FORWARD TECHNOLOGY, INCORPORATED

FT-68X Single Board Computer Reference Specification

Doc. #78-0006-00 Rev. A

TABLE OF CONTENTS

FT-68X SPECIFICATION SUMMARY

PHYSICAL:

WIDTH:	12.ØØ in. (3Ø.48 cm)
HEIGHT:	6.75 in. (17.15 cm)
DEPTH:	.45 in. (1.14 in)
WEIGHT:	2Ø.Ø oz. (.512 Kg)
SHIPPING WEIGHT: (without manual)	25.Ø oz. (.654 kg)
FORM FACTOR:	IEEE P-796

ELECTRICAL:

5 V + 5 %, 4.1 A(max)

ENVIRONMENTAL:

OPERATING TEMPERATURE: Ø C TO 5Ø C
STORAGE TEMPERATURE: -1Ø C TO 7Ø C
RELATIVE HUMIDITY: 9Ø % WITHOUT CONDENSATION

INTERFACE:

Bus conforms to IEEE P-796
Interrupt Requests: TTL compatible
Serial I/O: RS-423 compatible

SYSTEM CLOCK

FT-68X 9.83Ø4 Mhz +/- Ø.Ø1%

PROCESSOR:

MOTOROLA 68ØØØ or the equivalent
Direct Addressing to 8 Megabytes of Memory
Byte, Word and Longword Data
16 individual 32 bit registers

SERIAL COMMUNICATIONS CHARACTERISTICS

Asynchronous:	5 to 8 bits, 1, 1.5, or 2 stop bits
Synchronous:	5 to 8 bits, internal or external sync. Automatic sync insertion, CRC-16 generation/checking
Bit Synchronous:	SDLC/HDLC; Automatic sequence generation/ checking; Bit insertion deletion; CRC generation/checking.

Baud Rates: 300, 1200, 2400, 4800, 9600, 19200

1.0 Multibus Characteristics - P1 Connector

This section details the Multibus P1 interface. All parenthetical references given in this section refer to specified paragraphs within the proposed IEEE Multibus Standard, IEEE P796.

1.1 System Master Operation - (ref P796, 2.1.1)

This card acts as a Mode 2 system master. This means that it is not limited in the time it is allowed to control the bus. However it can operate as one of a set of system masters using built-in Multibus arbitration logic. See the section on Bus Arbitration for further information.

1.2 System Slave Operation - (ref P796, 2.1.2)

This card is designed to operate as a system slave, in its DMA mode of operation. It accepts Memory Write and Memory Read commands.

1.3 System Clock Lines - (ref P796, 2.1.3.1.1)

The signals Bus Clock (BCLK*) and Constant Clock (CCLK*) are driven to the multibus with the 68000 system clock at 10 Mhz. Both can be optionally disconnected since there may be more than one system master card in the multibus but only one is allowed to drive these signals. For the 10 Mhz 68000 option, both BCLK* and CCLK* are driven at 9.8304 Mhz.

1.4 Read/Write Command Lines - (ref P796, 2.1.3.1.3)

The signals MWTC*, MRDC*, IOWC*, and IORC* are generated onboard for write or read access to memory or to memory mapped I/O devices. Varies from definition of P796 in that a 24 ma drive capability is provided instead of the 32 ma specified in P796.

1.5 XACK* - (ref P796, 2.1.3.1.3)

Conforms to the multibus definition except that the timeout period is 12 microseconds instead of 1 millisecond. A short timeout guarantees that the 68000 can resume normal operation and satisfy real-time requirements without compromising system integrity. In a multi- master environment, it also ensures that the bus is not locked out for excessive time. In other words, MPU requests to the bus are always serviced in a maximum of 12 microseconds.

1.6 INIT* - (ref P796, 2.1.3.1.4)

This signal is driven to the Multibus from the on-board precision voltage comparator, generated when the supply voltage falls below 4.75V. It can be optionally disconnected. INIT can be optionally driven from the Multibus overriding the on-board RESET.

1.7 LOCK* - (ref P796, 2.1.3.1.5)

This signal is not supported.

1.8 Address Lines - (ref P796, 2.1.3.2.1)

This card drives the 20 least significant of the 24 available Multibus address lines. Can optionally be jumpered for connection to the upper 4 lines.

1.9 BHEN* - (ref P796, 2.1.3.2.2)

This signal is generated on this card for 796 bus accesses. It is software controlled by use of byte or word instructions. Internally, the system accesses its memory as a sequence of 16-bit words. The expansion memory sold by Forward Technology for this card does not decode its address or data from the P1 connector and therefore the BHEN* signal is not required for correct access to the expansion memory.

1.10 INH1* and INH2* - (ref P796, 2.1.3.2.2)

These are multibus inhibit lines. These signals are neither generated nor responded to by this card.

1.11 Data Lines - (ref P796, 2.1.3.3)

This card drives and receives 16 data lines. The byte ordering conforms to the specified definition for the 68000 rather than that specified for the Multibus.

1.12 Interrupt Request Lines - (ref P796, 2.1.3.4.1)

Supports a total of 7 levels of interrupt. Interrupt priority conforms to the 68000 definition which labels the 7 levels as INT1 (lowest priority) through INT7 (highest priority, non-maskable) interrupt. INT0 is not supported since the 68000 only has 7 levels of interrupt available.

All interrupts are handled by the 68000 as Auto-vectored interrupts. This means that the card itself is defined as having non bus-vectored interrupts and conforms to the definition given in P796, par. 2.3.2.1. Basically, the system, when it interrupts, need not (shall not) respond with an interrupt vector in that the 68000 already knows where to go to handle that type of interrupt.

