	Specifies an angle predict for elevation or declination. The value is interpreted according to the MODE. Declination

	values may be formatted as decimal degrees or as degrees, minutes, and seconds.
Tn ⟨day–time for Xn, Yn⟩	Time reference for 3RAD or 3AZL modes.
WRAP ⟨L, R, C⟩	The mechanical range of movement in azimuth is extended 90 degrees beyond the normal 360 degrees of azimuth designated as center. The counterclockwise extension is designated the left cable wrap and the clockwise extension is designated the right cable wrap. A new track, defined by issuing an OPR command, normally starts in the center wrap regardless of the antenna's current position. Tracking is allowed to continue into either the left or the right wrap. This command allows an AZEL mode track to be started in left, right, or center cable wrap-up for maintenance and testing purposes.
REFC ⟨ON, OFF⟩	Allows manual control of the refraction correction. The default is on for right ascension–declination predicts and off for azimuth–elevation predicts.
DFAZ ⟨az value⟩	Directs azimuth offset.
DFEL ⟨el value⟩	Directs elevation offset.
AEAZ ⟨az value⟩	Azimuth offset corrected for distortion. The value entered is divided by the cosine of elevation and is not cleared by CLDO.
AEEL ⟨el value⟩	Directs elevation offset. This parameter results in the same offset as DFEL but is not cleared by CLDO.
DFRA ⟨ra value⟩	Directs right ascension offset.
DFDC ⟨dec value⟩	Directs declination offset.
AZRT ⟨az rate⟩	Azimuth rate offset in degrees/second.
ELRT ⟨el rate⟩	Elevation rate offset in degrees/second.
TOFF ⟨day–time value⟩	Allows a time offset from the station clock for testing purposes.
HSAZ ⟨T, F⟩	If true, this parameter causes the azimuth servo channel to remain in high speed. If false, the normal speed selection logic is in effect.
HSEL ⟨T, F⟩	If true, this parameter causes the elevation servo channel to remain in high speed. If false, the normal speed selection logic is in effect.

C. Status Request Commands (All Preceded by STR)

These commands will display the requested data on the operator display, printer, and station controller.

(no parameter)	Reports general subsystem health and whether in local or remote control (selected by local function key).
CNF	Displays the entire current set of CNF parameters.
ANA [⟨channel IDs⟩]	Displays the selected analog channel. If no channel is specified, all analog channels are displayed. The analog channels and channel IDs are shown in Table 2.
DIG [⟨channel IDs⟩]	Displays the selected digital channel. If no channel is specified, all digital channels are displayed. The digital channels and channel IDs are shown in Table 3.
ANG [⟨channel IDs⟩]	Displays the selected angle data value. If no value is specified, all angle values are displayed. The angle data channels and channel IDs are shown in Table 4.
F	Displays all analog, digital, and angle values.
TANG	Displays time-tagged (at the time of sampling) azimuth and elevation position readouts. Normally, all data values are time-tagged when displayed. This may be a second or two after the sample is actually read due to higher priority tasks.

D. Maintenance Mode Commands

A set of low-level commands is available to assist station personnel in performing periodic maintenance or troubleshooting functions. These commands bypass most automatic sequences and allow actuation of primitive control functions such as high–low pump selection, brake actuation, pre-limit overrides, and application of direct drive voltages for the servo valves. These commands are available only in the local control mode.

III. Hardware

The system consists of two controllers as shown in Fig. 1. The Data Collection Controller, located in the antenna pedestal, collects analog and digital data from various sensors and reports this data to the main controller, located in the control room. The main controller collects additional data from monitor points located in the control room. The existing sensors and the data collection configuration were not altered to allow switching control between the Modcomp and the new controller during the debugging phase. This allowed the station to perform its normal tracking schedule with minimal interference. The Modcomp was removed upon the completion of integration of the new controller.

The main controller's task is to provide operator interface, communication, display, local sensor interface, and the necessary logic and calculations for controlling the antenna to perform tracking and maintenance functions. Communication with the station controller is through a star switch utilizing a JPL standard 15-line interface.

The Data Collection Controller configuration is shown in Fig. 2. It consists of a single board computer (8080 CPU) with 8 kbytes of on-board PROM and 4 kbytes of RAM. The CPU drives a short-haul modem through a serial port and is interfaced to 24 digital channels through on-board parallel ports. An external digital interface chassis uses commercial optical isolation modules to convert 24-Vdc antenna levels to TTL. A 32-channel differential input multiplexer and an analog-to-digital converter provide interface to the analog channels. All input lines are protected from lightning discharges by filters and varistor-type surge protectors. A 64-kbyte RAM module, a floppy disk controller, a communication expansion module, and dual floppy disk drives allow development and testing of the software in the controller. Both controllers utilize the Intel Multibus I. After checkout, the program was burned into PROM and the development modules removed, leaving only the CPU and analog-to-digital converter cards in the Data Collection Controller.

