KG11-A exclusive-OR and CRC block check manual

pdp11

digital

DEC-11-HKGAA-B-D

KG11-A exclusive-OR and CRC block check manual

digital equipment corporation · maynard. massachusetts

1st Edition, May 1972
2nd Printing (Rev), October 1972
3rd Printing, February 1973
4th Printing, July 1973
5th Printing, October 1974

The material in this manual is for informational purposes and is subject to change without notice.

Printed in U.S.A.

CONTENTS

Page

CHAPTER 1 INTRODUCTION

CHAPTER 2 GENERAL DESCRIPTION

2.1 Introduction 2-1
2.2 Functional Description 2-1
2.2.1 System Description 2-1
2.2.2 KG11-A Functional Units 2-2
2.3 Overall Functional Operation 2-4
2.3.1 Normal Cycle Operation 2-4
2.3.2 Single-Step Operation 2-4
2.4 Physical Description 2-5

CHAPTER 3 DETAILED DESCRIPTION

3.1 Introduction 3-1
3.2 Address Selector and Gating Control 3-1
3.2.1 Address Assignments 3-1
3.2.2 Address Selector and Gating Control 3-2
3.2.3 Gating Control Logic 3-4
3.3 Control and Status Register 3-6
3.4 Data Register 3-6
3.5 Block Check Character (BCC) Register 3-9
3.6 Control Signals Generated by the CSR 3-10
3.6.1 Introduction 3-10
3.6.2 Mode Selection Bits D02:00 and DDB Bit D03 3-10
3.6.3 CLEAR Bit D04 3-11
3.6.4 STEP Bit D05 and Shift Enable (SEN) Bit D06 3-12
3.6.5 Shift Clock and Shift Control Logic 3-14
3.66 Exclusive-OR Operators and Feedback Logic 3-17

CHAPTER 4 PROGRAMMING INFORMATION

4.1 Introduction 4-1
4.2 LRC and CRC Codes 4-1
4.2.1 Introduction 4-1
4.2.2 CRC-16 4-1
4.2.3 CRC-12 4-2
4.2.4 CRC-CCITT 4-2
4.2.5 LRC-8 4-2
4.2.6 LRC-16 4-2
4.3 Recommended Programming Practices 4-3
4.3.1 Introduction 4-3
4.3.2 BCC Computation on a Message Basis 4-3
4.3.3 Partial BCC Computation (Character-by-Character) 4-5

CONTENTS (Cont)

Page

4.3.4 Single-Step Operation 4-5

CHAPTER 5 MAINTENANCE

5.1 Introduction 5-1
5.2 Testing and Diagnostics 5-1
5.3 Test Points 5-1

APPENDIX A BCC COMPUTATION FOR LRC AND CRC

A.1 Introduction A-1
A.2 Longitudinal Redundancy Checking A-1
A.3 Cyclic Redundancy Checking A-1
A.3.1 Introduction A-1
A.3.2 Mathematical Background A-2
A.3.3 Hardware Implementation of CRC A-4
A.4 Hardware Configuration of BCC Register A-5

APPENDIX B ABBREVIATIONS

APPENDIX C INTEGRATED CIRCUIT DESCRIPTIONS

7474 Dual D-Type Edge-Triggered Flip-Flops C-3
8242 4-Bit Digital Comparator (Quad Exclusive-NOR) C-5
8266 2-Input 4-Bit Digital Multiplexer C-7
8271 4-Bit Shift Register C-9
74197 Presettable Binary Counter/Latch C-11

ILLUSTRATIONS

Figure No. Title Page

2-1 Typical System, Block Diagram 2-2
2-2 KG11-A X-OR and CRC Block Check Option, Block Diagram 2-2
3-1 KG11-A Address Word for Data Register 3-1
3-2 Typical Jumper Connections in Address Decoder, Logic Diagram 3-3
3-3 Address Selection Logic and Gating Control Logic, Block Diagram 3-5
3-4 Control Signals Generated During DATI (Read) Operation, Logic Diagram 3-5
3-5 Control Signals Generated During DATO (Write) Operation, Logic Diagram 3-6
3-6 Control and Status Read/Write Functions, Logic Diagram 3-7
3-7 Control and Status Register Bit Assignments 3-7
3-8 Data Register Format 3-9
3-9 BCC Register Format 3-9

