SLAC-PUB-3540
December 1984
(A)

A New Serial System for CAMAC*†

D. Horelick and L. Paffrath

Stanford Linear Accelerator Center
Stanford University, Stanford, California 94305

Summary

This paper describes a new high speed serial CAMAC system developed at SLAC for general use in control systems and experimental applications. The line protocol on which the system is based will be described, followed by the serial branch driver and CAMAC crate controller which support this protocol. The serial branch driver contains an assortment of scan modes particularly useful for data gathering in experiments.

Submitted to Interfaces in Computing

---

* Work supported by the Department of Energy, contract DE-AC03-76SF00515.

† Portions of this paper will appear in IEEE Transactions on Nuclear Science.

1.0 Introduction

A new high-speed Serial Crate Controller (SCC) and Branch Driver (SBD) for CAMAC has been designed at SLAC.

This system is based on

1. Standard RS-422A differential line drivers and receivers
2. Up to 16 crates on a twisted pair party line
3. Self-clocking bi-phase encoded line signals
4. Single master, multiple slave line control
5. Separate twisted pair for prompt L signals
6. Response after every command (system handshake)
7. No parity - for use in limited distance, known environments
8. Versatility and simplicity in the SCC, with more complex scanning algorithms in the SBD.

With a bit rate of 5 Megabits/sec, the overall serial line transaction time is 11 usec for 16-bit read data transfers, independent of the number of crates on the bus. Both 16-bit and 24-bit data transfers are separately supported for the most efficient line utilization.

The SBD contains circuitry to handle some 22 CAMAC scanning modes, and is patterned after a similar parallel branch driver

developed previously at SLAC to achieve a high degree of code compatibility with the prior unit.

2.0 Typical Application

The combination of SCC's and the SBD is particularly useful in applications where either the crates are in remote locations (up to 1200 ft), or a large number of crates need to be controlled. Figure 1 shows a typical system configuration. In this system, one twisted pair is required for the command and data transfers; the second pair, which is an OR of SCC LAMS, is implemented only if the system requires and can handle LAMS.

For systems where the cable length does not exceed 500 feet, cable such as Belden 9730 is satisfactory; for distances up to 1200 feet, cable such as Belden 9851 is used. Both ends of the cable are terminated to eliminate signal reflection.

It should be noted here that the maximum achievable block transfer rate is 276 K bytes per second, assuming a bit rate of 5 Megabits/Sec.

At SLAC, the first operational system using SBD's and SCC's has 3 SBD's driving 3 serial branches. One serial branch 1200 feet in length communicates with 7 crates containing magnet power supply controllers and monitors; the other two branches communicate with 3 crates containing a variety of data acquisition modules.

3.0 Line Protocol

The discipline of the serial link connecting the SBD and SCC is shown in Fig. 2. The messages are grouped into "commands," from SBD to SCC, and "responses," from the addressed SCC back to the SBD. For most efficient use of the line, each message is preceded by a unique SYNC bit at the electrical line level. This SYNC consists of a double width pulse not occurring in any message, which initializes all units on the line for every message and assures that they properly interpret the 3 line control bits A,B,C, on every transaction thereby avoiding troublesome sync search and recovery procedures. Once a crate is addressed by a command message, it remains in this state until unaddressed by another command. Hence an inherent feature of this line protocol is "single address" block transfers for read, write, and control, achieving an increased transfer rate due to lower line overhead. Other types of block transfers are achieved by SBD algorithms; these achieve higher data rates through CPU overhead reduction.

Another approach to improved line utilization is the use of dual data formats, one for 16 bits, the other for 24 bits (note use of the "C" bit). This capability exists for both random operations and the block transfer mode. Actual transaction times for 16 bit transfers are given in Table 1, not including CPU overhead.

For every action in an addressed SCC a response message is returned to SBD. With every such message the state of Q,X, and L is included, thereby providing a type of "polled" L capability,

in addition to the prompt L described previously. Once a crate containing a positive L response is identified, a special command to the SCC returns the state of all 23 L's in the affected crate to further identify the source(s).

The presence of a defined response after every operation gives continuous confirmation of system operation as well as providing a system handshake for optimum timing of operations.

4.0 Special Commands of the SCC

A set of special commands directed to the SCC itself is patterned after similar operations in the standard type A and L-2 crate controllers. These are shown in Table 2 and are self-explanatory.