1.13 INTA* - (ref P796, 2.1.3.4.2)

This is the interrupt acknowledge line, only used for systems in which the bus-vectored interrupt type is used. This signal is not supported by this card. (See 1.12)

1.14 Bus Arbitration - (ref P796, 2.1.3.5)

Bus arbitration is controlled by an onboard Intel 8289 Arbiter. It handles Bus Request (BREQ*), Serial Bus Priority signals (BPRN* and BPRO*) and the bus busy signal (BUSY*). Common bus request (CBRQ*) operation is NOT supported. Serial Priority as defined in P796, 2.4.2.1, is supported. Parallel Priority, P796, 2.4.2.2, is not supported. In the serial mode, up to three system masters may be coresident in the system, more if the user implements parallel priority logic.

1.15 Notes On Read/Write Memory Timing

The FT-68X is IEEE-P796 compatible as described in the preceeding paragraphs of this section. The P796 bus is an asynchronous bus, accommodating devices with various transfer rates while maintaining maximum throughput. A timeout is provided to abort Multibus cycles if the addressed device does not respond within 12 microseconds. Note
however that the highest performance is achieved by operating the 68000 from onboard memory. Accesses to the Multibus stretch the 68000 cycle by an amount proportional to the device access time.

The FT-68X contains 256K bytes of dynamic RAM with byte parity and up to 32K bytes of EPROM/ROM. The RAM consists of 64K bit dynamic RAM chips.

RAM is refreshed in software by a non-maskable interrupt routine that executes every two milliseconds for a period of fifty-two microseconds. Therefore any Multibus device in a system with this card MUST be able to support a service latency of 52 micro-seconds maximum.

2.Ø Multibus Specifications - P2 Connector

The P2 connector is not intended to be bussed to anything other than the Extended memory card P2 connector. This connection is accomplished using a pair of 6Ø pin connectors which are directly wired pin 1 to pin 1, pin 2 to pin 2 and so forth for the entire set of 6Ø interconnects.

There are no power connections within the P2 connector, only signal (address lines, data lines) and ground. Connections are made as specified in the "P2 Connections" diagram which is pro vided at the back of this document.

3.Ø Onboard Jack and Jumper Specifications

The following chart details the connections for Jacks J1 and J2. Jack J1 is the serial input/output port (TCOM) connector. Jack J2 is the parallel input port connector.

SIGNAL NAME	PIN #	WHICH INTERFACE
Transmit data (TXD)	5	A
Receive data (RXD)	3	A
Signal Ground (GND)	13	A
Request to Send (RTS)	7	A
Clear to Sent (CTS)	9	A
Data Set Ready (DSR)	11	A
Carrier Detect (DCD)	15	A
Transmit data (TXD)	28	B
Receive data (RXD)	3Ø	B
Signal Ground (GND)	38	B

TAKEN FROM FT 3000 INSTALL GUIDE

V. FT-68X J2 JUMPER DIAGRAMS

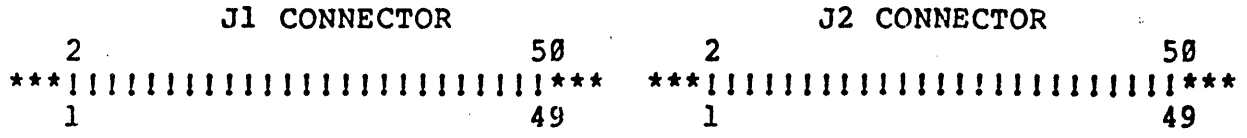


Figure 3 -- Top View of FT-68X Connectors

BOOT DEVICE	J2 PIN NUMBERS 1	3	5	7	9	11	13	15
SYSTEM DISC	O	X	O	O	O	O	O	O
ARCHIVE TAPE	O	O	X	O	O	O	O	O
CIPHER TAPE	O	O	O	X	O	O	O	O
B-PORT ON 68X	O	O	O	O	X	O	O	O
SYSTEM DISC standalone prog	X	X	O	O	O	O	O	O

where X = Jumper In
O = Jumper Out

Additional Jumpers:

Pin Number	Description
17	X = Skip power-up diagnostics
19	X = Verbose boot-up
31	X = Stay in diagnostic monitor do not "boot"

J2 Conector Jumper Settings On the FT-68X
Figure 4

JACK J2:

SIGNAL NAME	MEANING	J2 PIN #
IN0	Input bit 0	01
IN1	Input bit 1	03
IN2	Input bit 2	05
IN3	Input bit 3	07
IN4	Input bit 4	09
IN5	Input bit 5	11
IN6	Input bit 6	13
IN7	Input bit 7	15
IN8	Input bit 8	17
IN9	Input bit 9	19
IN10	Input bit 10	21
IN11	Input bit 11	23
IN12	Input bit 12	25
IN13	Input bit 13	27
IN14	Input bit 14	29
IN15	Input bit 15	31
SET INIT*	Active contact of INIT toggle	45
REF*	Halt indicator output	47

All even numbered pins are grounded.

JACK JP1

Pin 1 to 2 jumper: Connects +5V to J1-19
Pin 3 to 4 jumper: Connects +5V to J1-33

Note that this MUST NOT BE USED if the current draw per pin is greater than 100 ma.

JACK JP2

Pins 2-7, 4-9, 5-6 closed, remainder open configures communications port B for RS-423 or RS-232.

Pins 1-6, 3-8, closed, remainder open configures communications port B for noninverted TTL use.