ILLUSTRATIONS (Cont)

Figure No.	Title	Page
3-10	Mode Selection Control Signals, Logic Diagram	3-11
3-11	Clear and Initialize Control Signals, Logic Diagram	3-12
3-12	STEP and SEN Control Signals, Logic Diagram	3-13
3-13	Shift Clock and Shift Control, Logic Diagram	3-15
3-14	Counter States and Overflow for Eight Shift Pulses	3-17
3-15	Counter States and Overflow for 16 Shift Pulses	3-18
3-16	Control of 8266 Multiplexer in Feedback Circuit, Logic Diagram	3-19
4-1	BCC Computation on a Message Basis, Flow Chart	4-3
4-2	Step Mode Operation, Flow Chart	4-6
A-1	BCC Computation for Four 8-Bit Characters Using LRC-8	A-2
A-2	BCC Computation Using Long Division Method With CRC-16 as Generator	A-4
A-3	BCC Accumulation Using CRC-12, Transmit Sequence	A-6
A-4	BCC Accumulation Using CRC-CCITT, Transmit Sequence	A-7
A-5	BCC Accumulation Using CRC-16, Transmit Sequence	A-8
A-6	BCC Accumulation Using CRC-16, Receive Sequence	A-9
A-7	BCC Register, Block Diagram	A-10

TABLES

Table No.	Title	Page
3-1	KG11-A Gating Signals	3-2
3-2	KG11-A Select Signals	3-4
3-3	Control and Status Register Bit Functions	3-8
5-1	KG11-A Test Points	5-2

CHAPTER 1
INTRODUCTION

The KG11-A Exclusive-OR and CRC Block Check Option provides error detection capabilities for data communication applications. It is used primarily to compute the block check character (BCC) that appears at the end of a block of data transmitted over a serial synchronous line. The KG11-A operates, non-concurrently, with any number of devices.

Although signals are transferred between the KG11-A and the Unibus, this manual does not describe the operation of the Unibus. A detailed description of the Unibus is presented in the *PDP-11 Unibus Interface Manual,* DEC-11-HIAB-D.

The manual provides the user with the theory of operation necessary to understand and maintain the KG11-A Exclusive-OR and CRC Block Check Option. The level of discussion assumes that the reader is familiar with basic digital computer theory.

This manual is organized into five chapters: Introduction, General Description, Detailed Description, Programming Information, and Maintenance. An appendix is included that describes various BCC accumulations in graphic form. A separate set of engineering logic drawings is provided with each KG11-A and is identified as D-CS-M7251-0-1, sheets 1 through 4.

CHAPTER 2
GENERAL DESCRIPTION

2.1 INTRODUCTION

This chapter presents an overview of the operation of the KG11-A Exclusive-OR and CRC Block Check Option. It contains a functional description that is keyed to the block diagram level. In addition, the discussion includes some background information on longitudinal and cyclic redundancy checking.

2.2 FUNCTIONAL DESCRIPTION

2.2.1 System Description

Under program control, the KG11-A implements any one of five error detection codes. Two codes utilize longitudinal redundancy checking (LRC); they are LRC-8 and LRC-16. Three codes utilize cyclic redundancy checking (CRC); they are CRC-12, CRC-16, and CRC-CCITT. The generator polynomials for these codes are:

$$\begin{aligned}&\text{LRC-8} = X^8 + 1\\&\text{LRC-16} = X^{16} + 1\\&\text{CRC-16} = X^{16} + X^{15} + X^2 + 1\\&\text{CRC-12} = X^{12} + X^{11} + X^3 + X^2 + X + 1\\&\text{CRC-CCITT} = X^{16} + X^{12} + X^5 + 1\end{aligned}$$

These five error detecting codes represent the most popular techniques for providing error detection in domestic and foreign data communications systems. The KG11-A computes a block check character (BCC) as instructed by the code selected. The BCC is added to a block of data transmitted over a serial synchronous line. Both the transmitting and receiving stations must use the same code for computing the BCC.