5.0 Description of the Serial Branch Driver (SBD)

The design of the SBD was greatly determined by the requirement to be highly software compatible with a parallel branch driver system already existing at SLAC. A restriction in the use of the SBD is that it must be used in a CAMAC crate where the crate controller recognizes and obeys the HOLD line.

The basic operation requires two CAMAC cycles in the SBD crate for every CAMAC cycle in the remote crate. First, a control word (Fig. 3) is loaded with an F(17)A(x). This control word can be read back by an F(1)A(x) read command. Then, either an F(16)A(x) or an F(0)A(x) will cause a serial transmission to, a CAMAC cycle in, and a response from the remote crate:

Upon receipt of this command, the HOLD line is set to one. Then the control command is transmitted via the mux (shown in Fig. 4) thru the data encoder onto the data bus. The following three actions occur:

1. If the F code in the control word is a write, the data on the W lines is transmitted via the mux and data encoder; the SBD then turns off the transmit enable and awaits the response.

2. If the F code in the control word is a read, the SBD turns off the transmit enable, and awaits the read data and response.

3. If the F code in the control word is a dataless operation, the SBD turns off the transmit enable and awaits the response.

Upon receipt of the response, the SBD releases the hold line, allowing the normal completion of the dataway cycle. Should there be no response within the expected time, the SBD times out, releases the hold line and again allows the dataway cycle to complete. Figure 5 shows a typical SBD timing sequence. The dataway S1 & S2 strobes occur after the release of the hold line.

The X and Q response of the remote crate is passed to the dataway. It should be noted that although the SCC includes in its response the state of L within its crate, this is ignored by the

current implementation of the SBD for software compatibility with the SLAC parallel branch driver.

The mode bits are used for scanning operations, which are used to improve system throughput for large data acquisition systems. Of the 32 possible modes, 22 are implemented. Table 3 shows several examples. Mode 1 allows scanning of a single module. Mode 2 a scan of all modules within a crate. Mode 4 a scan of all crates, Mode 15 a scan of all modules in all crates. When using these modes, the SBD takes full advantage of the short command for Read Block Transfer or Repeat Control.

6.0 Description of the Serial Crate Controller (SCC)

System design of the SCC to implement the described protocol is quite straightforward. A block diagram is shown in Fig. 6. The data decoder regenerates NRZ data, clock, and sync from the encoded data. The data is then shipped to serial to parallel shift registers as directed by the control circuitry. The internal 10 MHz crystal clock takes over, and runs a CAMAC cycle followed by parallel-to-serial conversion in shift registers as appropriate. A multiplexer selects the "read" register, or the L register as appropriate, for transmission to the data encoder, which adds sync bits A,B,C, encodes into bi-phase, and terminates the message properly.

All special commands are decoded in an 82S100 FPLA. The key elements in the control sections are two 82S105 FPLS which direct

actions and store sequence states depending on the bit count state contained in a common 64-bit counter. This same "count state" philosophy is used to generate the CAMAC cycle during the "transmit sequence" instead of using discrete logic.

7.0 Line Signals

Differential line signals are used. A typical encoded data stream is shown in Fig. 7. Note that a transition occurs at each normal 200 nsec bit boundary to permit bit by bit clock recovery in the decoder, which avoids phase locked loops. A transition in the center denotes a "one"; no transition indicates a "zero." For a self-clocked system, this type of data encoding makes the most efficient use of line bandwidth. The unique sync pulse preceding each message is a 400 nsec positive pulse, easily distinguishable from the rest of the message. Termination following the message consists of returning the line to the "zero" level one-half bit time later, if the final level at the end of the message is at the "one" level. Both sync and termination are shown in Fig. 7.

The twisted pair is terminated in its characteristic impedance at each physical end to eliminate reflections. The termination network shown in Fig. 8 also includes a small amount of inherent bias, to keep the receivers in the "zero" state when all drivers are in the high impedance state.

As the signals propagate along the line, they are distorted by line dispersion and a "jitter" is introduced at the zero, or slicing level, where an ideal receiver switches. This jitter, or timing error, is what primarily limits the line length.

We have run systems at 5 megabits per second using Belden 9730 cable over 500 feet with immeasurable error rates, and over 1000 feet using Belden 9851 cable. We have found that the maximum transmission distance is achieved when the S.B.D. driver is in the high impedance state rather than the "zero" state, prior to the message transmission.

8.0 Encoder/Decoder

The encoder (transmitter) and decoder (receiver) circuits used in both SBD and SCC are shown in Fig. 9. In the encoder the J-K flip-flop assures a transition at each bit boundary, as well as at the center for a "one." Sync and termination generation circuitry are not shown.