JACK JP3

Pin 1 to 3 jumper: Use for 2716 PROMs
Pin 2 to 4 jumper: Use for 2732, 2764 PROMs

NOTE: Do NOT connect both jumpers, use only one or the other.

JACK JP4

These are test points only! Use NO jumpers here.

Pin	Name
1	SYS.ACCESS*
2	+5V
3	CLK2Ø
4	GND
5	TIMEOUT*
6	DTACK*
7	BERR
8	GND

JACK JP5

Pin 2 to 4 jumper: Drive interrupt #5 with 72Ø1 USART
Pin 1 to 3 jumper: Drive interrupt #6 with AM9513 timer 2

JACK JP6 - Bus Control

The following chart details the connections at JP6 which specify the option selection for Multibus control.

Pin 1 to 2 Jumper: Resets 68K from Multibus
Pin 3 to 4 Jumper: Drives INIT to Multibus
Pin 5 to 6 Jumper: Drives BCLK to Multibus
Pin 7 to 8 Jumper: Grounds BPRN for first Master in the chain.
Pin 9 to 1Ø Jumper: Drives CCLK to Multibus.

-NOTE: Jumpers 1-2 and 3-4 CANNOT BOTH be made.

JACK JP7

Pin 1 to 3 jumper: Activate DMA with BA19
Pin 2 to 4 jumper: Activate DMA with BA23

NOTE: Use only ONE of these jumpers at a time.

Jack JP8 - Interrupt Control

The following chart details the connections at JP8 which specify the option selection for Multibus Interrupt Response.

SIGNAL NAME	PIN-TO-PIN *	MEANING
B.INT7	1 2	(non-maskable interrupt)
B.INT6	3 4	(used by timer 2)
B.INT5	5 6	(used by UART)
B.INT4	7 ----- 8	(* dotted lines specify
B.INT3	9 ----- 10	factory installed jumpers)
B.INT2	11 ---- 12	
B.INT1	13 ---- 14	
B.INT0	15 ---- 16	(not used)

JACK JP9

Pins 4-5, 1-2 jumpers: Always on a 68X board
Pins 5-6, 2-3 jumpers: If a 68X substrate is being wired as an extended memory

JACK JP10

Pin 3-4, 7-8, 11-12: Implements 20 bit multibus addressing.
Pin 1-2, 5-6, 9-10 : Implements 24 bit multibus addressing.

JACK JP11

Pins 1-2, 5-6: Always

4.Ø Functional Block Diagram

Figure One, bound at the back of this document, shows a block diagram of the FT-68X. Each block in the diagram is numbered. The numbers correspond with the descriptions below.

BLOCK #	NAME	FUNCTION
1	Clock Generator	This circuit provides all of the necessary clocks for the CPU, Timer, and associated circuity.
2	CPU	This is the central processing / arithmetic unit of the FT-68X. Based on the 68ØØØ series Microprocessors, it has 16 bidirectional data lines and 24 address lines.
3	PROM Select Logic	This circuitry examines the address information from the CPU. If it falls within the range of PROM, it will generate the chip Enable Signal for the appropriate PROM.
4	Page MAP	The Page Map forms part of the Memory Management Unit. The Page MAP converts address information from the segment MAP to a physical address which is used in conjuction with address information from the CPU to address RAM or devices on the Multibus.
5	Segment MAP	The segment MAP forms part of the Memory Management Unit. The Segment MAP translates address information from the CPU to a virtual address and sends this address information to Page MAP. The Segment MAP can be used to provide "Protection bits". These bits will determine what type of access will be allowed to a particular Memory location.
6	PROM	This block represents the PROM Memory on the FT-68X.
7	RAM Address Mux	This multiplexor presents 1/2 of the RAM address at RAS time and the other half at CAS time.

8	RAM Clock Generator	The RAM clocks (RAS & CAS) are generated by this circuitry.
9	Address Bus Drivers	These drivers provide address information to the multibus.
10	RAM	This block represents the RAM Memory of the FT-68X. The standard board has 256k of RAM.
11	Data Transceiver	The data transceiver allows data to flow from/to the RAM Memory.
12	Parity RAM	This portion of RAM is used to store parity information generated by the parity generator.
13	Parity Generator	The parity generation circuit examines the data flowing to/from RAM and generates a parity bit.
14	Parity Error Detection	The parity error detection circuit examines the output of the Parity generators. If an error is detected this circuit will notify the CPU.
15	Parallel Input Port	The parallel input port is 16 bits wide.
16	Bi-Directional Bus Drivers	These drivers handle all data communications between the FT-68X and the Multibus.
17	USART	The USART converts parallel Data from the CPU to Serial Data to be sent to an external device; it also converts serial data from an external device to parallel data for use by the CPU. The USART is programmable and is capable of using either asynchronous or synchronous protocols.
18	Transmit Drivers	This circuit converts the TTL logic levels from the USART to EIA levels
19	Receive Drivers	This circuit converts the EIA signal levels on the communications line to TTL levels for use by the USART.

20	Timer	The timer is a programmable device with 5 independent sections. Two of the sections are dedicated to providing the TCOM clocks to the USART. One section is used for Memory Refresh, another Section is utilized as a watchdog timer. The remaining section is for user applications.
21	Interrupt Logic	The interrupt logic examines interrupts from the TCOM, Timer, and Multibus circuitry and passes the interrupts on to the CPU.
22	DMA	The DMA logic routes addresses to a section of multibus memory into the M68000 local memory.