The main controller, configured as shown in Fig. 3, utilizes two CPUs on the same Multibus. CPU 1 provides interface to the angle encoders through its parallel ports and performs most of the tracking-related calculations. CPU 2 provides interface to the star switch through its parallel ports and the serial data from the Data Collection Controller and modem through its serial port. CPU 2 also performs the conversion of data to floating point and ASCII, performs the major part of the tasking logic, and initiates appropriate action in the event of a fault detection. A video-keyboard display card (actually a third 8080 CPU) handles the screen display on a high-resolution monitor and operator interface. The design allows near-real-time data to be displayed in the non-scrolling upper portion of the screen while scrolling information, alert messages, operator inputs, and specific data requests (via STR) are displayed in the lower portion. Upper screen data that is not within expected limits or states is displayed in reverse video and flagged for possible monitor action. The following screen displays and formats may be selected by single-stroke function keys:

(1) Normal tracking operation.

(2) Maintenance display.

(3) Analog sensors (displayed in engineering units).

(4) Digital sensors.

(5) Help menus (multilevel).

Another function key is used for the selection of local or remote control. Sensor interface to the main controller is similar to that of the Data Collection Controller for monitor points originating or reported near the control room. The addition of an input–output expansion module is necessary because the parallel ports of both CPUs are used by the star switch and angle readout interfaces. Numerical computations are facilitated by three floating point processors dedicated to tracking computations, digital servo loop computations, and various data conversion tasks such as binary to floating point or ASCII. A communications expansion module provides interface to a printer for data logging or program lists and an additional terminal to allow each CPU separate control for debugging purposes. A digital-to-analog converter module provides the output drive voltages to operate the servo hydraulic system. Two extra digital-to-analog channels are available for recording of test data such as offset overshoot. As with the Data Collection Controller, all software was developed in the controller using a floppy disk controller and two disk drives.

IV. Software

The software is written in Intel's PLM-80 language. The program contains 10,209 lines of PLM-80 source code and 808 lines of assembly language for high-speed peripheral drivers for special input–output (floating point cards, 15-line software handshaking, and the binary angle position). The compiled program requires 51,523 bytes of program storage and 5948 bytes of data RAM. For comparison, the Modcomp minicomputer program contained 2300 lines of FORTRAN and 4600 lines of assembly language, requiring 30k words of memory (a Modcomp word is 2 bytes).

Loading of the program is accomplished by reading a fully linked execution module from disk and operating the program from RAM. Loading time is approximately 1 minute and is required only after a power failure.

The top-level flowchart for the Data Collection Controller is shown in Fig. 4. All of the flowcharts shown have been condensed for illustration purposes, and some modules may represent a combination of several. After an initialization module, the program enters an endless loop containing the routines REMDAT, which inputs data from the transducers, and REMSEND, which formats, adds a checksum, and sends the data over a serial link to the main controller. A timer on the CPU board provides a 50-ms interrupt which is divided modulo 4, causing the main loop to be executed every 200 ms. The serial line operates at 9600 baud. The data is formatted in two 73 ASCII character blocks, including a checksum, and sent continuously. Unused channels are included in the fixed format to allow for expansion. The data block can be displayed directly on a terminal if desired for verification.

The main controller top-level software is shown in Fig. 5. Both CPUs begin execution of the same code from the same memory location. Each CPU has the ability to interrogate itself to find out which CPU it is and to jump to separate initialization routines. From there, each enters a separate endless background loop except for the servicing of interrupts.

The level 2 flowchart for CPU 1 is shown in Fig. 6. The background modules perform the following functions:

CHKSUM — Verifies that the program code has not changed.

CHKSTAK — Generates a warning message if the stack is getting too large.

HISTBUF — Snapshots data RAM for analysis in case of a failure.

There are three interrupts used with CPU 1. Interrupts 3 and 4 are driven from the station clock at rates of 50 Hz and 1 Hz, respectively, and control all the time-dependent motion of the antenna for tracking. Interrupt 2 is used to read a character of serial time from the station clock. The station clock is read initially to establish time. Thereafter, the antenna clock is incremented in the 1-Hz interrupt routine. The 50-Hz routine counts each interrupt to ensure exact synchronization with the 1-Hz interrupt. If the count is not correct or if the antenna time does not match station time, an error message is generated. The count is also used modulo 5 to call the following modules every 100 ms:

READOUTS — Inputs current position from the angle encoders.

UPDATE — Updates the angle commands to the current time.