The decoder is one-shot based for simplicity and toleration of line distortion. The circuit derives clock pulses, sync, and NRZ data from the received signal stream.

Acknowledgements

The authors wish to acknowledge the support of R. Melen and M. Breidenbach at SLAC, and the contributions of M. Browne

and S. MacKenzie. H. Walz first suggested the use of the FPLS to implement the control logic of the SCC.

TABLE 1

Transaction Times at 5 Megabits/Sec.

Random Read	11	usec
Random Write	12	usec
Control	8	usec
Single Address Read BT	7.5	usec
Single Address Write BT	7.5	usec
Control BT	4.5	usec

Note 1: Read and Write times assume 16 bit data transfers

Note 2: BT denotes Block Transfer

TABLE 2

Special Commands of the SCC

N(31)	Addresses all modules
N(30)F(0)A(0-7)	Read L signals, I line, L Enable
N(30)F(24)A(9)	Set I = 0
N(30)F(26)A(9)	Set I = I
N(30)F(24)A(10)	Disable L
N(30)F(26)A(10)	Enable L
N(28)F(26)A(9)	Run CAMAC Cycle with C = 1
N(28)F(26)A(8)	Run CAMAC Cycle with Z = 1, Set I = 0, Disable L
Power on	Same as Z

TABLE 3

Selected SCAN Modes of the SBD

Scan Mode	LX	LQ	SC	SN	SA	Description
1	0	0	0	0	1	Scan A up to 15 with the same N & C. L is generated after A = 15.
2	0	0	0	1	0	Scan N up to 23 with the same A & C. L is generated after N = 23.
4	0	0	1	0	0	Scan C up to 15 with the same A & N. L is generated after C = 15.
7	0	0	1	1	1	Scan A up to 15 with the same N & C. Then scan next N with A = 0 to 15 until N = 23. Then scan next C with N = 1 to 23, and A = 0 to 15 for each N. L is generated after A = 15, N = 23 and C = 15.