A typical system is shown in Figure 2-1. All units except the modem are connected to the Unibus. The DP11 is a synchronous line interface that allows the PDP-11 processor to control the modem. For received data, the characters are moved to the KG11-A and a BCC is computed for the data and compared to the BCC received. If they are equal, the data is assumed to be correct and is accepted. If they are not equal, the message is not accepted and the data is retransmitted. For data to be transmitted, the characters are moved to the KG11-A and the BCC is computed. The BCC is moved to the message buffer for transmission at the end of the message.

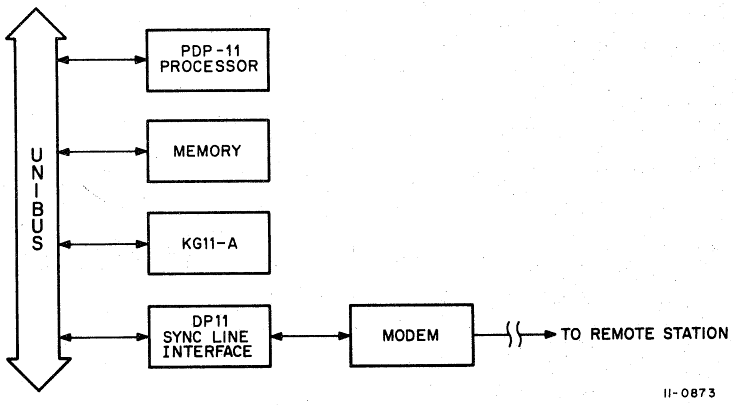


Figure 2-1 Typical System, Block Diagram

2.2.2 KG11-A Functional Units

The major functional units of the KG11-A include the following (Figure 2-2):

a. address selector
b. gating control logic
c. Unibus receivers and drivers
d. control and status register (CSR)
e. data register
f. BCC register
g. shift clock and shift control logic
h. exclusive-OR and feedback logic

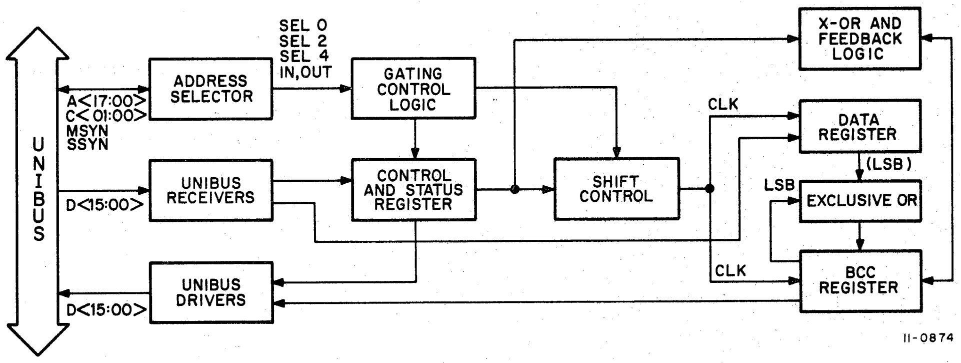


Figure 2-2 KG11-A X-OR and CRC Block Check Option, Block Diagram

a. Address Selector – The information on the Unibus address lines A ⟨17:00⟩ and control lines C ⟨01:00⟩ is decoded by the address selector. It provides select and gating signals that determine which unit (CSR, data register, or BCC register) has been selected and whether it is to perform an input or output transaction. The logic is equivalent to the standard Address Selector Module M105.

(continued on next page)

b. Gating Control Logic – The gating control logic uses the address selector outputs to provide input/output signal access for the CSR, data register, and BCC register. It generates enabling signals to read and write the CSR, load the data register, and read the BCC register.

c. Unibus Receivers and Drivers – Buffer all signals to and from the Unibus and the KG11-A.

d. Control and Status Register – The 9-bit CSR controls the operation of the KG11-A. It selects the mode of operation and the polynomial (code) to be used. The control functions require five bits that are read/write and two that are write-only. Two bits are read-only.

A list of the bits is shown below.