SERVO — Updates the digital servo loop. The loop equation was adapted from the MODCOMP program and is of the form Rn = K1 * Ro + K2 * Pe, where K1 and K2 are constants, Ro is previous rate command, and Pe is the position error. The program uses eight sets of constants for all three combinations of azimuth–elevation, high–low speed, and type I or II servo. If starting from rest, the servo is type I and switches to type II after 2 seconds. The values of K1 and K2 were taken from the MODCOMP program and adjusted by observing the system response to step functions. The final values are shown in Table 5.

D2AOUT — Updates the current output drive voltage.

After the interrupt 4 routine steps the clock, the following modules are called:

ANAACTCNT — Decrements counters that control action on analog channels that exceed error limits until specified delays have expired.

TIMERS — Decrements various state timers.

CMPRATES — Computes the measured angle rates by differencing the position readouts.

Depending on the tracking mode, one of the following modules is called to perform the appropriate calculation of the desired position for the next second:

IDLE, AZEL, SID, 3DAY, 3RAD, 3AZL

Three more routines are called:

CHKPRELIM — Checks the physical movement limits of the antenna and stops motion if the limits are about to be exceeded.

REFC — Computes a refraction correction using the corrected Berman model. The refraction correction automatically defaults to off in modes using azimuth–elevation inputs and turns on in modes using right ascension–declination coordinates. The defaults may be overridden if desired.

WINDAVG — Calculates the average wind based on the last 2 minutes. If the average wind exceeds 45 mph or if a gust exceeds 55 mph, the antenna will automatically stow even if off or unattended.

The CPU 2 level 2 software is shown in Fig. 7. The background routines perform the following functions:

SPEEDSELECT — Selects either high or low speed mode, depending on the angular distance to the desired point. An azimuth distance greater than 10 degrees or an elevation distance greater than 3 degrees will select high speed in that channel. If the channel is already tracking in low speed, the speed change algorithm will be performed to switch the channel to high speed. Similarly, an azimuth distance less than 10 degrees or an elevation distance less than 3 degrees will cause the speed change algorithm to switch to low speed. To prevent excessive overshoot when approaching a target in high speed, the maximum high speed rate is reduced to 0.5 degree/second if the distance to the target is less than 5 degrees.

CONVERT — Inputs the control room monitor channels and converts all channels (including those from the Data Collection Controller) to floating point and ASCII. Each sensor has a minimum and maximum value limit that may change depending on the current state. This module also checks sensor limits and writes data to the DATAOUT buffers. If monitor action is required due to a sensor out-of-limit condition, flags are set for the TASKREQ module.

MONACT — Performs the actual tasks selected by the logic in TASKREQ as a result of a monitor action.

TASKREQ — Selects the highest priority task and initiates required action. If it is a multiple-state task, the next state is initiated when ready.

TASKACT — Performs the actual tasks selected by the logic in TASKREQ as the result of a monitor alert, operator, or configuration input.

DATAOUT — Outputs data from the screen, star switch, or printer buffers.

Four interrupt routines handle the receipt of a character from the Data Collection Controller, a star switch message, an operator keyboard character, or a star switch message time out.

Five help menus are available at the local display, the station controller display, or the remote display in the NOCC to provide the operator with assistance in remembering the commands available and their syntax. All operator and monitor events are logged on the printer as well as reported through the star switch for eventual recording on magnetic media. In addition, any sensor data may be selected for logging.

V. Discussion and Conclusions

Many lessons were learned configuring a maximum loaded 8080-based system with multiple processors. Difficulty was encountered in debugging problems relating to critical timing relationships between the CPUs. Available logic analyzers lacked the capability to trigger on combined events in each CPU and back-trace either the CPU or the bus.

The inability of the 8080 (and of most other CPUs, including 16-bit devices) to perform true bit operations greatly slows down the transfer of digital monitor and control data. The least addressable element is a byte. Therefore, changing a single bit in memory, an input–output port, or an interrupt mask requires reading the byte, changing the desired bit with a logical AND, OR, or XOR operation, and storing the byte. If more than one CPU or an interrupt routine attempts to alter a bit in the same byte, one of the changes may not be made. The Z80 CPU does allow bit operations and should be considered if one decides to use an 8-bit processor for monitor and control applications.

All software was developed in the controllers or in similar microprocessor computers. While this proved convenient from the standpoint of always having an editor or a compiler available, the time required for making changes in several modules, compiling, relinking, and listing was typically 4 hours. However, most of the debugging and patching was performed at the machine code level. One of the advantages of PLM is that the source statements are easily traceable and alterable at the machine code level. An extremely useful feature of the program was a built-in simulation of the servo loop and angle readouts. This allowed much of the debugging to be done at JPL rather than at the station.