5.0 OPERATION

This section assumes that the user has correctly installed the FT-68X in a Multibus compatible chassis. An ASCII terminal capable of 9600 bits per second operating must be connected to serial port A. It also assumes that the user has installed a RESET switch. The latter item is not essential, but makes the procedure that follows much easier to perform, (rather than powering up again).

5.1 Initializing the FT-68X

Ensure that power is applied to the terminal connected to the FT-68X. Apply power to the Multibus chassis. This also applies power to the FT-68X. On power up, the FT-68X sends a message similar to the following to the terminal:

FTI GATEWAY MONITOR, VERSION 1-40000 BYTES OF MEMORY

If this message does not appear, check the terminal status and the terminal-to-FT-68X connection. If these check out correctly, toggle the RESET switch. The message should appear.

If all the connections and power are correct and repeated use of the RESET switch fails to generate the message, the FT-68X may be defective.

Successful receipt of the message means that the FT-68X has performed a complete initialization procedure, and that the processor board can be assumed to be operating normally. The > symbol means that the PROM-resident monitor program is in COMMAND mode, and ready to communicate with the terminal. The Monitor has performed the following operations after a successful RESET:

- Normalizes all contiguous on-board RAM by writing data with correct parity into each address.

- Write the memory refresh routine and initial trap and vector settings into the first 2 Kb of RAM. Execution of the memory refresh routine is started.

- Sets the USART channels A & B to 9600 baud asynchronous ASCII protocol.

- Initializes and starts the counter/timer functions.

- Sets the Supervisor/Stack (SS) pointer to address X'0017fe'.

- Sets the User Stack (US) pointer to the top of available on-board memory.

- Sets the memory map Segment Table for context 0 and protection level 5 (Supervisor mode has read, write, and execute access to every segment).

- Sets the memory map Page Table. Existing pages of on-board RAM are mapped so that the physical and virtual addresses are identical.

- Trap vectors are set so that Monitor gains control of any exception interrupt.

Physical address space is divided into eight parts as follows:

X'00000' through X'1FFFFF': Mapped address space. The standard board has 256 Kb of on-board RAM, but may optionally have 512 Kb. This space can also be mapped to the Multibus I/O or Multibus memory space.

X'200000' through X'3FFFFF': On-board PROM 0. Actual addresses are: X'200000' to X'201FFF' for 8 Kb
X'200000' to X'203FFF' for 16Kb

X'400000' through X'5FFFFF': On-board PROM 1. Actual addresses are: X'400000' to X'401FFF' for 8 Kb
X'400000' to X'403FFF' for 16 Kb

X'600000' through X'7FFFFF': The on-board dual USART. The Channel A data register is at X'600000': the Channel A command register is a X'600002': the Channel B data register is a X'600004': and the Channel B command register is at X'600006'.

X'800000' through X'9FFFFF': The on-board counter/timer. The data register is a X'800000' and the command register is X'800002'.

X'A00000' through X'BFFFFF': The page Map.

X'C00000' through X'DFFFFF': The Segment Map.

X'E00000' through X'FFFFFF': The Context Register.

5.2 Using The Monitor Program

The Monitor Program performs four basic functions: initializing the processor on power up or RESET, executing the memory refresh routine as described above, communicating with the user via console functions, and providing Emulator Trap service.

The console functions are routines that communicate with the user via the dual USART. All input/output is done using "busy-waits," and the code runs at the highest interrupt priority. Therefore, if a user program is interrupted with some exception, the Monitor will run correctly unless the first 2 Kb of RAM has been damaged. If the user program is undamaged, it can be continued after the interrupt and should be unaffected by the interruption except for the possible loss of some I/O data.

W A R N I N G

There are two rules for dealing with the monitor that ordinarily should not be violated.

1. User programs should not modify the context register.

2. User programs should not write indescriminately into the first two kilobytes of RAM. It is legal for user programs to change exception vectors in Monitor dedicated RAM. However, no Changes are allowed to the level 7 autovectors at address X'00007C' or to the User Interrupt Vectors located between addresses X'000100' and X'0003FF', inclusive. IF DATA IN THESE ADDRESSES IS CHANGED, THE MONITOR PROGRAM WILL FAIL. It will be necessary to RESET the FT-68X, and programs/data resident in RAM will be destroyed.

Console functions are invoked by commands issued in the format:

Verb Space* (argument) carriage return

The verb is always one alphabetic character, and may be either upper or lower case. Space* means that any number of spaces between the verb and argument/carriage return are ignored. Argument is normally either a hexadecimal number of a single upper or lower case letter. As indicated by the (), the argument may be optional. If an argument is present, it must be followed immediately by a carriage return (no space allowed) to start command execution. If a command is entered incorrectly, it can be changed or cancelled at any time before the carriage return key is pressed. Keying BACKSPACE or DELETE (or RUB) erases one character on the current command line. Keying Control U erases the entire command. The correct character/command may then be re-keyed.

NOTE: The symbol <CR> will be used throughout this manual to indicate depressing the carriage return key on the terminal used to communicate with the Monitor program.

In the description of command formats that follow the word open has special meaning. Anytime a memory location, map register, or CPU register is "opened", the name and contents of that address or register is displayed. If you key <CR>, the contents of the address/register remain the same and the Monitor program advances to the next address/register. If you key 0 <CR>, the contents of the address/register are set to zero and the Monitor program advances to the next address/register. If you key new hexadecimal value and <CR>, the new value is entered into the address/register and the Monitor program advances to the next address register. Keying Q <CR> or q <CR> returns the Monitor program to general command level, indicated by the > symbol.

