DSS 13 Automated Antenna Pointing Subsystem Phase 1 Hardware

M. W. Sievers
Communications Systems Research Section

An automated antenna pointing subsystem (APS) is being installed at DSS 13 as part of the unattended station project. The function of the APS is to track an instructed position with the 26-meter antenna and to monitor the antenna servo systenr and meteorological conditions to ensure a proper operational environment. This article will discuss the now-completed first phase of the digital hardware portion of the APS development.

I. Introduction

An antenna pointing subsystem (APS) is being designed as part of the unattended station project now in progress at DSS 13. The function of the APS is to provide control and monitoring of the 26-meter antenna. The development has been divided into three phases:

Phase 1: Design and install hardware and software required to slew the antenna to a given source, to track that source, and to provide the minimum set of monitoring functions required by the tracking function.

Phase 2: Design and install hardware and software to monitor the antenna environment to ensure safe unattended operation.

Phase 3: Refine servo models used in phase 1 and design and install hardware and software to connect the APS to the station controller.

Phase 1 has now been completed, and the monitor hardware for phase 2 has been installed. This article will concentrate on the phase 1 hardware system.

II. Phase 1 Functional Requirements

Phase 1 functions can be classified into two categories: tracking, and the monitor and display needed to support tracking.

The tracking function requires that there be a means to move the antenna to a selected source and to track that source as a function of time. Specifically, this function is composed of the following subfunctions:

(1) Selection of source to be tracked

(2) Calculation of the azimuth and elevation positions of this source as a function of time

(3) Determination of the present antenna azimuth and elevation and calculation of an error signal

(4) Moving the antenna to correct for the error calculated in (3)

(5) Determination of when the antenna is pointed at the source

(6) The particular configuration of the DSS 13 antenna requires:

(a) Ability to set and reset azimuth and elevation brakes, to select azimuth and elevation high or low speed, and to turn on the hydraulic system

(b) Ability to sound a warning prior to moving the antenna

(c) Provide a signal to the servo system that indicates whether the digital system is functional.

The monitor and display requirements for phase 1 are:

(1) Antenna wrap-up and pre-limit indicators

(2) Brake and speed monitors

(3) Indicator that APS has control of antenna

(4) Warning displays

(5) Current antenna position and tracking error display

III. APS Phase Hardware Structure

A system that satisfies the requirements of Section II is shown in Fig. 1. At the heart of the phase 1 hardware is a Modcomp II minicomputer configured with a floating point arithmetic unit, a moving head disk, a terminal device, a quad Standard Interface Adaptor (SIA), and a dual digital-to-analog converter (DAC). A multiplexer-demultiplexer (MUX-DEMUX) connects the minicomputer to a display device interface, an antenna angle encoder interface, and the servo system monitors and control interface via one of the Standard Interface Adaptors. The instructed position is input through the terminal device and the dual DACs provide the azimuth and elevation rate commands which are amplified to drive the antenna. MUX-DEMUX details may be found in Section IV, and the interface descriptions are found in Sections V through VIII.

Universal time (UT) is produced by the UT generator in a 30-bit BCD format. In addition to time, the UT generator also supplies pulse trains of one pulse per second (1 PPS) and 50 pulses per second (50 PPS). Both the 1-PPS and the 50-PPS signals are synchronized with the units of seconds. These signals are provided as interrupts for the APS software (Ref. 1).

IV. Multiplexer-Demultiplexer

The purpose of the multiplexer-demultiplexer is to connect several device interfaces to the APS minicomputer via a single Standard Interface. The front end of this device is a Standard Interface compatible module (Fig. 2). Through a simple gating arrangement, the Standard Interface signals are buffered and made to look like a bus configuration. Device interfaces connect to this bus and communicate with the minicomputer at its request. Control information on the bus is derived from the interface function bits $\overline{\mathrm{F}}_0$ and $\overline{\mathrm{F}}_1$ as well as the data bits. Device interfaces are initialized by a simple 74121 power-up reset circuit.

When not outputting to the minicomputer, the MUX-DEMUX is always prepared to accept command information and data from it. MUX-DEMUX status information may be jammed into the minicomputer at any time, but a data transmission must be requested. The MUX-DEMUX always acknowledges a command by sending the minicomputer its status.

A watchdog timer, which is part of the servo control hardware (see Section V), must periodically be refreshed by the computer to indicate that the program is intact and executing. Should this timer fail, the MUX-DEMUX will send the computer its status. Since the act of sending status causes a computer interrupt, the action on watchdog timer expiration is performed in the hope that sufficient information is present for the program to respond to the interrupt and attempt recovery.