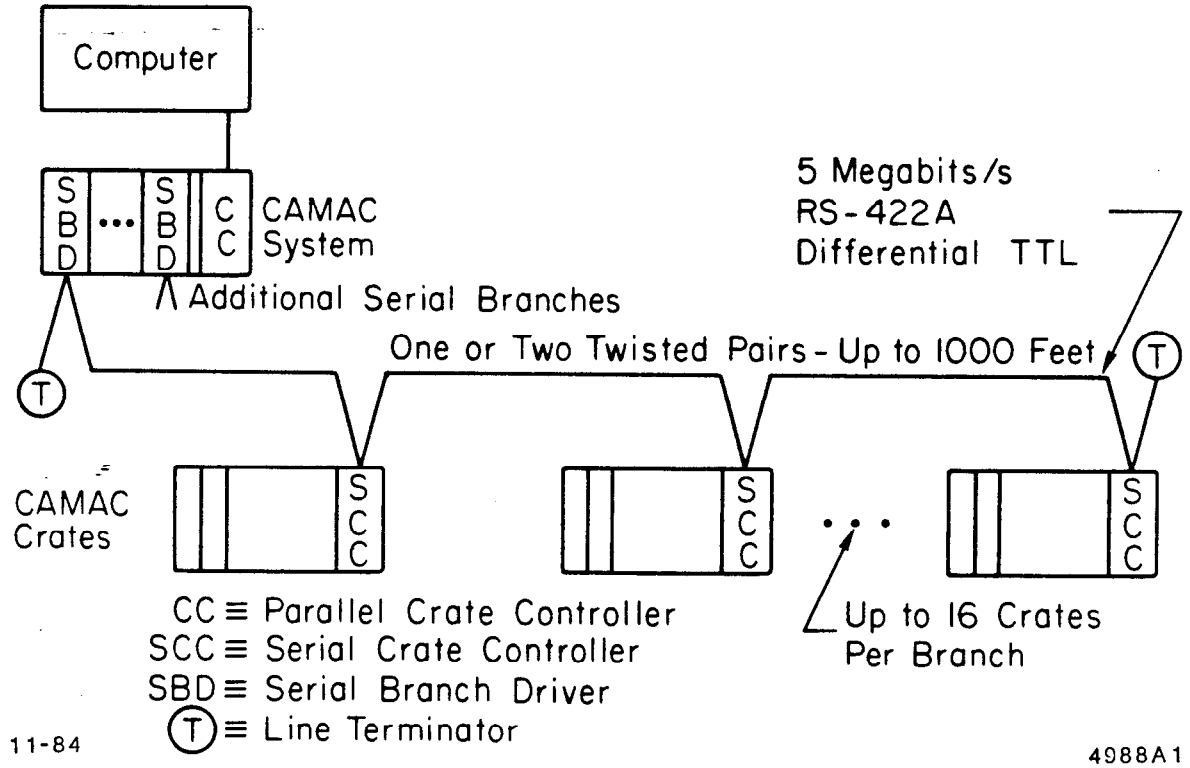


Fig. 1 Typical System Configuration

SERIAL LINE PROTOCOL

The protocol is defined by a special sync bit, followed by 3 line control bits, the command or data message, and completed by a terminator.

SYNC	A	B	C	Message	T

A ≡ Direction
B ≡ Type
C ≡ Word Length

Commands (From Driver to SCC):

CAMAC Command:

A	B	C	Crate				Function					Module					Subaddress			
0	0	X	1	2	4	8	1	2	4	8	16	1	2	4	8	16	1	2	4	8

Write Data - 16 bits or 24 bits:

A	B	C	Write Data														Opt. 24 Bits								
0	1	0	W1	16 Bits												W16	8 Bits								W24

Short Command for Read Block Transfer, or Repeat Control:

A	B	C
0	1	1

Responses (From SCC to Driver):

Read Data - 16 bits or 24 bits:

A	B	C	Q	X	L	Read Data														Opt. 24 bits								
1	0	X				R1	16 bits												R16	8 bits								R24

L = "or" of L lines in addressed crate, gated by L enable.

Read L Lines - 24 bit mode:

A	B	C	1	*	L	Read L Line																						L24
1	0	1				L1	24 bits																					

* Reads state of L Enable F-F

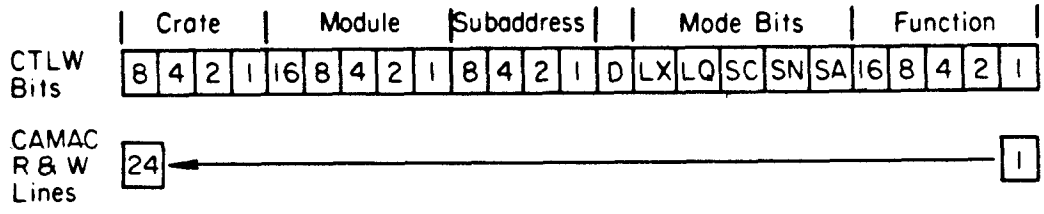
Short Response for Write, Write Block Transfer, Control:

A	B	C	Q	X	L
1	1	1			

10-84

4936A5

Fig. 2 Line Protocol



Where:

D - the data mode, D = 0 corresponds to the 16-bit mode
D = 1 corresponds to the 24-bit mode

LX, LQ, SC, SN, SA
- scan mode control bits where in general:
SA implies scan A (LSB)
SN implies scan N
SC implies scan C
LQ implies LAM on no Q
LX implies LAM on no X (MSB)