For addresses, "next address/register" means the next byte or word of memory with a starting address larger than the current address. In other words, the command ascends through memory starting at the address given in the first command entered. For registers, "next address/register" means following the sequence: 68000 data registers DO through D7; 68000 address registers A0 through A6; the Supervisor Stack (SS) pointer; the User Stack (US) pointer; the Status Register (SR); and the Program Counter (PC).

5.2.1 Monitor Commands
The Monitor Commands are:

>A n<CR>	Opens 68000 address register n(where n is 0 to 6).
>B<CR>	Set a breakpoint. The Monitor prompts with the old breakpoint address. The user enters a new address, if desired.
>c [address]<CR>	Continue a program starting at the address given. If no address is specified, program execution begins at the current address.
>D n<CR>	Opens 68000 data register n(where n is 0 to 7).
>E [address]<CR>	Opens the work at memory location address. An odd address will be rounded down to the next even address.
>G [address]<CR>	Start executing a program at the address, if given. Otherwise, start executing at the current program counter location.
>H<CR>	Displays the Basic Commands Menu.
>I "mode"<CR>	Set the USART operation to "Mode" where is A, B, or T. "A" means that Channel A to the terminal is operational. "B" means that Channel B to a host computer or terminal isoperational. "T" puts the USART in direct channel (or transparent) mode between Channel A and Channel B. This allows the user at a terminal connected to Channel A to Communicate with the host computer connected to Channel B directly. When communication with the Monitor is again required the user keys CRTL/SHIFT/6 orCRTL/ (up-arrow) followed by lower case C. The Monitor returns to A mode.

>L "Host Command"<CR>	This command is used to differentiate a host computer command from a Monitor command. The "host command" is sent to computer via USART Channel B. The Host nust send a back slash character ( ) to the FT-68X in order to reestablish communications with the monitor after a file has been downloaded. Any data received by the FT-68X prior to receiving the backslash ( ) will be ignored.
>M m<CR>	Opens Segment Map register m.
>O [address]<CR>	Opens that byte at memory location address.
>P p<CR>	Opens Page Map register p.
>R <CR>	Opens the miscellaneous register in order, starting with the SS pointer, then the US pointer, SR, and finally the PC. The SS pointer may not be altered, and any attempt to do so is ignored by the monitor.
>S S-Record<CR>	This condition the Monitor to accept data from the host computer is S-record format. This is the only format that can be accepted from a host computer that is down-line loading programs. The Monitor will return one of three messages in response to the downline load: L for Length Error: K for Checksum Error; or Y for successful load.
>X Character <CR>	Sets the transparent mode escsape character to "character". This allows the user to define a single character replacement for the control sequences given in the I command above.

Examples of the use of Monitor commands follow. The first sets memory address 1234 to ØF00. The second sets the contents of the 68000's data register 3 to 00CF. Underlined portions indicate the entries typed by the user. Non-underlined protions are generated by the Monitor.

```
>e          1234<cr>
 --         --------
001234:     23CF? 0F00 <CR>
                  ---------
001236:     46CF? Q <CR>
                  ------
>d   3 <CR>
 --  ------
D3:  00000231? 00CF <CR>
               ---------
D4:  01303405? Q <CR>
               ------
```

5.3 Loading Programs

The FT-68X is downline loaded using Motorola's "S" record format, so called because each type of record being with a byte containing the ASCII code for an S (Start of Record). An "S" record is a fefine format used in transmitting and receiving programs and data. There are ten possible "S" record types, six of which are in active use, two which are defined but not in active use and two which are reserved. They are defined as follows:

S0	Header Record	Active
S1	16 bit address Data Record	Active
S2	24 bit address Data Record	Active
S3	32 bit address Data Record	Not Active
S4	Reserved	---
S5	Transmitted Data Record Count Record	Active
S6	Reserved	---
S7	32 bit address End of File/Execution address record	Not Active
S8	24 bit address End of File/Execution address record	Active
S9	16 bit address End of File/Execution address record	Active

The "S" record format is shown below:

start record	record type	byte count	address	data (as define by byte count)	Checksum byte
S	2	N N	XXXXXX	YYYY.....YYYY	ZZ
-	-	---	------	----------	--

The "S" and the record type are represented directly in ASCII. The byte count, address, data, and checksum are represented in ASCII hexidecimal (i.e., two frames per data byte, with the most significant digit in the leading frame. The checksum is the 1's complement of the sum of all 8 bit data/address bytes from the

byte count to the last data byte, inclusive and truncated to a byte. In this example each N,X,Y or Z represents a single hexadecimal digit.

The FT-68X Monitor currently supports only record types 2 and 8 (S2 & S8).

Example 1

```
  S         2         Ø8        ØØØØØØ      Ø6532D53        1E
  -         -         --        ------      --------        --
Start     Record    Byte      Address       Data        Checksum
record    type      count                                 byte
```

```
Checksum=Ø8
         ØØ
         ØØ
         ØØ
         Ø6
         53
         2D
         53
         --
         El 1's compliment
             of El = 1E
```

Example 2

```
  S         8         Ø4        ØØ ØØ ØØ          FB
  -         -         --        --------          --
Start     Record    Byte      Execution       Checksum
record    type      count     address         byte
```

The maximum "S" record length is 7Ø frames (a frame is defined as one-byte).

```
Checksum = Ø4
           ØØ
           ØØ
           ØØ
           --
           Ø4 1's compliment of Ø4 = FB
```

When using the S8 record type, an execution address of 000000 will cause no execution to occur; all other addresses are valid.