Referring to Fig. 2, the minicomputer signals its desire to send information to the MUX-DEMUX by asserting $\overline{\mathrm{STC}}$. This signal is received and filtered by a 75182 integrated circuit (IC) chip. The 75182 output gates on the response signal $\overline{\mathrm{RSP}}$ and is inverted to inhibit the MUX-DEMUX data ready ($\overline{\mathrm{RDY}}$) and function code ($\overline{\mathrm{F}}_1\overline{\mathrm{F}}_0$) 5438 drivers and the data ($\overline{\mathrm{D}}_0$ - $\overline{\mathrm{D}}_7$) 75138 drivers. Additionally, the 75182 output enables one input of a 7400 NAND gate; the other input of this gate is connected to the 7404 that receives the $\overline{\mathrm{RDY}}$ signal.

After the minicomputer receives the $\overline{\mathrm{RSP}}$ signal, it places data on $\overline{\mathrm{D}}_0$ - $\overline{\mathrm{D}}_7$, the function code on $\overline{\mathrm{F}}_1\overline{\mathrm{F}}_0$, and asserts $\overline{\mathrm{RDY}}$. When $\overline{\mathrm{RDY}}$ is received by the 7400 discussed above, that gate's output goes low. This signal is inverted by a 5438 and filtered by a 75182.

The output of the 75182 gates on a 7411 AND and starts another 5438-75182 time delay circuit. This delay eventually turns the 7411 off and disables the $\overline{\mathrm{RSP}}$ signal. The 7411 output is a strobe signal that indicates that the data and function code being sent are stable.

The minicomputer generates a 11 function code when it is sending the MUX-DEMUX a command byte. A 7408 AND detects the 11 function code, and its output is gated with the strobe from the 7411 to enable a 74S138 decoder and set a status jam flip-flop. (Note that this flip-flop is also set by the time-out signal from the watchdog timer.) Three data bits at the input of the 74S138 select one of eight interfaces to which the minicomputer is sending a command. The five remaining bits are interpreted as command information by the selected interface.

After the minicomputer releases $\overline{STC}$, to acknowledge the command just received, the status jam flip-flop output is gated to assert $\overline{STD}$ with a 11 function code. The minicomputer responds to the asserted $\overline{STD}$ by signalling $\overline{RSP}$. In the MUX-DEMUX, $\overline{RSP}$ is gated and filtered through the same circuitry that gated and filtered $\overline{RDY}$ in the discussion above. This time, however, no strobe is issued and the MUX-DEMUX places its status on the $\overline{D}_0$ - $\overline{D}_7$ data lines and signals $\overline{RDY}$. $\overline{RDY}$, the function codes, and data are held asserted as long as the minicomputer is asserting $\overline{RSP}$.

When the minicomputer sends a full word to the MUX-DEMUX, the function code bits determine whether the Data Acquisition Interface or the Plasma Display Interface is to receive the data word. A pair of 7411s gate a strobe to the Data Acquisition Interface if $\overline{F}_1 F_0$ is true or to the Plasma Display Interface if $F_1 \overline{F}_0$ is true. Note that when a full word is being sent, all bits may be interpreted by the receiving interface and that the status jam flip-flop is not set.

If the command byte or full word sent to the MUX-DEMUX is a request for data from the selected device, that device issues a data ready signal $\overline{DTARDY}$ when it has data available and places its data on the output data bus. The $\overline{DTARDY}$ signal is latched in the data ready flip-flop, the output of which is gated to produce an $\overline{STD}$ signal and also enables the output data bus onto $\overline{D}_0$ - $\overline{D}_7$ by bringing the 75138 strobe lines low. The handshake process mentioned in the status byte jam discussion is started to transfer data bytes to the minicomputer, with the following differences:

(1) Data bytes are transferred to the minicomputer as long as the data ready flip-flop is set. It is the responsibility of the sending interface to clear the flip-flop when it has transferred all of its data by issuing $\overline{DONE}$.

(2) The status jam flip-flop being false enables the trailing edge of the $\overline{RSP}$ signal to fire a 74123 one-shot. The one-shot output, labeled NXTBYT, is a signal to the sending interface to place its next data byte onto the output data bus if one is available.

(3) The function code is 00.

Each interface capable of sending data to the minicomputer has an established number of bytes that it will transfer. Since the minicomputer has a priori knowledge of that number, it can prepare its input hardware accordingly.

V. Servo Control Hardware Interface

The Servo Control Hardware Interface connects the minicomputer to azimuth and elevation speed and brake control relays, the hydraulic pump starter, and the warning horn relay. Additionally, it provides the servo control system with an indication that the APS software and hardware are functional, i.e., the watchdog timer. The minicomputer controls these functions via command bytes.