Several undesirable features of the AMC-95/6011 floating point processors were encountered and should be given consideration when choosing components for future systems. Stack-oriented processors cannot be shared by multiple users (such as interrupt routines or other CPUs) if their state cannot be restored to the state at the time of an interrupt. The capability to save the processor stack and to restore it existed, but the status register could not be restored. It would also be useful in many applications to have access to the remainder following a fixed point division for modulo operations. Another annoying feature of the floating point processors was that if given operands were out of range, an unpredictable result was generated rather than a closest approximation. In the refraction algorithm, for example, it is necessary to evaluate small powers of *e*. If the argument is too close to zero, a very large result is obtained instead of unity. In time critical control applications, where it is not possible to check the range of all operands before computation and it is too time consuming to execute floating point error handling routines, a closest approximation would be desirable. It is unacceptable to simply report a range error and halt.

The PLM compiler available for the 8080 does not support floating point directly. Operands must be pushed onto the floating point processor stack, an operation code supplied, and the result taken off the stack and stored. The execution time for getting numbers in and out of the stack is longer than some of the floating point operations themselves. Using the floating point stack, some calculations can be chained to minimize this overhead, but clever manipulation of a stack is one of the tasks that one hopes to avoid by using a high-level language in the first place. The PLM-86 compiler for the 8086 CPU using an on-board floating point processor avoids this and allows floating point operations to be written in algebraic form.

While the objectives of this task have been achieved, it is clear that this represents a data point on the maximum capability of an 8-bit microprocessor-based system. Limits of memory size, execution speed, module space, and general complexity were encountered. All of these limits could be corrected by use of one of the more powerful 16-bit processors that have become available since the start of this project. The software could be transferred with minor modifications to a 16-bit processor (8086) by converting from PLM-80 to PLM-86. This would provide execution time and memory space for inclusion of additional features such as CONSCAN, table of offsets, and library of frequently tracked sources.

Reference

[1] R. M. Gosline, "DSS 13 Microprocessor Antenna Controller," *TDA Progress Report 42-77*, vol. January–March 1984, Jet Propulsion Laboratory, Pasadena, California, pp. 64–74, May 15, 1984.

Table 1. Acceleration and deceleration limits

Variable	Low speed, deg/s/s	High speed, deg/s/s
Azimuth acceleration	0.005	0.045
Azimuth deceleration	0.016	0.075
Elevation acceleration	0.00125	0.04
Elevation deceleration	0.002	0.075

Table 2. Analog sensors

Identifier	Description	Units
A00	TACH AZ LOW SPEED 1	V
A01	TACH AZ HIGH SPEED 1	V
A02	TACH AZ LOW SPEED 3	V
A03	TACH AZ HIGH SPEED 3	V
A04	TACH EL LOW SPEED	V
A05	TACH EL HIGH SPEED	V
A06	PUMP 125HP LEFT	PSI
A07	PUMP 125HP RIGHT	PSI
A08	PUMP 75HP LEFT	PSI
A09	PUMP 75HP RIGHT	PSI
A10	RCVR AGC 1	V
A11	RCVR AGC 2	V
A12	RCVR NAR	V
A13	ATMOSPHERIC PRESSURE	mBR
A14	OUTSIDE TEMPERATURE	C
A15	RELATIVE HUMIDITY	%
A32	EL LS MOTR DIF PRESS	PSI
A33	EL HS MOTR DIF PRESS	PSI
A34	EL SYSTEM PRESS	PSI
A35	EL HS PRESS	PSI
A36	EL LS PRESS	PSI
A37	AZ HS MOTR DIF PRESS	PSI
A38	AZ LS MOTR DIF PRESS	PSI
A39	AZ HS SYSTEM PRESS	PSI
A40	AZ LS SYSTEM PRESS	PSI
A41	WIND SPEED SW TOWER	MPH
A42	WIND SPEED SE TOWER	MPH
A43	WIND AZ SW TOWER	DEG
A44	WIND AZ SE TOWER	DEG
A45	PUMP VOL 75HP LEFT	GPM
A46	PUMP VOL 75HP RIGHT	GPM
A47	PUMP VOL 125HP LEFT	GPM
A48	PUMP VOL 125HP RIGHT	GPM
A49	AZ LS ACCUM	PSI
A50	AZ HS ACCUM 1	PSI
A51	AZ HS ACCUM 2	PSI
A52	EL SYSTEM ACCUM 1	PSI
A53	EL SYSTEM ACCUM 2	PSI
A54	FLUID LEVEL	V
A55	LS CW CNTRBAL PRESS	PSI
A56	LS CCW CNTRBAL PRESS	PSI
A57	HS CCW CNTRBAL PRESS	PSI
A58	HS CW CNTRBAL PRESS	PSI
AWS	AVERAGE WIND SPEED	MPH