A typical file to be transmitted would be made up of the following sequence:

(LEADER) nx (S RECORD) (TRAILER)

The leader is a string of a minimum of 32 nulls followed by a carriage return, line feed, and null; sequence. The trailer is a string of a minimum of 32 nulls.

A trailer must appear and goes at the end. Each S record is loaded into memory, starting with the address specified in the S 2 record, provided it passes the checksum test. The trailer serves two functions: to terminate reading the S-records and to load the program counter with the starting address. This is the mechanism for defining the entry point of a program. Program execution can then be started by issuing the G command.

5.4 DMA

The DMA circuit functions to translate the top half of the multibus address space through the 68K mapping hardware into the local memory.

In the 20 bit multibus mode, (see jumpers JP7 and JP10), any multibus memory command from 0x80000-0xFFFFF will be translated through the 68K segment and page maps into 68X local memory.

In the 24 bit multibus mode, (see jumpers JP7 and JP10), any multibus memory command from 0x800000-0xFFFFFF will be translated through the 68X segment and page maps into 68K local memory.

Local memory can be from 256K to 1 Megabyte and is accessed by the 68K over the P2 connector.

Note: The 68K writes to its local memory directly. It does not (and MAY NOT) attempt to access multibus memory space to DMA into its own local memory. Only peripheral controllers to the 68X use this DMA.

The DMA is used as follows:

Software sets up the 68X context register "F" segments 48 (decimal) and above to map the upper half (0x80000-0xFFFFF) of the multibus memory (or some part thereof) to 68X local memory.

Note two items here:

First, DMA always occurs through context "F" segment

registers regardless of which context the 68X is executing.

Second, DMA occurs in segment (decimal) 48 to 64 because that is the manner in which the upper bits (greater than 20) are handled by the hardware. Notice that this gives the software the ability to map the entire local 1 megabyte memory space.

At this point, when any controller attempts a multibus access to the upper half of multibus memory, it will be routed into 68X local memory. Note that you may not have multibus memory (hardware) in this space in that a conflict between the two devices will occur, accessing multiple devices at the same address.

Thus, in a 20 bit multibus mode, you may have 1 megabyte of local memory and 1/2 megabyte of non-DMA multibus memory.

In a 24 bit multibus mode, you may have 1/4 megabyte of local memory and 8 megabytes of non-DMA multibus memory.

6.0 INSTALLATION

The FT-68X board physically conforms to the standards set by the IEEE P-796 form factor. Consequently, the easiest method of installing a FT-68X board in a system is to construct the system using a Multibus standard chassis. Such chassis are commercially available from several sources. In addition, the IEEE P-796 standard provides sufficient information for the user to design a chassis to meet the specific application requirements.

If the user is integrating the FT-68X into an already existing chassis that does not conform to IEEE P-796 standards, the responsibility for insuring rigid mounting with proper clearance from other boards in the system is the user's. As a minimum, however, the board should be mounted on 0.6 inch centers parallel to other large boards. An adequate supply of cooling air, either convection or forced, is essential. The FT-68X consumes approximately 16 watts of power when running. Before performing the installation procedure described below, be sure to remove power from the system.

Inspect the board. If any components are mounted in sockets, ensure that they are firmly seated in their sockets (EPROMs at U101 and U103, and possibly U105).

Insert the FT-68X P1 connector into the 86 pin Multibus connector (Viking 3 KH 43/9Am2 or equivalent). This connect or should be wired for DMA device use -- that is, BPRN is forced low unless there is a multimaster environment.

If the memory expansion board (MEB), is being used, install it in the desired position in the chassis. Connect the FT-68X to the MEB with the cable supplied by FORWARD TECHNOLOGY. Connect the

cable from P2 on the Ft-68X to P2 on the MEB.

A female 50 pin leader connector (Augat part number 110-50001-1-2 or equivalent) is required for access to the Serial I/O ports on connector J1. Port A and Port B are factory configured for EIA-RS423 level signals, using a 9600 baud asynchronous data rate.

To hook up a terminal via an RS-232c line, make connections as illustrated below. View the FT-68X board from the front, with the bus edge connector down. Viewed this way, J1 is the left-most 50-pin connector along the top of the board, and pin 1 is the left-most pin in the near row. Even numbered pins are in the far row.

Don't forget that many terminals require jumpers for:

Request to Send - Clear to Send (4 to 5)

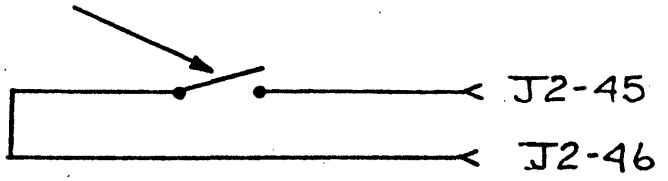
Data Terminal Ready - Carrier Detect (20 to 8)

In order to operate correctly, the Ft-68X receives and transmits at 9600 bits per second.

A switch to reset the FT-68X can be wired onto another 50-pin connector as shown below. Connect pin 45 through the switch to ground.

The pin numbering of J2 is the same as J1 described above, and all of the even pins are grounded on J2.

momentary contact switch



DEFAULT JUMPERING

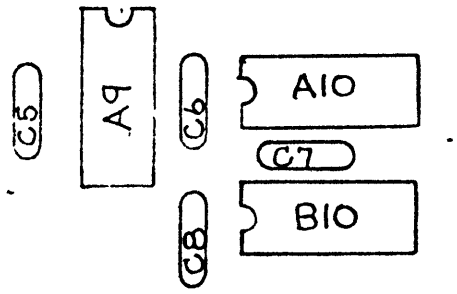
The FT-68X comes with the 8289 bus arbitrator installed and jumpered to be the highest priority bus master. The bus signal BPRN is jumpered to ground on JP6 (7 to 8). Also the FT-68X is strapped at JP6 to drive the bus INIT line. If it is desired to use the bus INIT to reset the FT-68X, then cut JP6 (3 to 4) and put a jumper on JP6 (1 to 2). The FT-68X supplies an 8 MHz or 9.83 MHz to both BCLK and CCLK via JP6 (5 to 6) and (9 to 10).