In Fig. 3, brake and speed control information is stored in a 74175 register. This register is clocked by a signal from the 74S138 interface selection decoder in the MUX-DEMUX. Four data bits on the input data bus are clocked into the 74175 and buffered by 75453s to drive brake and speed control relays. High speed and brakes released is indicated by a low signal at the 75453 outputs.

A 74S138 decoder in the Servo Control Interface decodes command bytes from the minicomputer to refresh the watchdog timer, turn the hydraulic pumps on or off, and sound the warning horn. The watchdog timer consists of a 74123 retriggerable one-shot with a one second delay and a 75452 driver. The driver pulls in a relay in the servo control system. As long as the relay is held on by the timer, the minicomputer has control of the servo system. Should the watchdog time out, the relay is dropped out and a time out signal from the watchdog timer is issued to the MUX-DEMUX. Once the relay has been turned off, the minicomputer no longer has control of the servo system and reset must be done manually.

The hydraulic pumps are controlled by a 74S112 flip-flop and 75453 driver. When the driver output is low, the pumps are on, and off otherwise.

Finally, the minicomputer can actuate a warning horn. The horn circuitry consists of a 555 oscillator, a 74191 counter, a 74S112 flip-flop and a 75453 driver. A single command byte from the minicomputer sets the 74S112 and enables the 555. Each pulse from the 555 sounds the horn and increments the 74191. When the horn has been sounded three times, the 74S112 is cleared and the 555 disabled. The horn on time is approximately one second with an on-to-off ratio of about 2:1.

VI. Plasma Display Interface

A Burroughs Self-Scan plasma display panel was made part of the APS so messages could be quickly displayed. Interface circuitry (Fig. 4) provides control signals for cursor manipulation, character input-output, and screen clear. The interested reader should consult a Burroughs Self-Scan manual for a signal glossary and timing diagrams.

Command byte information is used for cursor operations and screen clear. Data bytes are separated into cursor pre-load addresses and characters. This is done by examining the most significant bit of a data word. If the bit is set, the circuitry generates signals to pre-set the cursor. If not set, a write command is issued. This permits mixing cursor addresses and characters in a single transmission.

VII. Angle Encoder Interface

The angle encoder interface (Fig. 5) consists of registers to hold the azimuth and elevation data words and circuitry to send these coordinates to the MUX-DEMUX in 8-bit bytes.

The angle encoder is made by Northern Precision Laboratory and generates a 20-bit azimuth and a 20-bit elevation word. The encoder produces a data ready signal when it has data available. The data ready signal is filtered and used to strobe a set of 74174 registers when these registers are not being read by the minicomputer.

A 74109 flip-flop is set when the minicomputer is reading angle encoder data from the 74174s. This flip-flop output disables an input of a 7408 AND, thus preventing the data ready from clocking the 74174 registers. Data is placed on the output data bus by 74253 tristate data selectors. A 74193 counter is decoded to select the byte to be placed on the bus. Each time NXTBYT is received by the 74193, a new byte is placed on the bus. When all bytes have been transferred to the minicomputer, the decoder issues the DONE signal.

VIII. Data Acquisition Interface

The Data Acquisition Interface was designed to control a Data Acquisition Device. This device, shown in Fig. 6, is a 64 differential channel analog-to-digital converter. The device is made of Burr-Brown components: SPM 851 eight differential channel, 12-bit analog-to-digital (A-D) converter, MXP 321 and MXP 320 analog multiplexers, and 74365 tristate buffers.

The MXP 321 contains a binary counter which may be incremented by a strobe signal or randomly loaded by a strobe signal with load enable asserted. The output of this counter selects the analog channel to be sampled. The MXP 320 contains an analog multiplexer only.

In the SDM 851, another differential multiplexer selects analog data from one of the MXP 320 or the MXP 321. It also contains all circuitry necessary to start the A-D conversion and indicate when the conversion is complete. The SDM 851 has been set up to start an A-D conversion after a new channel has been selected either by incrementing the channel address or loading a new one.

The interface that controls the Data Acquisition Device is shown in Fig. 7. A command byte sent by the minicomputer is interpreted in the interface to produce a strobe to the Data Acquisition Device. A full word sent to the interface causes it to bring the channel address load enable signal low and then issue a strobe pulse. Data sent in the full word is used for the channel address data.

The end-of-conversion signal from the SDM 851 is filtered and gated to produce the data ready signal to the MUX-DEMUX. A 74S112 flip-flop controls the select line on a pair of 74257 tristate data selectors. The NXTBYT signal clocks the 74S112 and is gated with its output to produce the DONE signal.

Data is sent to the minicomputer left adjusted. In this form, it is convenient for the using software to select either a logical or arithmetic shift to correctly align the data.

