THIRD GENERATION MIL STD 1553B LSI CHIP SET

Robin D Beasley

Flight Automation Research Laboratory
Marconi Avionics Ltd
Rochester
Kent
UK

1. INTRODUCTION

Marconi Avionics Limited's experience with Mil Std 1553 commenced with the implementation of an 'A standard' remote terminal in the Head Up Display for the General Dynamics F16 by our Airborne Displays Division. The Flight Automation Research Laboratory (FARL) have subsequently completed a circuit design for the digital section of a Mil Std 1553B terminal with the LSI implementation carried out by Marconi Electronic Devices Ltd. These LSI devices are currently available through Circuit Technology Inc of New York.

This paper will review the current LSI terminal activity undertaken by FARL. This activity has used experience gained during the previous five-element 1553B LSI development as the foundation for a third generation two-element 1553 LSI terminal design.

The subsequent semiconductor implementation is a collaborative exercise between the GEC Hirst Research Centre and FARL.

2. PRE DESIGN DEFINITION

This development originated from the requirements of a number of future military projects within the United Kingdom. These requirements can be summarised as follows:

- o Nuclear Hardness
- o Improved Architecture
- o Better Bus Control Facilities
- o Long Market Life.

The main features which required definition before the detailed logic design could commence were:

- o Modes of Operation
- o Chip Set Architecture
- o Subsystem Interface Philosophy.

2.1 Modes of Operation

The chip set can operate in three modes; as a remote terminal, a controller terminal and as a passive monitor terminal. As a remote terminal it is fully compatible with Mil Std 1553B. All options and mode commands specified by the Mil Std are implemented. Full and meaningful use is made of status word bits and a comprehensive BIT word is provided. A unique mechanism has been incorporated that permits the subsystem to declare an illegal command legal and vice versa before the chip set services the command. Use of this mechanism is optional, normal operation will ensue if this option is not taken.

The previous Marconi chip set, in bus controller mode, could initiate messages on a word by word basis under subsystem control. This mechanism has been greatly improved and this chip set can initiate complete transfers and error recovery under its own control.

As a passive monitor the two-element chip set will decode all messages on the bus, carry out error checking and pass all valid words to the subsystem.

2.2 Architecture

The basic parameters which governed the chip set architecture are the system requirements equated against the semiconductor technology. This chip set is to be implemented in Silicon on Sapphire (SOS) CMOS which is being developed at the GEC Hirst Research Centre for the UK MOD specifically as a nuclear hard military process. The existance of a five-element terminal chip set (dual, standby redundant configuration) meant that four architecture options existed. A five-to-four conversion offered insignificant architectural benefits. A five-to-three conversion introduced some undesirable aspects. The five-to-two and five-to-one options both gave sound architectures but the five to one conversion exceed the integration capability of the semiconductor technology available.

The details of the chosen two-element architecture is shown in Figure 1. The first element is a Tx/Rx function which can support up to a triple standby redundant bus system with no additional logic or LSI elements. The second element is the terminal control function. Both elements will be compatible with a 48 pin DIL package, the interconnections and system interface will consist of a sixteen bit highway and discrete control lines.

2.3 Subsystem Interface Philosophy

The subsystem interface philosophy adopted by Marconi Avionics for the five-element chip set has been maintained in the two-element, and so 1553 terminal logic defined by the subsystem rather than by the Mil Std, has not been included in either chip set. Again, effort has been concentrated on producing a set of interface signals that allow a user to integrate the chip set into a system efficiently.

Experience gained by users of the previous LSI chip set has allowed the selection of a more efficient set of subsystem lines. A full definition of all the RT subsystem signals is given later in the paper.

The Tx/Rx element carries out the following functions:

- o Waveform reception
- o Message validation checks
- o Broadcast command detection
- o Reply timeout and end of transmission detection
- o Bus selection and shutdown control
- o Terminal loop test and self test
- o Partial BIT word recording
- o Partial mode command execution
- o Waveform transmission.