12-84
4988A3

Fig. 3 SBD Control Word Format

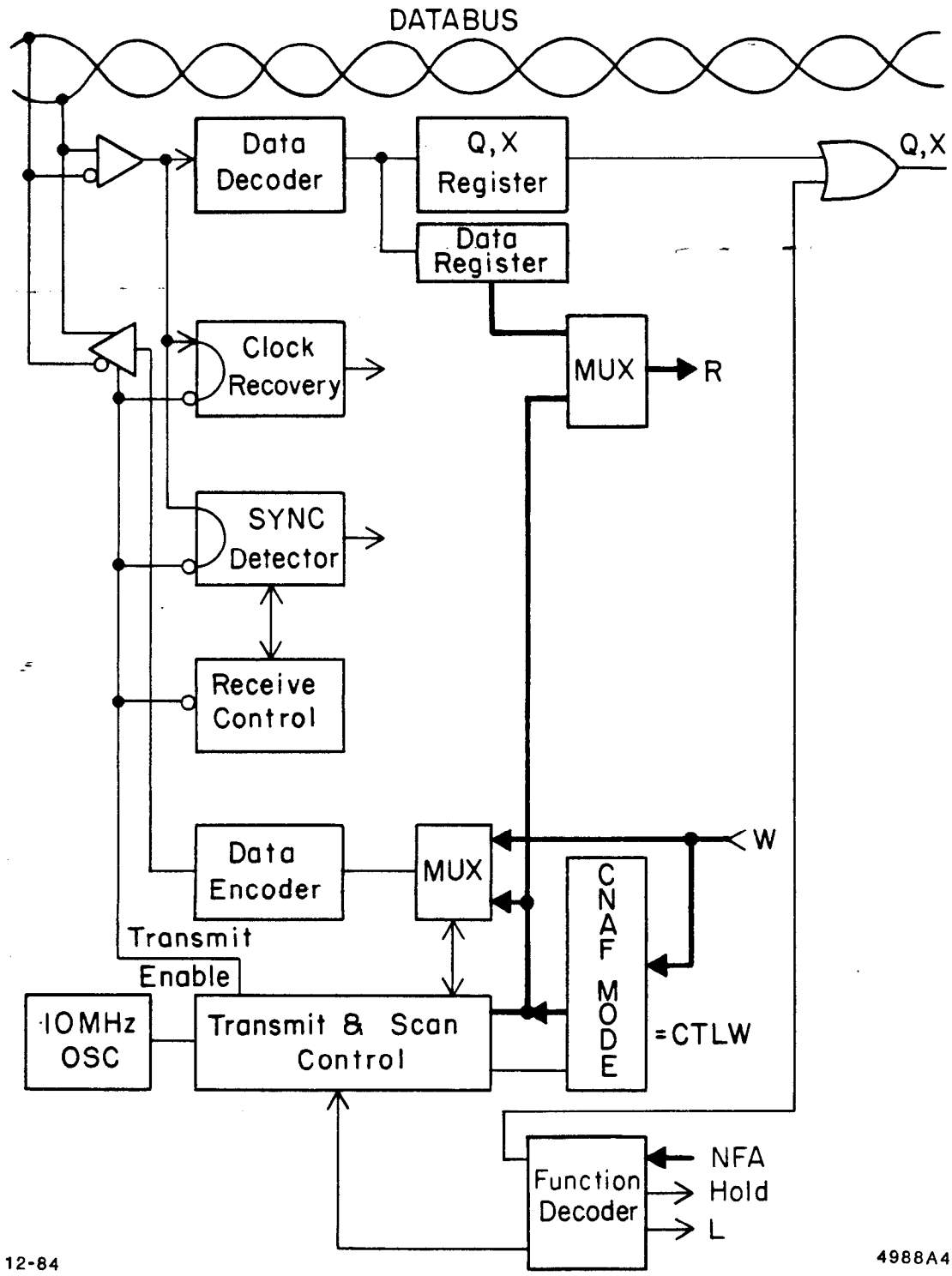


Fig. 4 SBD Block Diagram