Reference

1. Sievers, M. W., "An Automated Tracking System for the ARIES Antenna," in *The Deep Space Network Progress Report 42-32*, pp. 139-150, Jet Propulsion Laboratory, Pasadena, California, April 15, 1976.

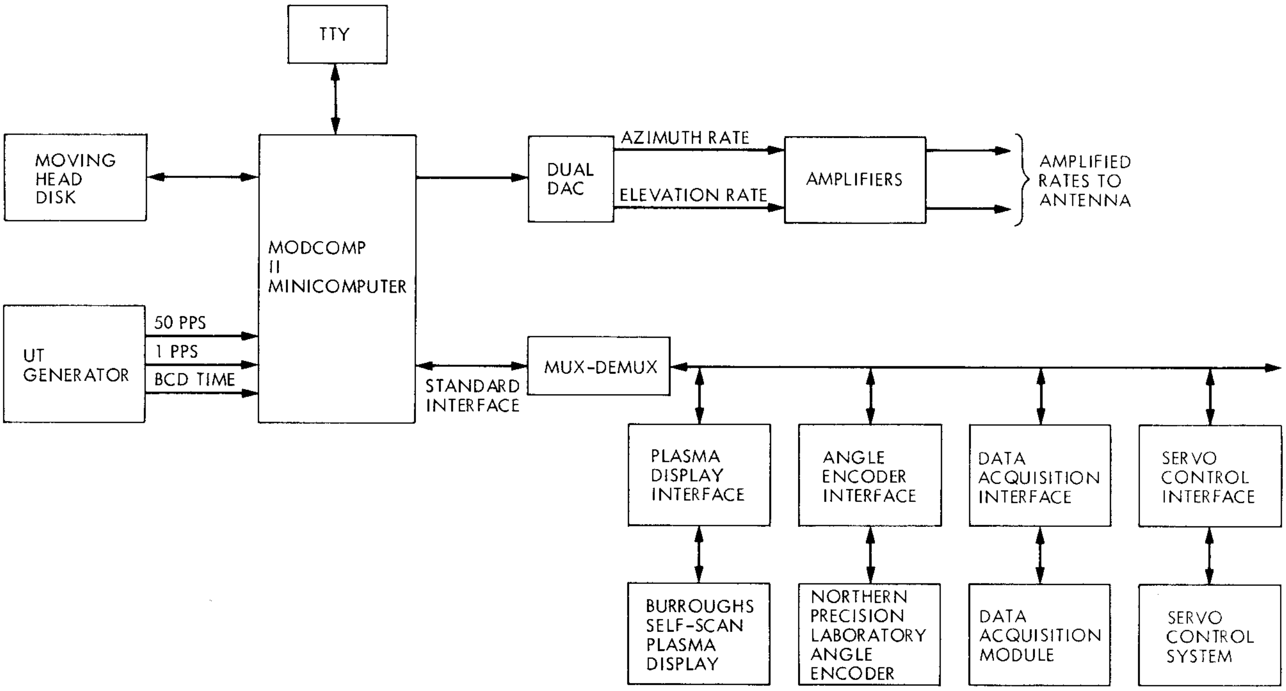

Fig. 1. APS phase 1 block diagram