The terminal controller carries out the following functions:

- o Command word recording
- o Command legality checks
- o Command execution state sequencing
- o Partial status recording
- o Partial BIT word recording
- o Parital mode command execution
- o Data word count
- o RT/BC subsystem interface control
- o RT subsystem handshake failure check
- o Instruction decoding
- o Message execution state sequencing
- o Report word generation and control
- o Subsystem interrupt control
- o Automatic retry control

3. FUNCTIONAL DEFINITION

This section reviews the main design features of the chip set. The chip set is fully compliant with Mil Std 1553B and the design has been arranged such that this compliance does not require a rigorous knowledge of the Mil Std by the user, neither will it permit an invalid bus response by incorrect use of the subsystem interface lines. The only functional aspects requiring subsystem intervention are those defined by 1553B as being subsystem dependent, such as the contents of the Vector word.

3.1 Basic Characteristics

The chip set is designed to operate over the temperature range of +125°C to -55°C and has a storage range of +150°C to -65°C. A power supply of 5 volts is required. Internal power up initialisation allows the first command to be fully serviced. The active user I/O lines to the chip set will be TTL compatible.

3.2 Data Transfers/Mode Codes

The chip set can handle all types of data transfers and mode codes. The mode codes have been fully implemented and protected against incorrect T/R bit and broadcast bit. The chip set will also check that the correct number of contiguous data words are present.

This chip set includes an illegal/legal message enable/disable facility which will allow a subsystem to selectively make any valid subaddress and/or word count illegal before the chip set starts to service the command.

A terminal loop test is also included, by which a receiver monitors the output of its associated transmitter. Loop test fail will cause a transmission abort and setting of the terminal flag.

3.3 Status Words

The bits in the status word have been meaningfully utilised. The instrumentation bit, busy bit, service request bit and the broadcast command received bit are utilised as per Mil Std 1553B.

The message error bit is set if:

- o The message is too long or too short.
- o The message or words are invalid.
- o Illegal use of broadcast is made.
- o A Tx command word with contiguous data is received.
- o The subsystem sets ILLEGAL COMMAND.

The terminal flag will be set if:

- o The loop test fails.
- o The RT address parity check fails.
- o Terminal self test failure occurs.
- o A transmitter overrun occurs.

The subsystem flag will be set if:

- o The subsystem makes incorrect use of the data transfer mechanism.
- o Set by the subsystem.

3.4 BIT Word

An internal BIT word is available via the bus by use of the relevant mode code. The bits of this word have been defined as follows:

LSB
- o Tx timeout error
- o Subsystem handshake failure
- o Loop test failure
- o Illegal T/R bit
- o Illegal command
- o Word count low
- o Word count high
- o Illegal broadcast
- o Bus 0 shutdown
- o Bus 1 shutdown
- o Bus 2 shutdown
- o Terminal flag inhibited
- o Tx time out on Bus 0
- o Tx time out on Bus 1
- o Tx time out on Bus 2
- o Reserved.

4. SUBSYSTEM INTERFACE

4.1 Signal Definition

Listed below are the main subsystem interface lines available to the user for remote terminal operation. Each signal definition has the signal name, with corresponding abbreviation, the number of lines and a short functional description.

Data Highway, B0 - B15, (16 off). This is a bidirectional highway used to transfer 16 bits of data to and from the subsystem.

Buffer Enable, $\overline{\text{BUFEN}}$, (1 off). This line goes low to enable the data highway buffer between the terminal and the subsystem.

Read/Write, R/$\overline{\text{W}}$, (1 off). This line indicates the direction of information transfer between the terminal and the subsystem.

Strobe, $\overline{\text{STROBE}}$, (1 off). This information transfer strobe will pulse indicating valid data present on the data highway.

Data Transfer Request, $\overline{\text{DTRQ}}$, (1 off). This line goes low to request permission to transfer a data word to or from the subsystem.

Data Transfer Acknowledge, $\overline{\text{DTAK}}$, (1 off). This line should be driven low to grant permission to perform the requested data word transfer.

Mode Data Transfer, $\overline{\text{MDT}}$, (1 off). This line goes low to indicate that the data word being transferred is associated with a mode command.