Bit	Type	Signal	Function
02:00	read/write	16 BCC (00) LRC (01) CRC I/C (02)	Mode selection bits. Determines the polynomial (code) to generate the BCC.
03	read/write	double data byte (DDB)	Determines whether one or two characters (16 bits maximum) are loaded into the data register.
04	write-only	clear (CLR)	Clears the BCC register.
05	write-only	STEP	Enables the single-step mode of operation which is used for diagnostic purposes. Used in conjunction with bit 06 (SEN).
06	read/write	shift-enable (SEN)	Determines the KG11-A mode of operation: cycle or normal mode; and step or diagnostic mode.
07	read-only	DONE	Controls the mode of operation (parallel load or shift) of the data register. Also controls the shift clock.
08	read-only	quotient (QUO)	Result of the exclusive-OR of the LSB of the data register and the LSB of the BCC register. Used for diagnostic purposes during the step mode.
15:09	not used		

e. Data Register – The data register is a 16-bit write-only register. It is the input register for the data on which the BCC is computed. Load is accomplished in parallel from the Unibus data lines. The data register is operated in a serial, shift right mode to enter data into the BCC register during computation of the BCC.

f. BCC Register – The BCC register is a 16-bit read-only register that maintains the BCC accumulation.

g. Shift Clock and Shift Control Logic – The shift clock provides the pulses to shift the data register and the BCC register during a BCC accumulation. The shift control logic provides the required number of shift pulses to these registers in accordance with the polynomial selected.

h. Exclusive-OR and Feedback Logic – This logic sets up the correct number of exclusive-OR operators and feedback paths for the BCC register in accordance with the polynomial selected.

2.3 OVERALL FUNCTIONAL OPERATION

2.3.1 Normal Cycle Operation

The KG11-A can be used to compute a BCC in two ways: on a message basis or on a character-by-character basis. It is recommended that the KG11-A be used to compute the BCC on a message basis. This discussion assumes a message BCC computation; details of a character-by-character (partial) BCC computation are covered in Chapter 4, Programming Information.

The first step in operating the KG11-A is the initializing process. The program moves a command word to the CSR by placing the CSR address (7707X0) on address lines A ⟨17:00⟩ and selecting control lines C01 = 0 and C00 = 1, which provide a DATO (write) bus data transfer. The signals are sent to the address selection logic which decodes them and asserts SELECT 0 H and OUT LOW H. These signals are sent to the gating control logic to enable a write operation for the CSR which strobes the information on data lines D ⟨06:00⟩ into the CSR. The data lines were selected by the program simultaneously with the A and C lines and are the control bits for the CSR. They perform the following functions:

a. Bits 16BCC (D00), LRC (D01), and CRC I/C (D02) select the polynomial to be used in computing the BCC. They provide the correct exclusive-OR operator and feedback configuration for the BCC register in accordance with the polynomial selected. LRC and 16BCC also preset the shift counter to 0 or 2, as required, to generate the correct number of pulses (8 or 6 pulses if DDB = 0; 16 or 12 if DDB = 1).

b. Bit DDB (D03) determines whether one or two characters are to be loaded into the data register. When DDB = 1, the number of shift pulses is doubled.

c. Bits STEP (D05) and SEN (D06) select the mode of operation: cycle or single step. In this discussion, assume that they have been selected to provide the cycle or normal mode of operation.

d. Bit CLR (D04) clears the BCC register to ensure that a new BCC accumulation starts with all 0s in the BCC register. The CLR bit also presets the DONE flip-flop, which places the data register in the parallel-load mode. The data register is also cleared by the initialize signal from the PDP-11 processor during the power-up cycle.

After the control word has been written into the CSR, the program loads the data register as specified by the selected polynomial. This is accomplished by placing the address of the data register (7707X4) on address lines A ⟨17:00⟩ and selecting control lines C01 = 0 and C00 = 1, which provide a DATO (write) bus data transfer. Data lines D ⟨16:00⟩ are also selected by the program. The control line signals and address line signals are sent to the address selection logic, which decodes them and asserts SELECT 4 H and OUT LOW H. These signals enable a load operation for the data register which strobes the information on data lines D ⟨16:00⟩ to the data register. After the data register is loaded, the select signals also clear the DONE flip-flop, which puts the data register in the shift mode and qualifies the control gate of the shift clock. When the select signals are dropped by the program, the clock starts. The cycle begins and after the correct number of shifts the clock is stopped and the BCC is in the BCC register.

2.3.2 Single-Step Operation

For diagnostic purposes, the KG11-A can be single stepped and its operation monitored after each step.