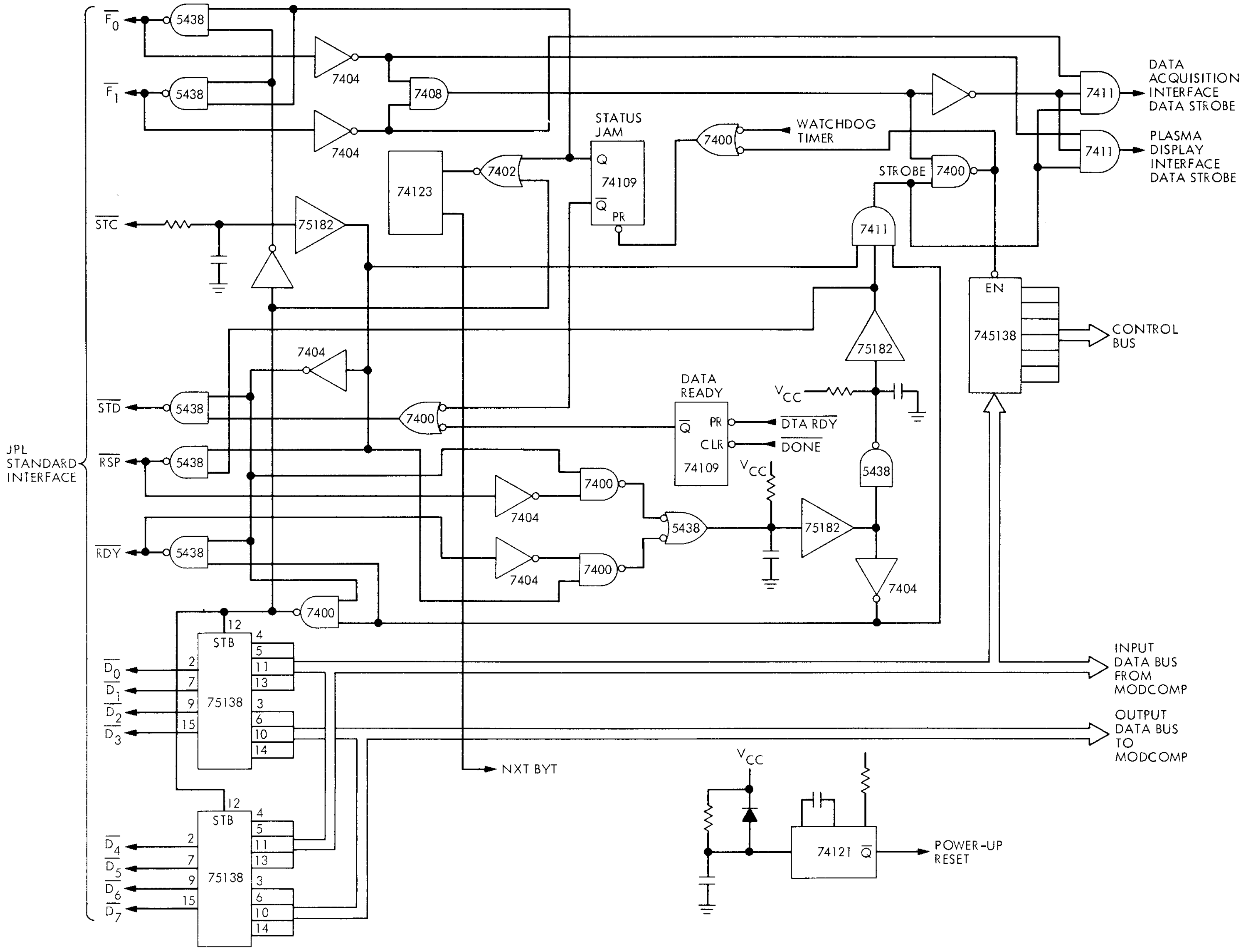

Fig. 2. Simplified MUX-DEMUX logic

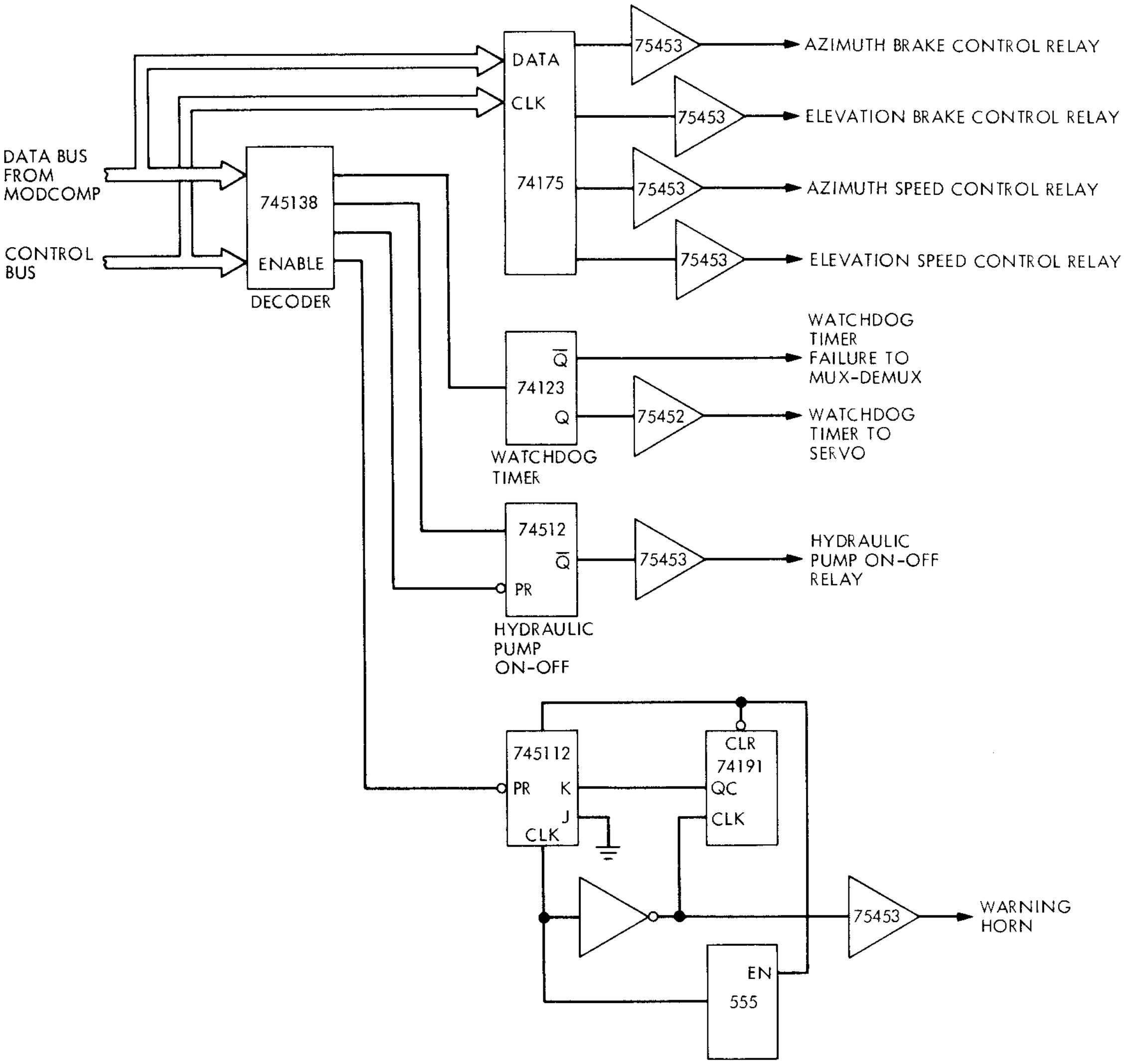

Fig. 3. Servo control interface logic