Receive Command, $\overline{\text{RXCMD}}$, (1 off). This line goes low to indicate that a valid command word for this RT is on the data highway and should be written into the subsystem command word latch.

Status Enable, $\overline{\text{STATEN}}$, (1 off). When low this line will enable the contents of subsystem status latch onto the data highway.

Address Enable, $\overline{\text{ADEN}}$, (1 off). During terminal initialisation this line will be used to enable the terminal address onto the data highway.

In Command, $\overline{\text{INCMD}}$, (1 off). When low this line indicates that the terminal is currently servicing a command word.

Good Block Received, $\overline{\text{GBR}}$, (1 off). When a fully validated block of data has been received this line will authorise its use by the subsystem.

Mode Data Received, $\overline{\text{MDR}}$, (1 off). This line will pulse low when valid mode data has been received.

Synchronise, $\overline{\text{SYNC}}$, (1 off). This line will pulse low if a valid synchronise without data mode code is received.

Busy Request, $\overline{\text{BUSYREQ}}$, (1 off). A subsystem taking this line low will cause the chip set to set the busy bit in the status word and inhibit all data transfers to or from the subsystem.

Busy Acknowledge, $\overline{\text{BUSYACK}}$, (1 off). This line will go low to indicate that the subsystem has free access to any shared store.

Reset, $\overline{\text{RESET}}$, (1 off). This signal when low causes the internal circuitry to reset to the quiescent initialised state.

RT/BC, RT/$\overline{\text{BC}}$, (1 off). This line when high will cause the terminal to function as a remote terminal, when low as a bus controller terminal.

4MHz clock, CK4, (1 off), 4MHz System Clock.

4.2 Remote Terminal Initialisation

When power is applied to the chip set a reset cycle will be automatically executed thus causing the internal circuitry of the chips to initialise.

Such a reset cycle will also take place if the open drain $\overline{\text{RESET}}$ input/output line is taken low. This can be achieved by sending a reset mode command to the RT via the data bus, in which case the chip set will pulse the $\overline{\text{RESET}}$ line low, or by the subsystem pulling the RESET line low for a minimum of 0.5 microseconds.

The reset cycle commences with the $\overline{\text{RESET}}$ line being taken low, this forces the $\overline{\text{ADEN}}$ line low which in turn enables the RT address, parity and broadcast enable information from the subsystem onto the data highway. The $\overline{\text{RESET}}$ line being low also forces the internal circuitry of the LSI devices to initialise. At the end of the reset cycle, that is on the low to high transition of $\overline{\text{RESET}}$, the RT address, parity and broadcast enable information is latched into the terminal and internal circuitry is released for normal operation.

The RT address, parity, and broadcast enable information is derived from the subsystem and should be buffered onto the 16-bit data highway, B0-B15, by a tristate buffer.

The terminal makes use of an 8-bit external latch to record subsystem status information which is used by the terminal to control the execution of certain commands and to determine the terminal status word contents.

$\overline{\text{DBCACC}}$ is the dynamic bus acceptance line which is used by the subsystem to indicate whether or not it is willing to accept bus control if offered. If this line is low then the dynamic bus control acceptance bit of the terminal status word will be set in response to a legal mode command for dynamic bus control allocation.

$\overline{\text{SSERR}}$, the subsystem error line is the means by which the subsystem can flag an internal fault condition such as a self test or BITE failure. This line being low will cause the subsystem flag of the terminal status word to be set.

By pulling $\overline{\text{SERVREQ}}$ service request line low the subsystem can cause the service request bit of the terminal status word to be set and thus initiate some predetermined asynchronous operation.

The $\overline{\text{ILLEGAL COMMAND}}$ line provides a means by which the subsystem can declare any command word to be illegal. If this line is low the terminal will inhibit data transfers to or from the subsystem and after message validation will respond with the message error bit of the terminal status word set.

The most obvious use of this facility is in a system which makes use of the instrumentation bit of the terminal status word. Within such a system any command word which has the most significant bit of the subaddress field set low must be illegal. Hence, by connecting the most significant bit of the subaddress field of the command word latch to the $\overline{\text{ILLEGAL COMMAND}}$ line such commands would not be actioned by the terminal.