Table 3. Digital sensors

Identifier	Description	States	
D00	RIGHT WRAP UP	IN	OUT
D01	LEFT WRAP UP	IN	OUT
D02	AZ BRAKE 1	SET	REL
D03	AZ BRAKE 3	SET	REL
D04	EL BRAKE LEFT	SET	REL
D05	EL BRAKE RIGHT	SET	REL
D06	AZ SPEED	HIGH	LOW
D07	EL SPEED	HIGH	LOW
D08	DISABLE SWITCH	DISA	CLR
D09	PS 28VDC 1	ON	OFF
D10	PS 28VDC 2	ON	OFF
D11	AZ PRELIMIT	IN	OUT
D12	EL PRELIMIT	IN	OUT
D13	PS 28VDC ALL TIME	ON	OFF
D14	RCVR LOCK	IN	OUT
D15	REMOTE CONTROL IND	ON	OFF
D16	PUMP HP	125	75
D17	FINAL LIMIT BYPASS	SAFE	BAD
D32	PS 28VDC XDCR	ON	OFF
D33	H FLUID TEMP	SAFE	HIGH
D34	EL LUBE PRESS AND AC	ON	OFF
D35	PUMP 75HP START RLY	ON	OFF
D36	PUMP 125HP START RLY	ON	OFF
D37	PS 28VDC HYDROMECH	ON	OFF
D38	PS WIND TOWER	ON	OFF
D39	WRAP UP TENSION	SAFE	BAD

Table 4. Angle data

Identifier	Description
N00	AZ ANGLE POSITION
N01	EL ANGLE POSITION
N02	AZ ANGLE COMMAND
N03	EL ANGLE COMMAND
N04	AZ POSITION ERROR
N05	EL POSITION ERROR
N06	AZ RATE COMMAND
N07	EL RATE COMMAND
N08	AZ RATE ACTUAL
N09	EL RATE ACTUAL
N10	AZ TOTAL OFFSET
N11	EL TOTAL OFFSET
N12	HA TOTAL OFFSET
N13	DEC TOTAL OFFSET
N14	REFRACTION
N15	AZ DIRECT OFFSET
N16	EL DIRECT OFFSET
N17	HA DIRECT OFFSET
N18	DEC DIRECT OFFSET
N19	AZ AZ/EL OFFSET
N20	EL AZ/EL OFFSET
N21	AZ RATE OFFSET
N22	EL RATE OFFSET
N23	AZ ANGLE R/O DATA
N24	EL ANGLE R/O DATA

Table 5. Servo equation constants

Speed/type	Azimuth		Elevation	
	K1	K2	K1	K2
High speed – type I	0.600	0.200	0.650	0.300
High speed – type II	0.850	0.100	0.850	0.200
Low speed – type I	0.686	0.314	0.686	0.314
Low speed – type II	0.900	0.197	0.850	0.150