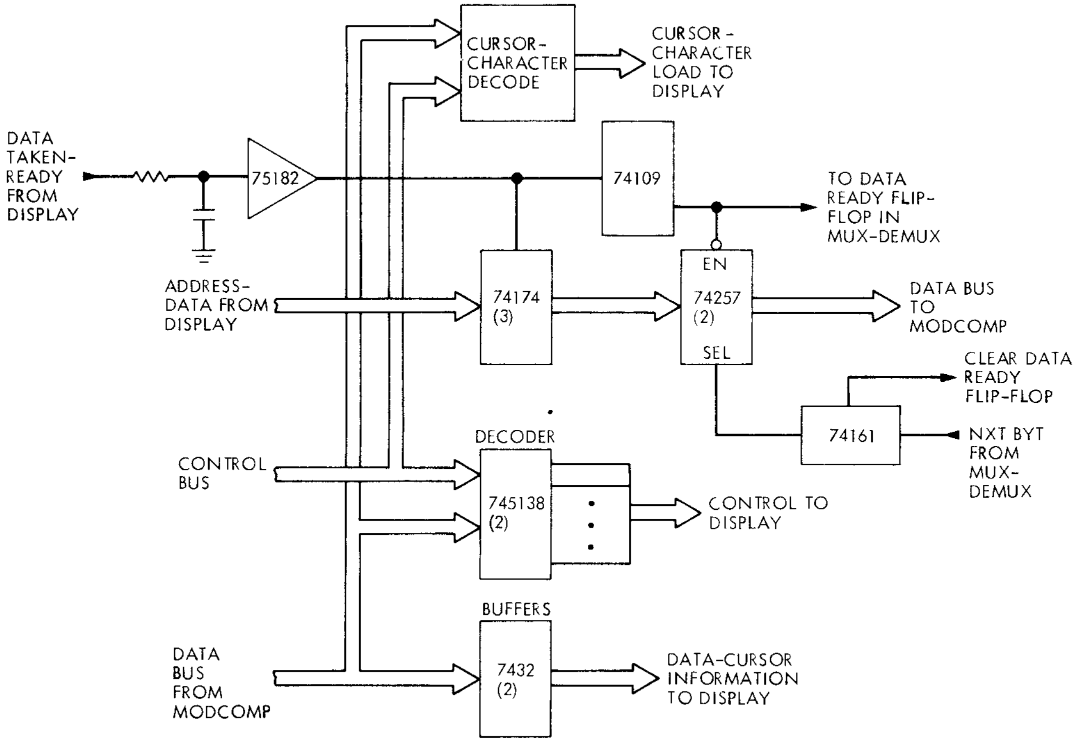

Fig. 4. Plasma display interface

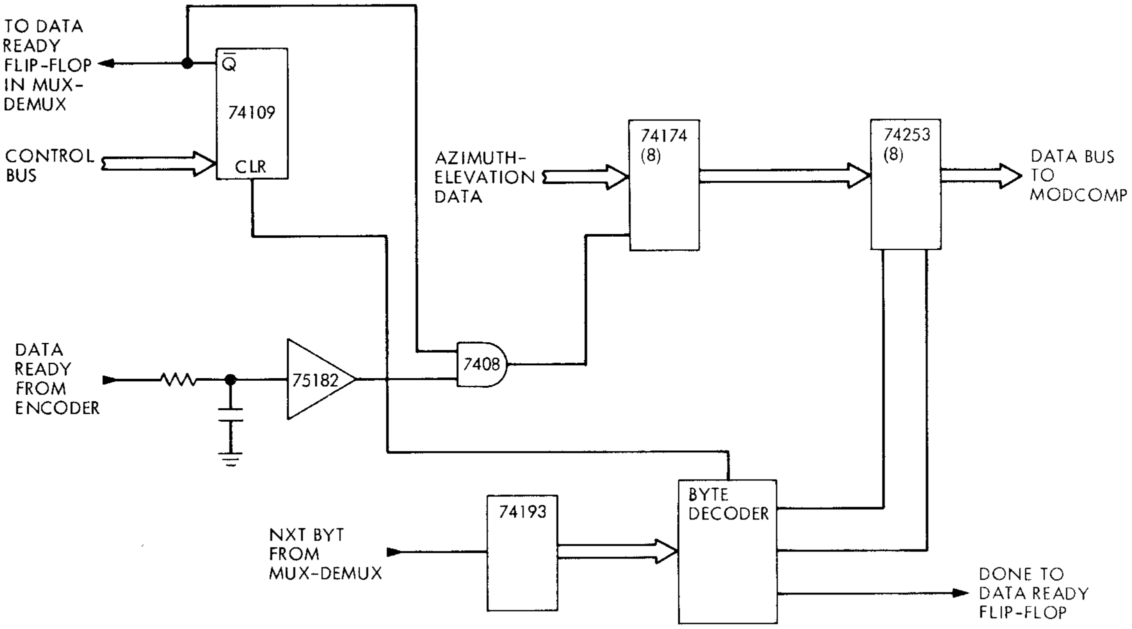

Fig. 5. Angle encoder interface