The $\overline{\text{ALLOW CODE}}$ line provides the subsystem with the capability to declare any of the currently reserved mode codes as being legal and meaningful to the subsystem.

If a reserved mode command is received and this line is not taken low within the allocated time then the terminal will treat the command as being illegal and after message validation will respond with the terminal status word with the message error bit set.

The $\overline{\text{RES0}}$, $\overline{\text{RES1}}$, $\overline{\text{RES2}}$ lines provide the subsystem with the capability of setting any or all of the currently reserved bits of the terminal status word, that is bits 5, 6 and 7 respectively.

4.3 Subsystem Interface Operation

Figure 2 shows the main subsystem interface waveforms that a user would encounter during a typical data transfer. This specific data transfer is a bus controller to remote terminal transfer of two data words into sub address one. The first word shown on line PDIN (the positive threshold logic line between the chip set and the bus driver/receiver) is the command word from the bus controller.

When the terminal receives a valid command word with the correct terminal address any current command execution will be aborted by the $\overline{\text{INCMD}}$ line being forced inactive. The terminal will then enable the command word onto the data highway B0-B15 and write this into the subsystem command word latch by means of the $\overline{\text{RXCMD}}$ and $\overline{\text{STROBE}}$ lines. There then follows a delay of approximately 1 us to allow the subsystem to compile its status information and specify whether or not it is busy, after which this information is latched by $\overline{\text{INCMD}}$ going active low. The subsystem status latch contents are then read into the terminal by means of the $\overline{\text{STATEN}}$ and $\overline{\text{STROBE}}$ lines.

Command execution will then take place as specified by the command word and subsystem status information.

When a valid, non-mode command word is received with a T/R bit of zero the terminal will initialise and the protocol state sequencer will enter the receive sequence.

The exact details of the receive sequence are dependent upon whether or not the subsystem has declared itself to be busy or has declared the command word to be illegal for that terminal via the ILLEGAL COMMAND bit of the subsystem status latch.

In order to declare itself busy the subsystem must pull the BUSYREQ line active low and wait for the BUSYACK line from the terminal to go low. So long as the BUSYACK line is low the busy bit of the terminal status word will be set. The BUSYACK cannot change state while the terminal is actively servicing a command, that is while INCMD is low. The terminal will not make any attempt to transfer data words to or from the subsystem while BUSYACK is low.

BUSYREQ and BUSYACK therefore constitute a true handshake mechanism between the terminal and the subsystem for access to the interface circuitry.

It is assumed the subsystem is not busy and the command has been allowed by the subsystem.

This condition is defined by BUSYACK and the ILLEGAL COMMAND bit of the subsystem status word being both high. In such a situation the normal receive data sequence will be entered.

The terminal will wait until the valid data word is received. When a valid data word is received the data transfer request line, DTRQ, to the subsystem will go active low. The subsystem must reply with a data transfer acknowledge, DTAK, within 1.5 us when the terminal will write the data word into the subsystem store by means of the DTRQ, BUFFEN, R/W and STROBE lines.

If the subsystem fails to reply with DTAK within the allowed time then a handshaking failure will be declared causing the Subsystem Flag of the terminal status word and the Subsystem Handshake Failure bit of the BIT word to be set. The GBR pulse to the subsystem will be suppressed.

At the end of the data word transfer the two data words received will be compared with that specified by the command word and the receive data sequence will continue until these numbers are equal.

When the correct number of data words have been received the terminal will check for an end of transmission, EOT, and if obtained will load the terminal status word into the transmitter, shown by PDOUT, for transmission on the bus. At the same time the good block received signal, GBR, to the subsystem will pulse low for 250ns to declare the received data valid.

If fewer valid data words are received than specified by the command word the Message Error bit of the terminal status word and the Word Count Low bit of the BIT word will be set and status transmission and the GBR pulse suppressed.

If the received message does not terminate after the number of data words specified by the command word then the Message Error bit of the terminal status word and the Word Count High bit of the BIT word will be set. Status transmission and the $\overline{\text{GBR}}$ pulse will also be suppressed.