The dynamic memory refresh method of the FT-68X causes interrupt 7 to occur every two milliseconds and therefore interrupt 7 is disconnected from the bus by cutting JP8 (1 and 2).

When the Ft-68X is correctly hooked up and jumpered it will respond to power applicaiton with this message:

"FTI Gateway Monitor Version 1.2 -- 40000 bytes of memory"

The FT-68X supports RS-423 specifications for voltage driving at +5V. This will support and interface to existing RS-232C hardware. The rise and fall time (slew rate) of the FT-68X meets RS-232C specifications. To modify the slew rate for longer cables (e.g: 100'-4000'); the RS-423 slew rate of 10-90% must be maintained. To adjust the slew rate for Channel A (TxDa, RTSa and DTRA). Capacitors need to be added at the following locations (refer to the figure below).



The capacitor value is dependent on the cable length and modulation rate. Refer to Figure 4.2 for information.

A capacitor of 50pf will increase the rise time (slew rate) by 3us. The present rise time is 100ns. A typical value for RS-423 specification is 3us, which will support either a cable length of 100 ft or modulation rate of 100k bps. If cable length is 50 ft or less, no modification should be done. If channel A cable length is to be increased and capacitors are to be added, the capacitors must be added to all of the lines used. (TxDA, RTSA or DTRA).

Multibus P2 Connections

Sig: Name	Pin #	Sig: Name	Pin #
M.CAS2/	1	M.CAS3/	2
M.RASL/	3	M.REF/	4
M.WE/	5	GND	6
M.DIØ	7	M.DI1	8
M.DOØ	9	M.DO1	1Ø
M.AØ	11	GND	12
M.DI2	13	M.DI3	14
M.DØ2	15	M.DO3	16
M.A1	17	GND	18
M.DI4	19	M.DI5	2Ø
M.DO4	21	M.DO5	22
M.A2	23	GND	24
M.DI6	25	M.DI7	26
M.DØ6	27	M.DO7	28
M.A3	29	GND	3Ø
M.DIL	31	M.DIU	32
M.DOL	33	M.DCU	34
M.A4	35	GND	36
M.DI8	37	M.DI9	38
M.DO8	39	M.DO9	4Ø
M.A5	41	GND	42
M.DI1Ø	43	M.DI11	44
M.DO1Ø	45	M.DO11	46
M.A6	47	GND	48
M.DI12	49	M.DI13	5Ø
M.DO12	51	M.DO13	52
M.A7	53	GND	54
M.DI14	55	M.DI15	56
M.DO14	57	M.DO15	58
M.RASU/	59	Unused	6Ø