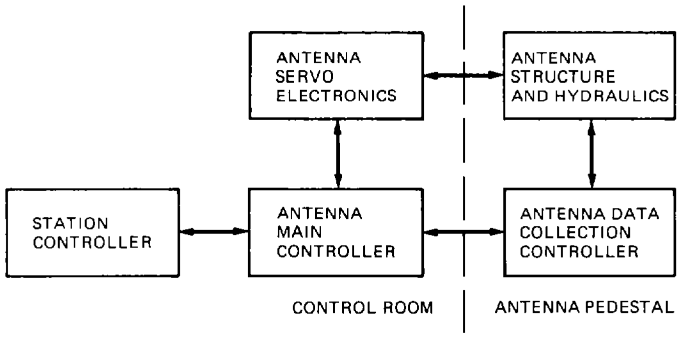


Fig. 1. DSS 13 antenna controller functional block diagram

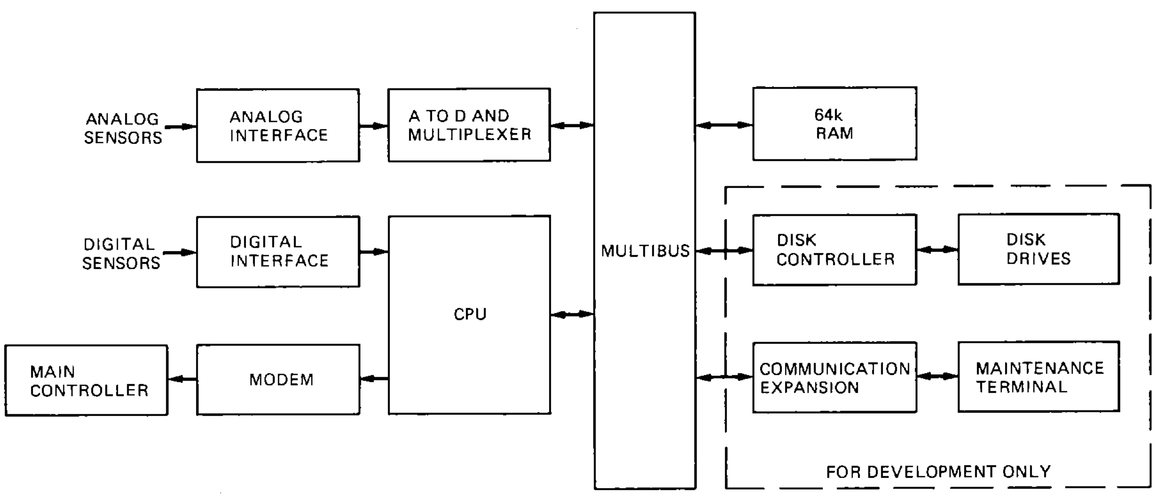