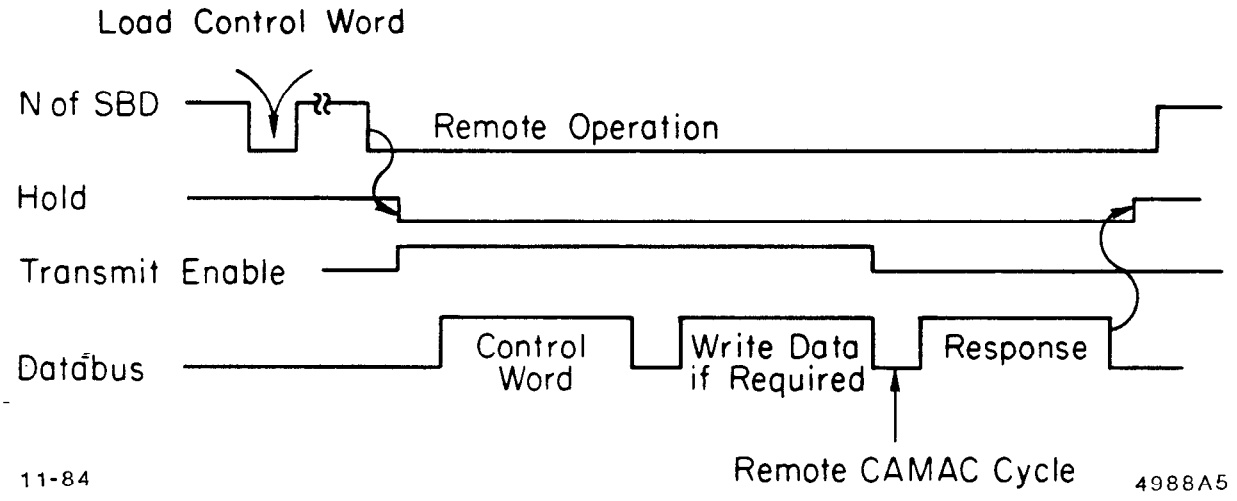


Fig. 5 Typical SBD Timing Sequence

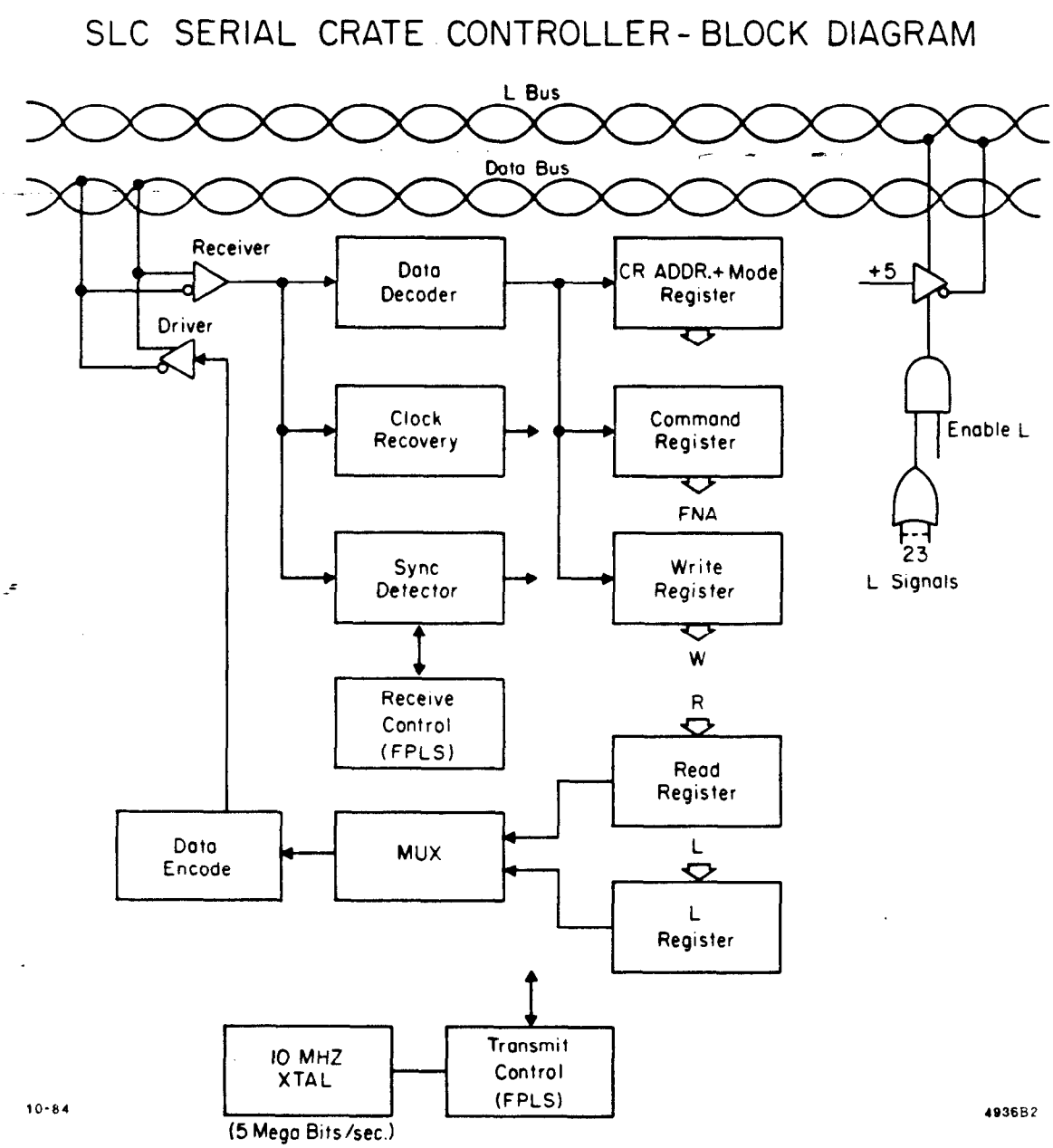