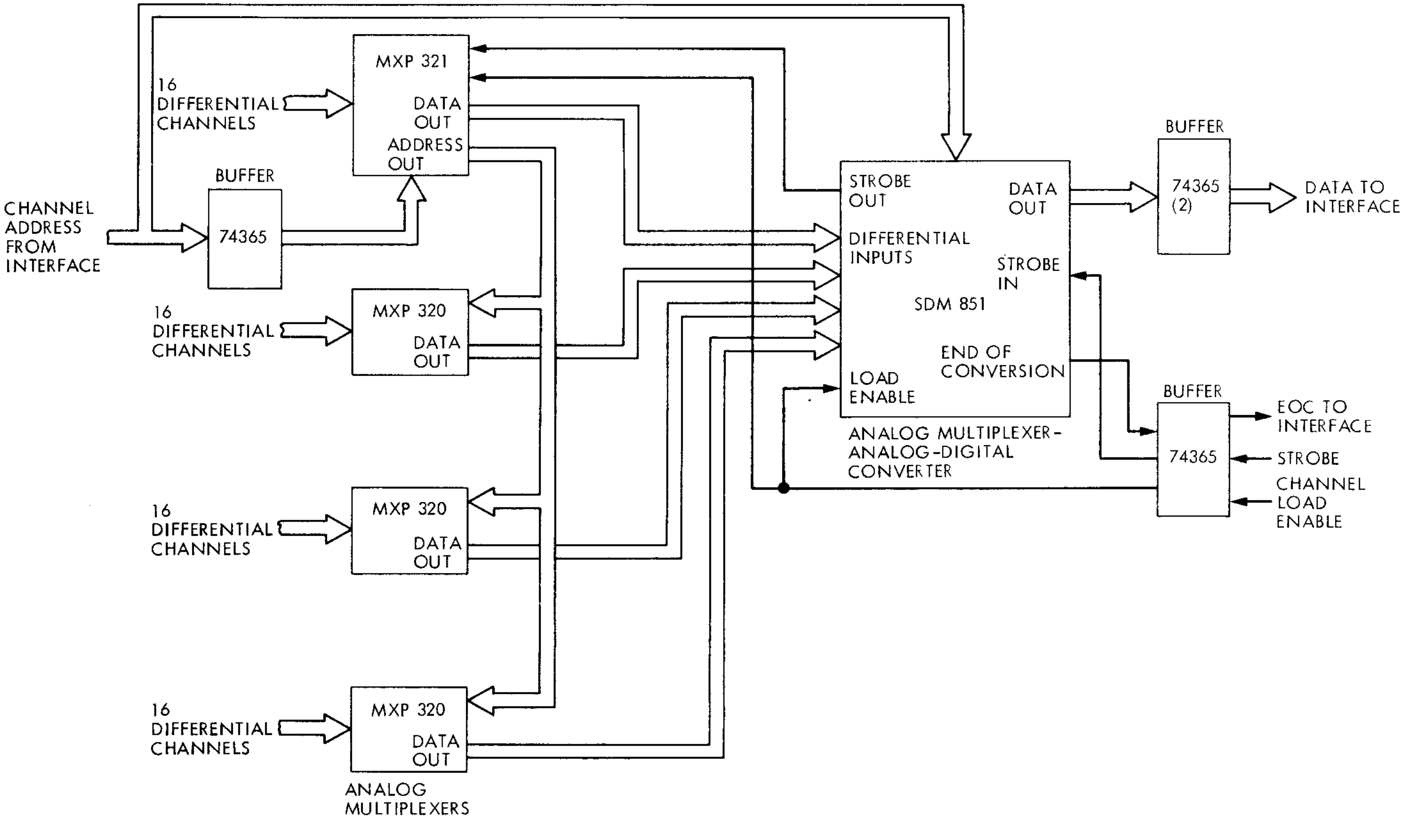

Fig. 6. Data acquisition device

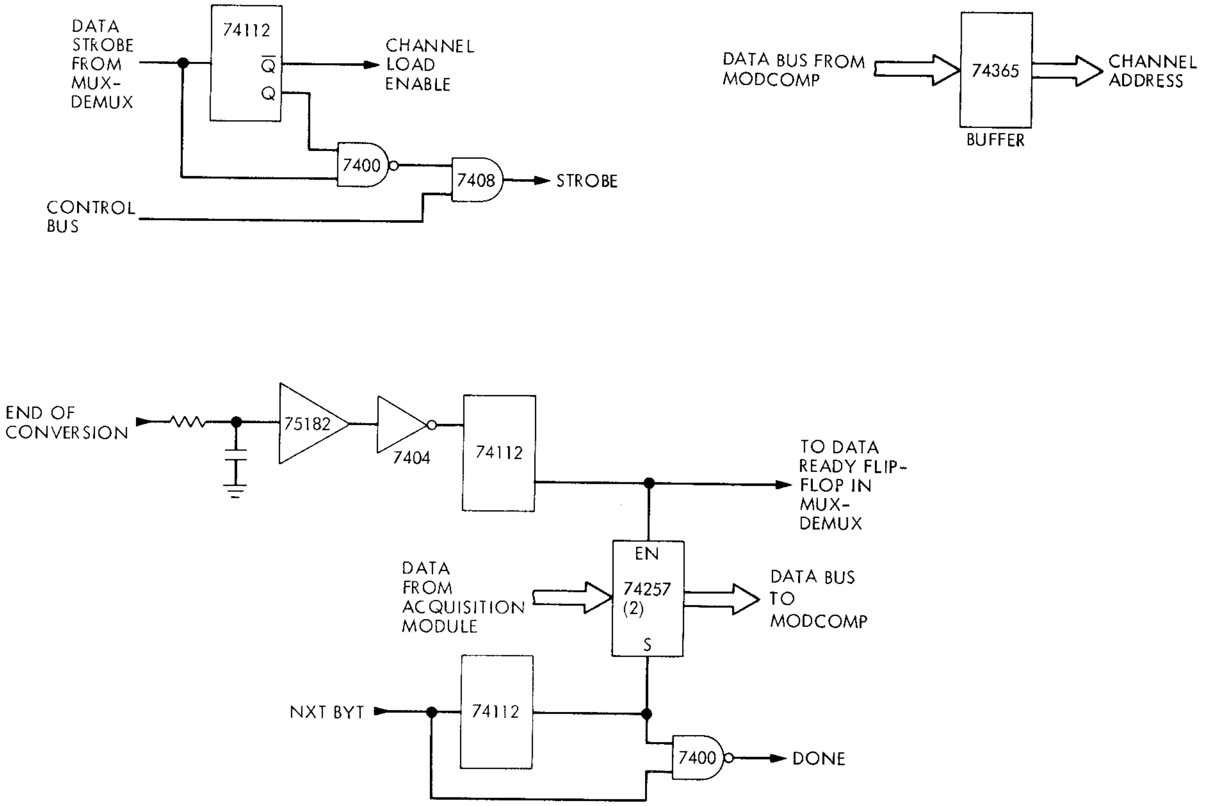

Fig. 7. Data acquisition interface