If the command word has the broadcast terminal address, execution will take place as above with the exception that transmission of the terminal status word will be suppressed.

ACKNOWLEDGEMENT

The author wishes to thank Marconi Avionics management for permission to publish this paper and MOD for their support of the technical development by the Data Management Team of FARL.

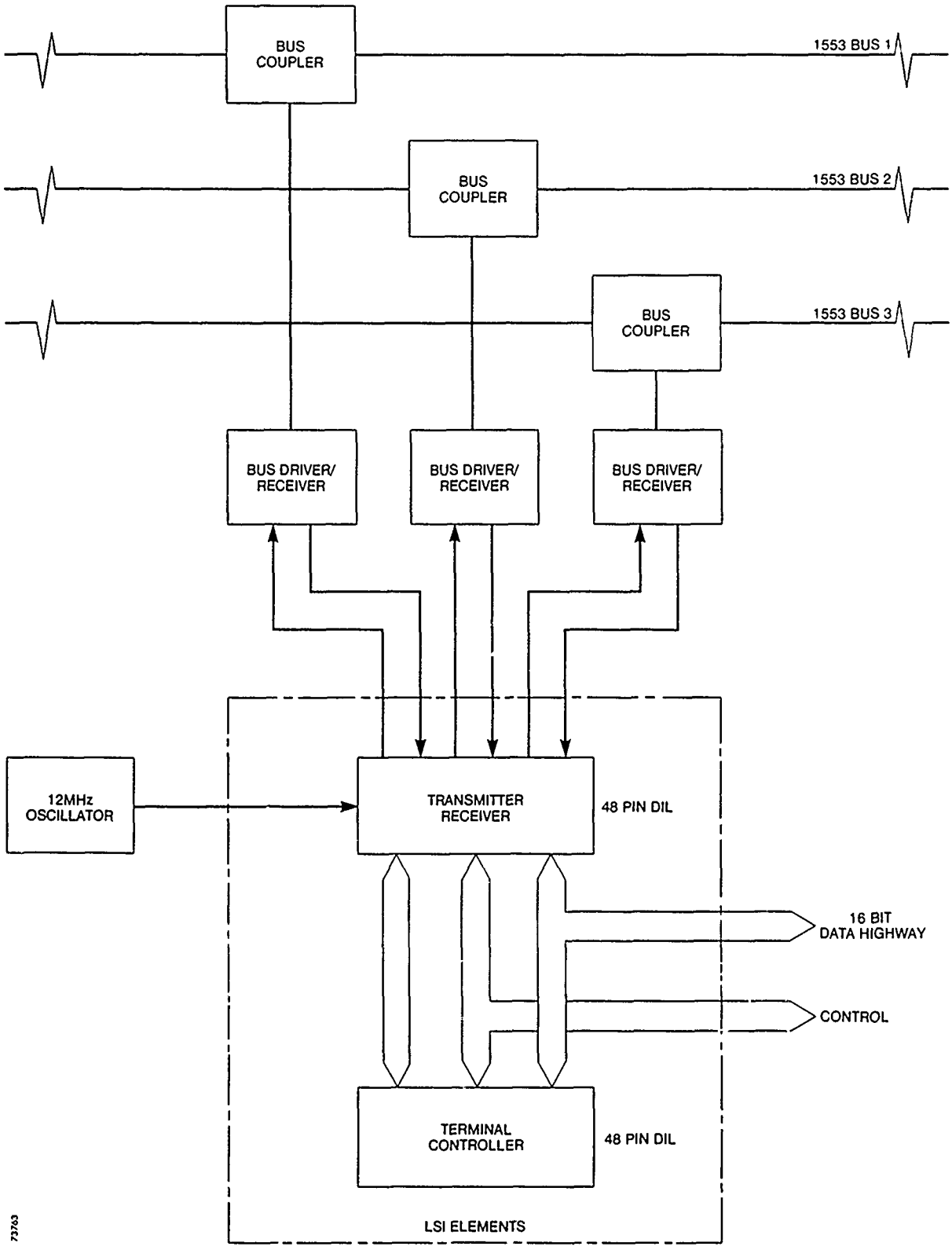

Figure 1 LSI Architecture

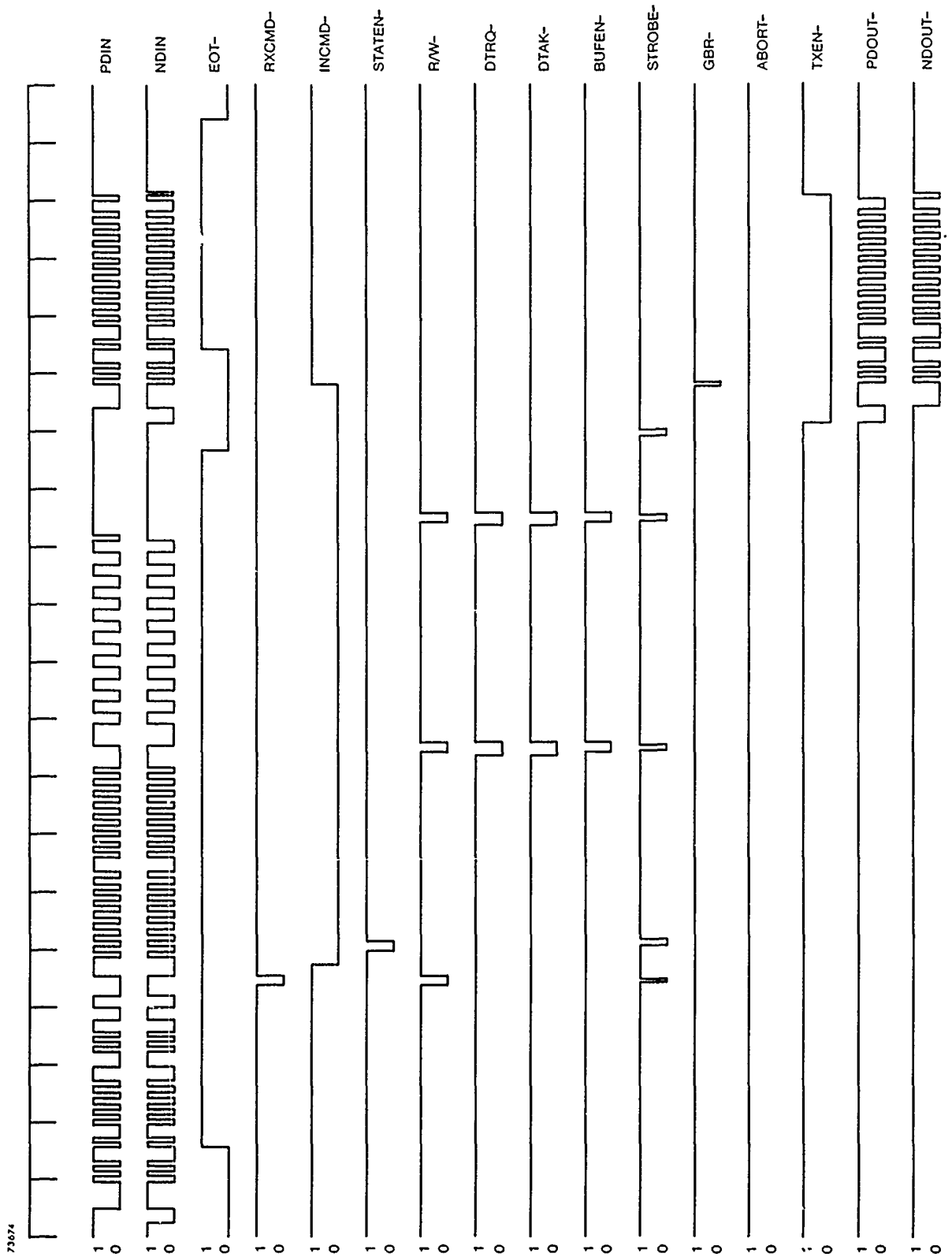

Figure 2 Subsystem Interface Signals