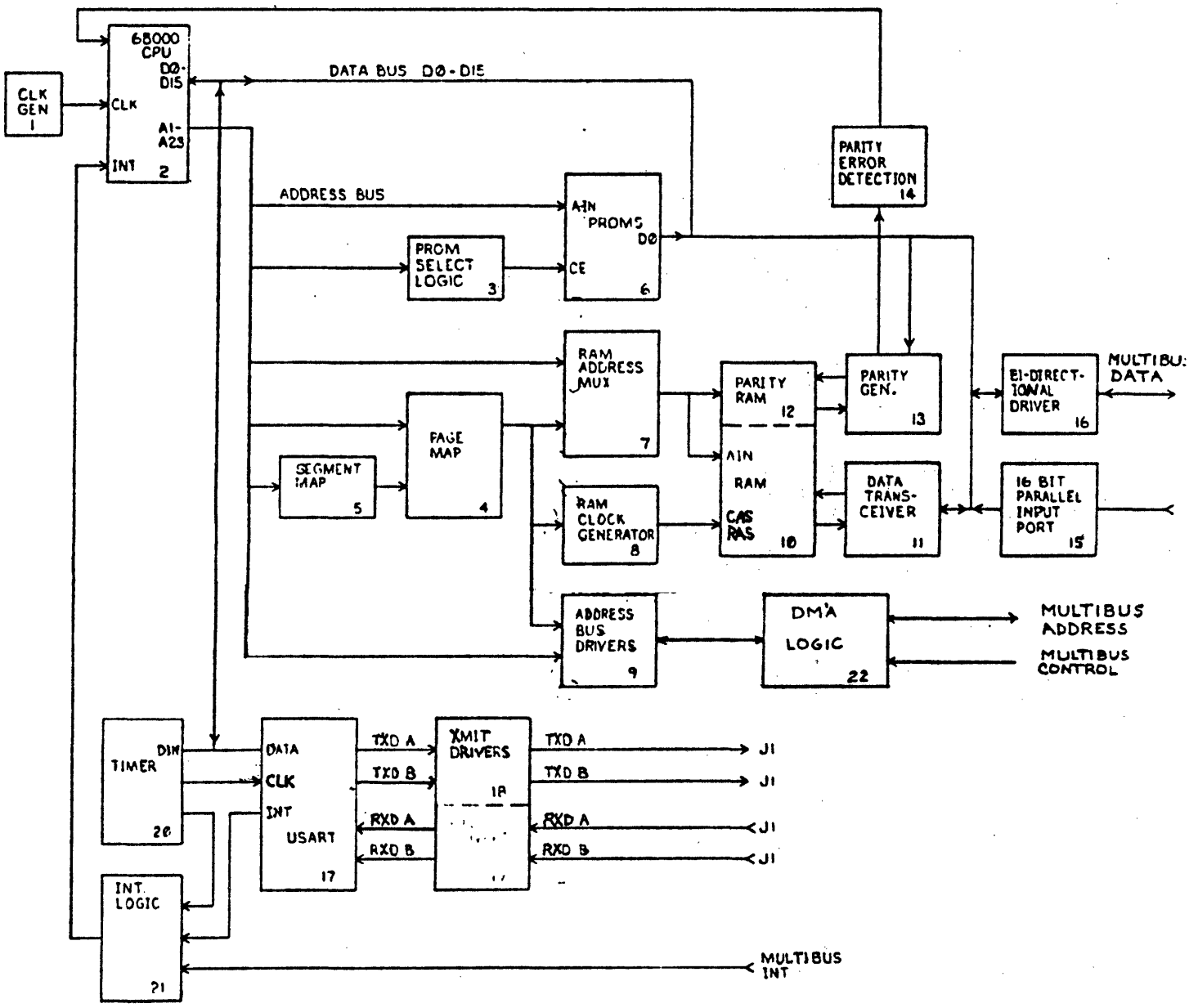
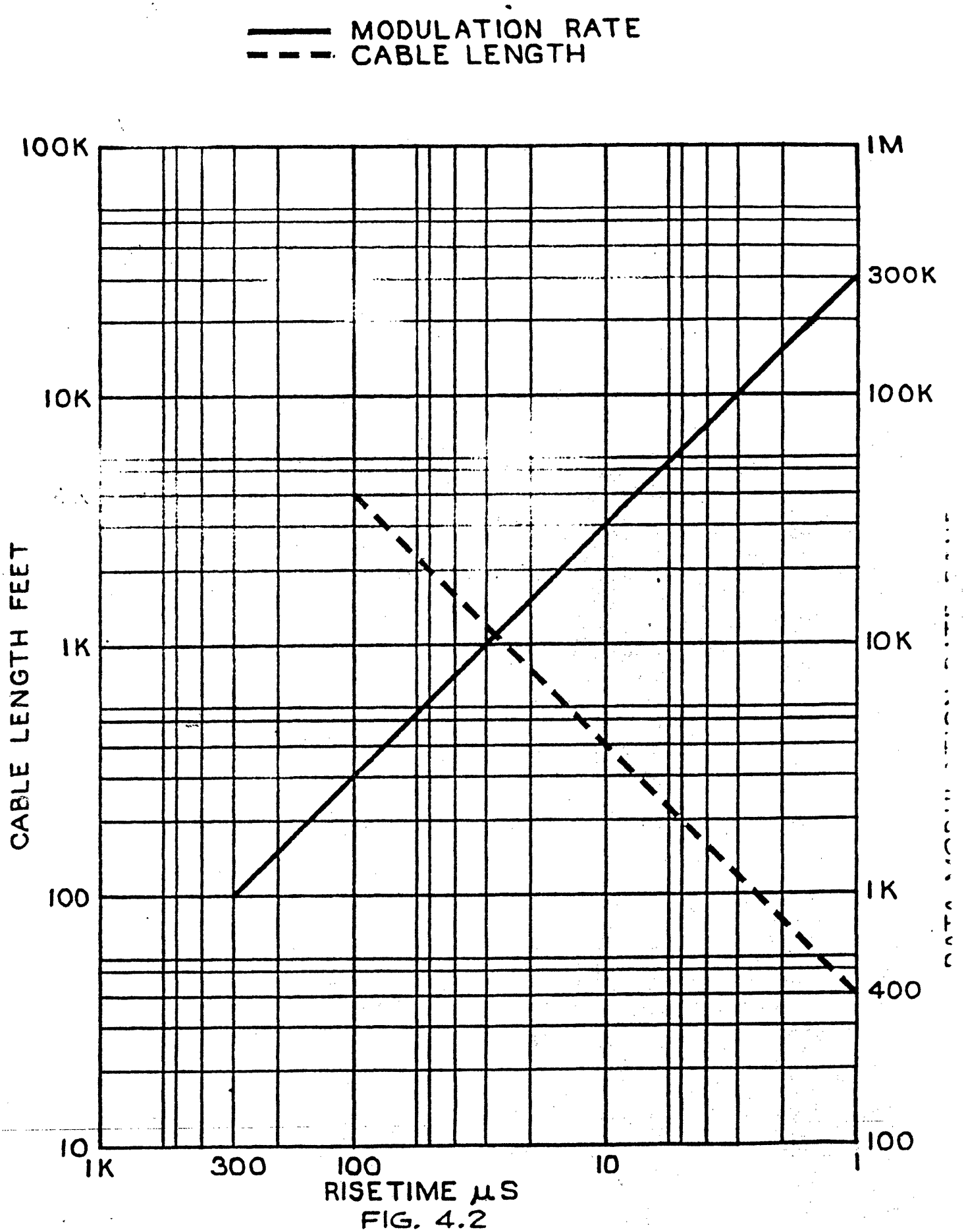


FIG. 1
FTGBX BLOCK DIAGRAM


MODULATION RATE
CABLE LENGTH
100K
10K
1K
100
10
1M
300K
100K
10K
1K
400
100
CABLE LENGTH FEET
1K
300
100
10
1
RISETIME µS
FIG. 4.2