Fig. 2. Data collection controller configuration

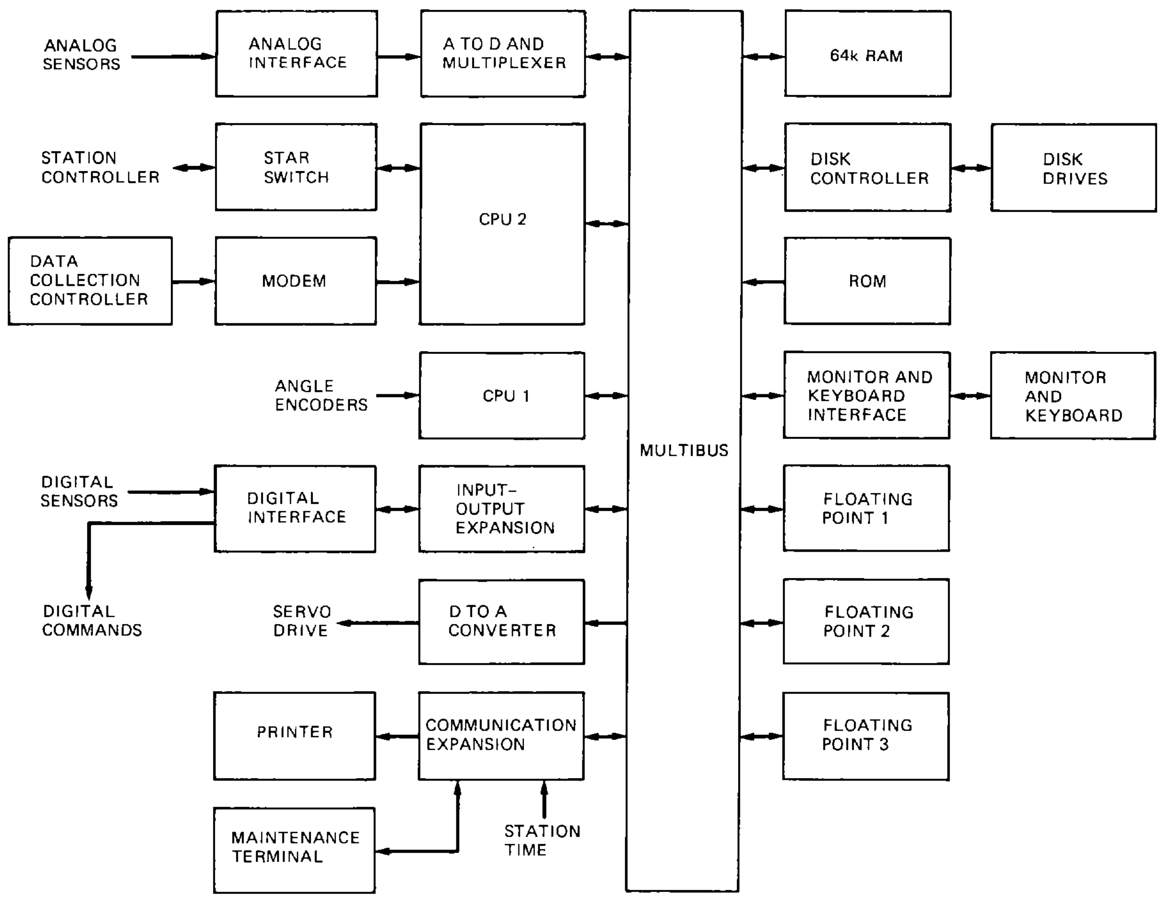
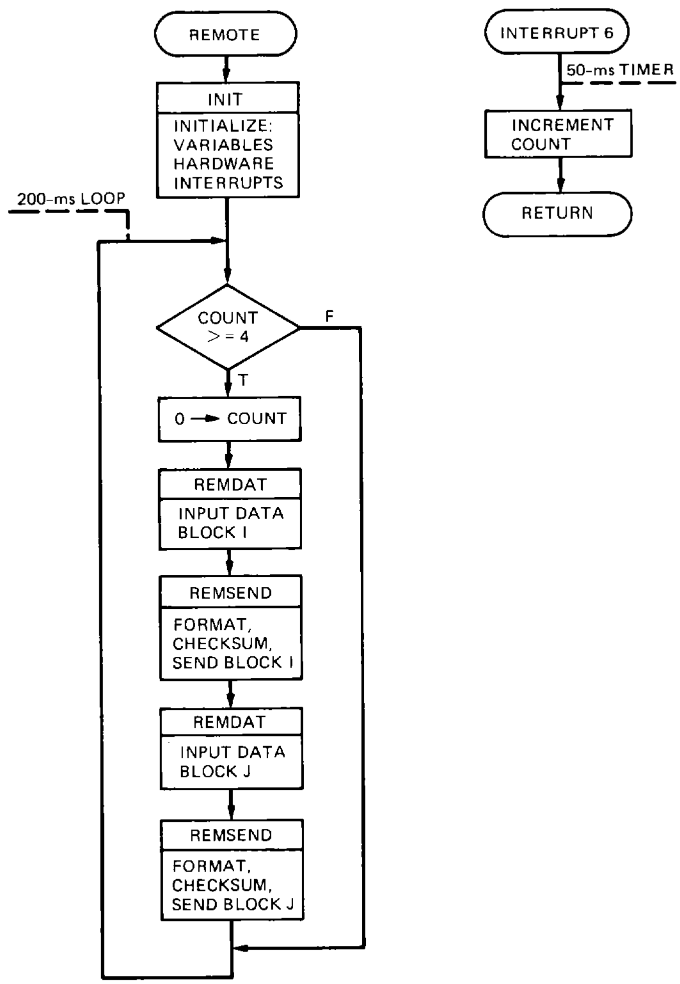