Fig. 6 Block Diagram of SCC

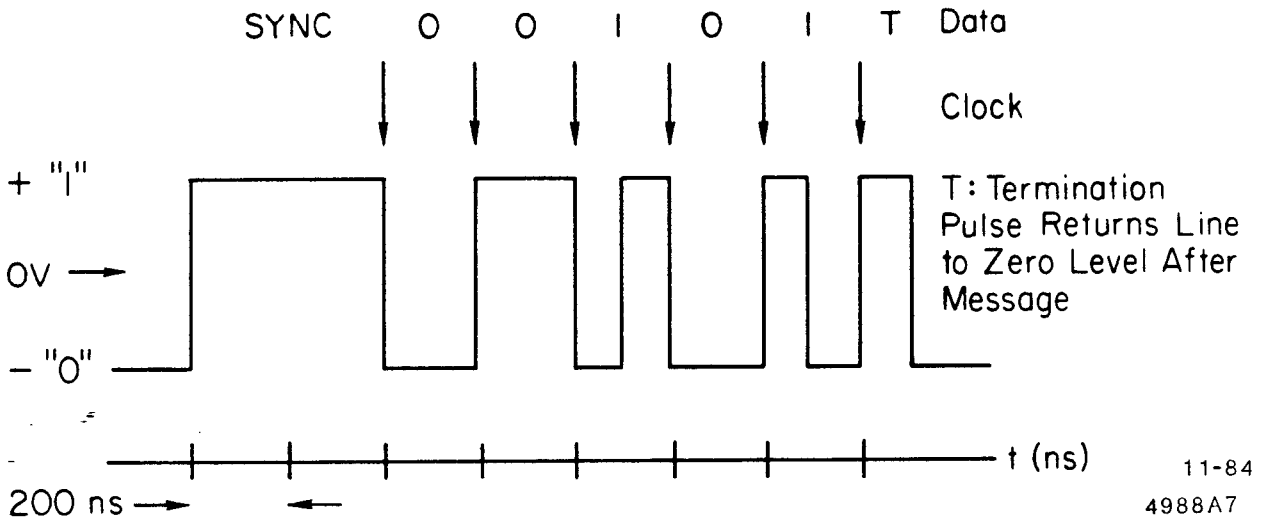


Fig. 7 Line Signal Waveform

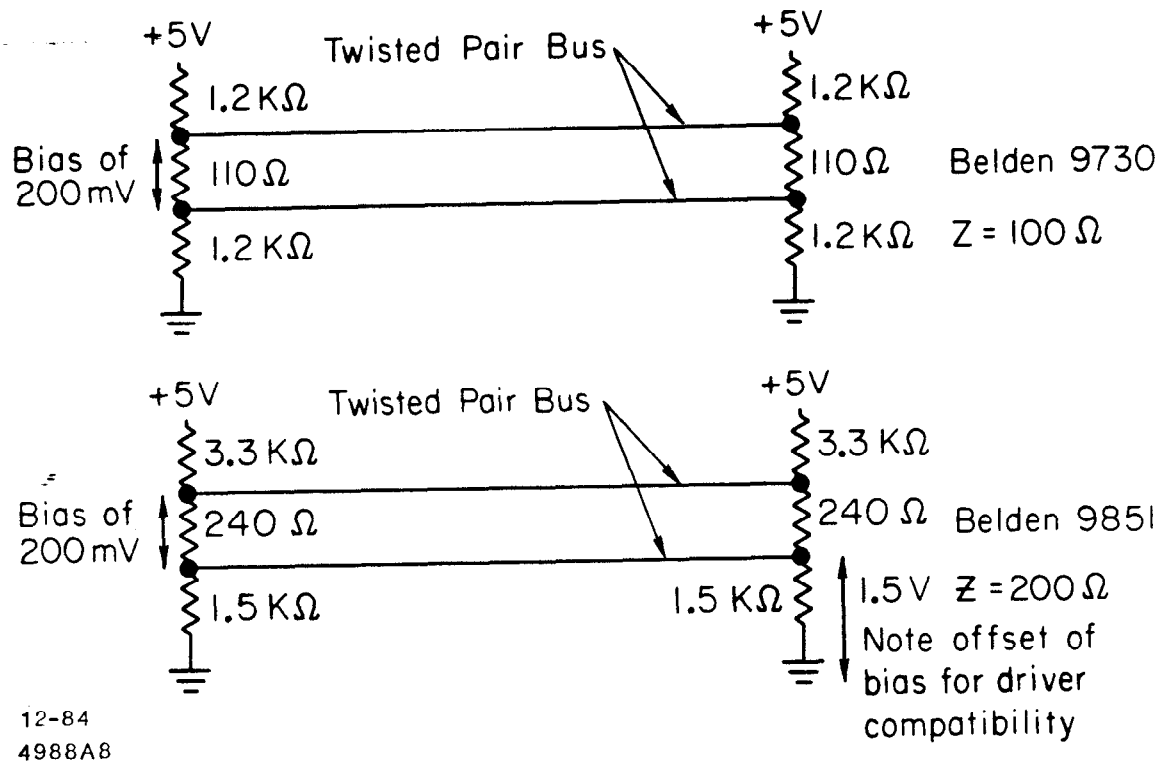