Fig. 3. Main controller configuration



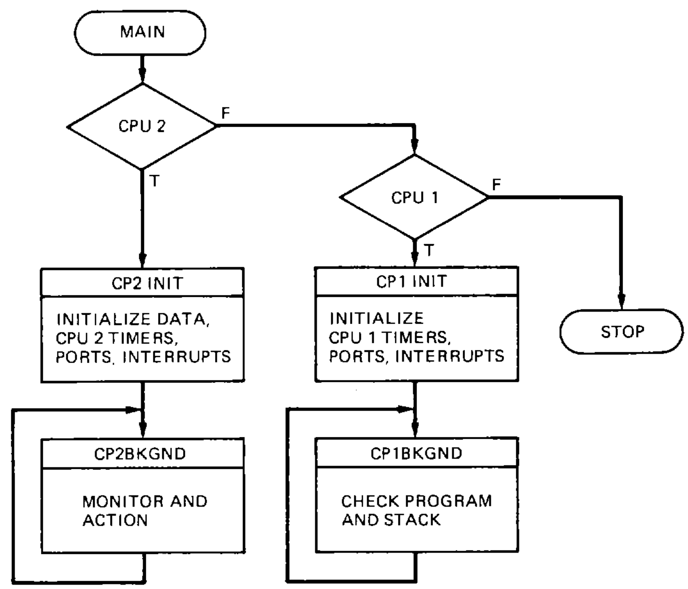


Fig. 5. Main controller top-level software

Fig. 4. Data collection controller top-level software

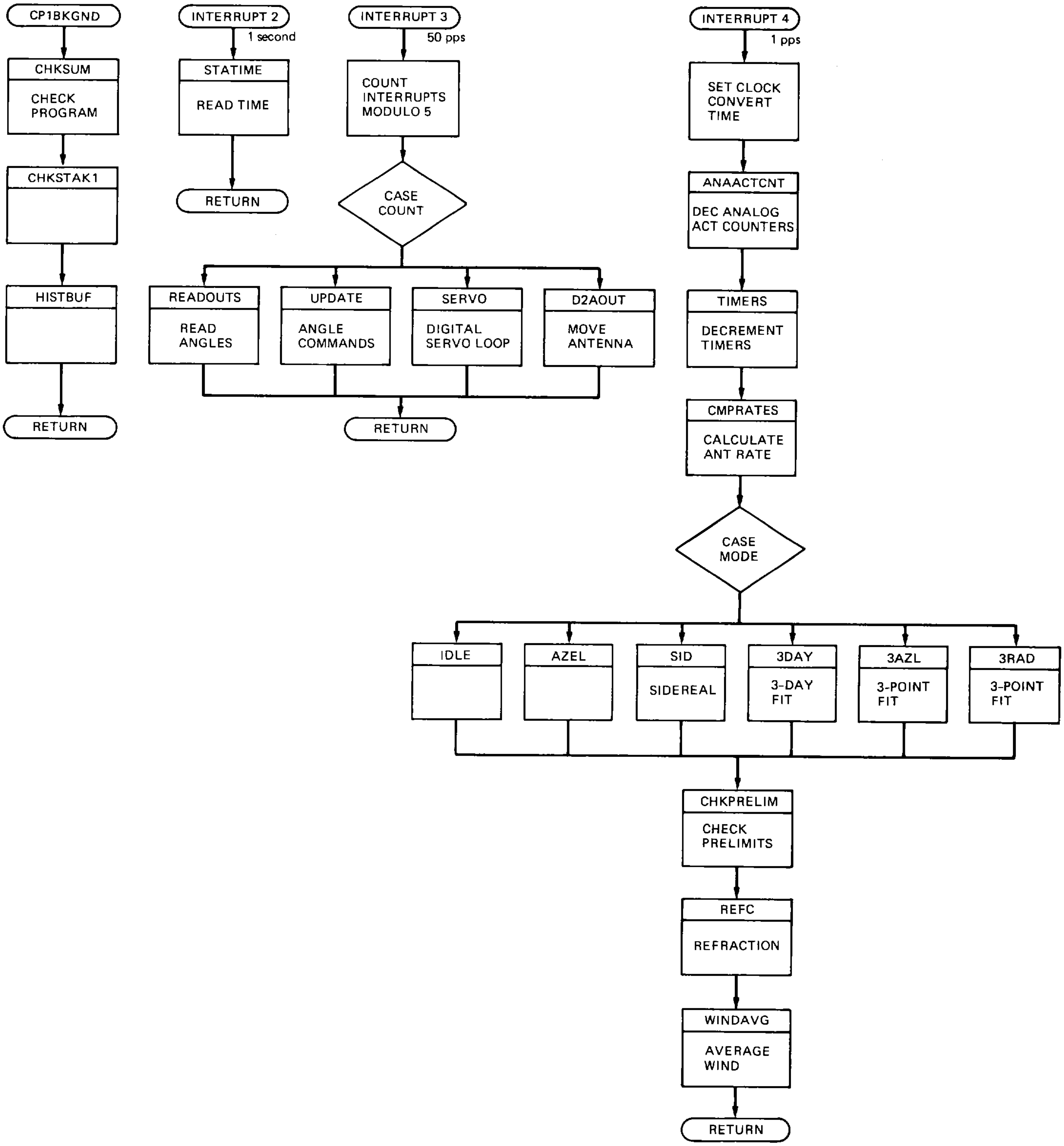


Fig. 6. CPU 1 level 2 software

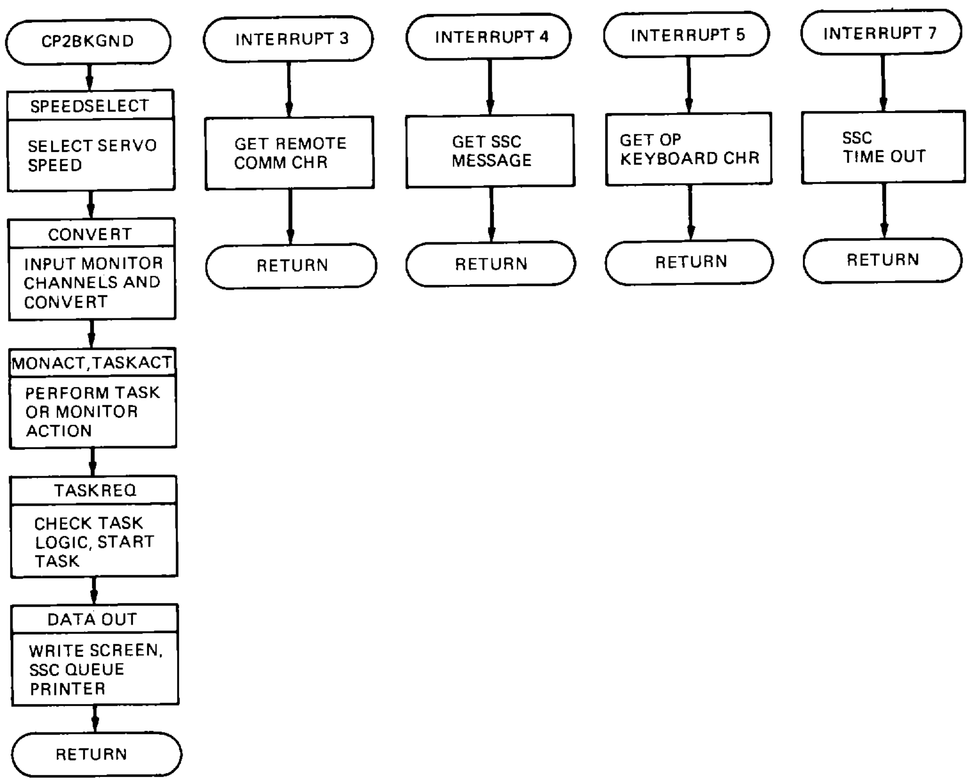


Fig. 7. CPU 2 level 2 software