Fig. 8 Line Termination Networks

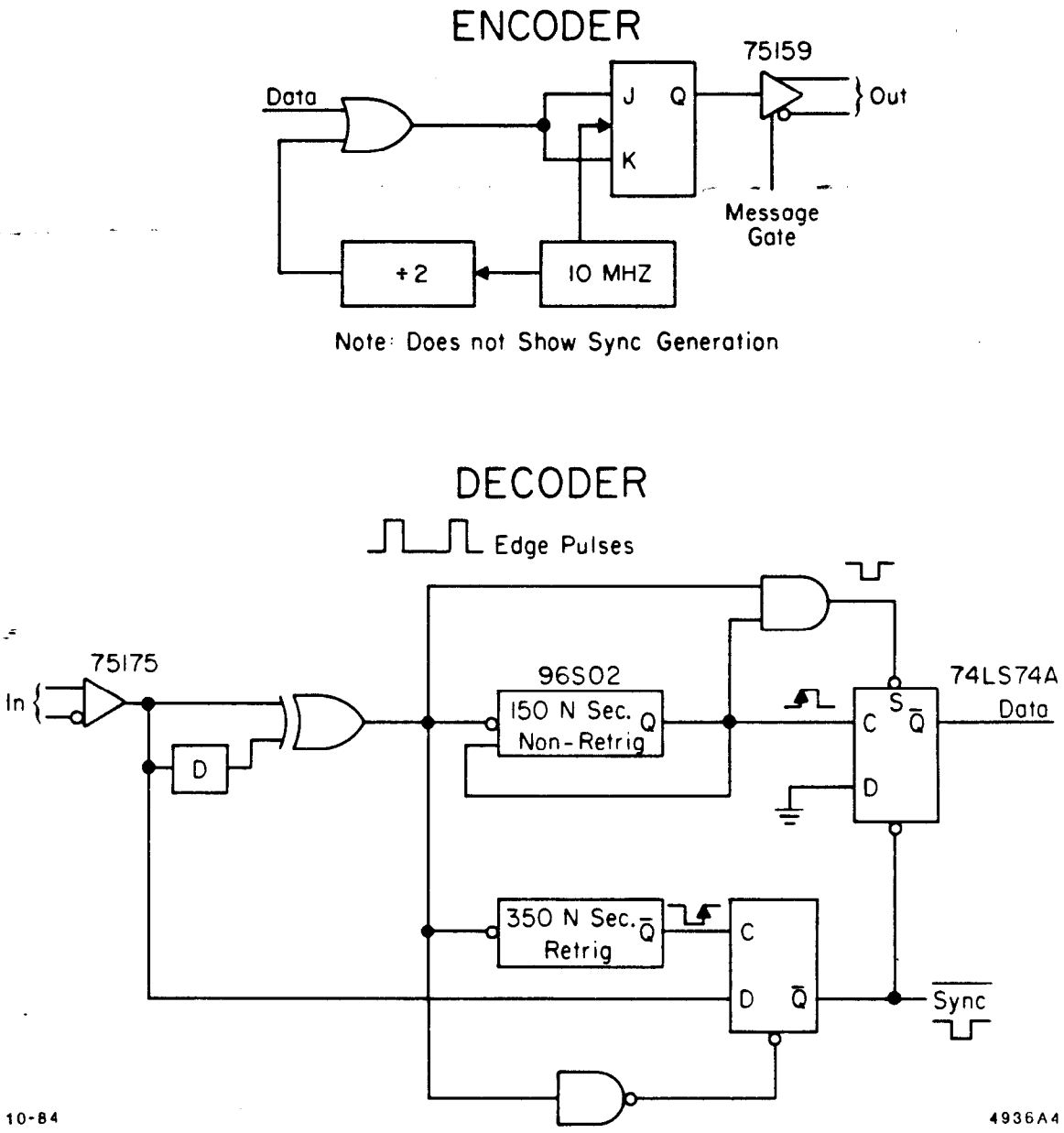


Fig. 9 Encoder/Decoder Circuits