The operation starts with the initializing process. It is the same as that described for the normal cycle mode except that the SEN bit (D06) is cleared. The data register is loaded as in the normal cycle mode. The program sends another command word; this time only the STEP bit (D05) is asserted. This action produces a single clock pulse for the data register and the BCC register. Each time the STEP bit is asserted by the program, a single step is performed. Actually, one exclusive-OR operation has been performed. After each step, the program can read the contents of the BCC register and read the quotient (QUO) bit in the CSR to check the operation of the KG11-A.

2.4 PHYSICAL DESCRIPTION

The KG11-A consists of a single quad-height module (M7251) that can be installed in either a DD11-A peripheral mounting unit or in one of the two PDP-11 processor small peripheral controller slots. The component placement on the module is shown in drawing D-CS-M7251-0-1, sheet 1. It presents one unit load to the Unibus and has a maximum power requirement of 1.2A at +5V. The operating temperature range is 10°C to 50°C, with up to 90 percent humidity (non-condensing).

CHAPTER 3
DETAILED DESCRIPTION

3.1 INTRODUCTION

This chapter provides a detailed description of the KG11-A Exclusive-OR and CRC Block Check Option. Each major functional unit is discussed separately and in regard to its interrelation with other functional units. Basic descriptions of data-in (DATI) and data-out (DATO) Unibus transactions are provided to assist in understanding overall system operation.

The text refers to engineering logic drawings D-CS-M7251-0-1, sheets 1 through 4, which are supplied separately. Simplified logic diagrams and block diagrams are also used to support specific areas of discussion.

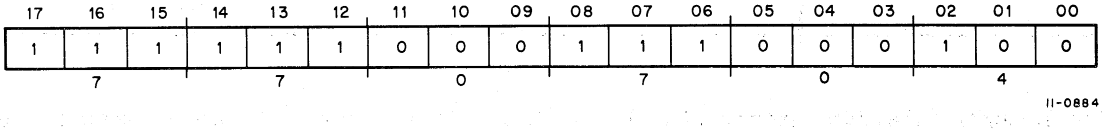
3.2 ADDRESS SELECTOR AND GATING CONTROL

3.2.1 Address Assignments

Each KG11-A contains three registers that require 16-bit addresses (bits 0–15) which must be assigned consecutively as follows.

Address	**Register**
7707X0	status
7707X2	BCC
7707X4	data

Addresses are assigned for eight KG11-As per system. In the address designation, X = 0 for the first KG11-A, X = 1 for the second, . . ., and X = 7 for the eighth. Figure 3-1 shows a typical KG11-A address word in both octal and binary notation.



17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
1	1	1	1	1	1	0	0	0	1	1	1	0	0	0	1	0	0
	7			7			0			7			0			4	

Figure 3-1 KG11-A Address Word for Data Register

The Unibus address word contains 18 bits (A ⟨17:00⟩) which provide the capability of addressing 256K memory locations, each of which is an 8-bit byte. The basic PDP-11/15 and 11/20 processors provide program control for only 16 bits of address information (64K bytes or 32K words). The word length and bus width in these processors is 2 bytes (16 bits). A word operation accesses two locations at once: the Unibus contains the address of the even-numbered location only and the next higher odd location is selected, as well, to provide a 16-bit word. A byte operation (DATOB) accesses any location (odd or even numbered) to select a byte (8 bits).

Assuming the use of a PDP-11/15 or PDP-11/20 processor with a maximum of 64K memory locations, bits A ⟨17:16⟩ are forced to 1s if bits A ⟨15:13⟩ are all 1s when the processor is master. With bits A ⟨17:13⟩ all 1s, the last 8K-byte locations are relocated to become the highest locations accessible by the bus. All device addresses and internal processor locations are assigned in these 8K locations. The assigned addresses for the KG11-A registers are in these locations.

3.2.2 Address Selector and Gating Control

The address selection logic is shown in drawing M7251, sheet 4. It is functionally equivalent to the standard PDP-11 M105 Address Selector Module; its input/output pin designations are identical to the M105.

The M7251 address selection logic uses only three of the four select signals (SEL0, SEL2, and SEL4); SEL6 is not used. It uses only two of the three gating signals (IN and OUT LOW); OUT HIGH is not used. The KG11-A operates as a slave device only. It uses only two bus transactions: data-in (DATI) which transfers (reads) a 16-bit data word from the KG11-A (slave) to some master device; and data-out (DATO) which transfers (writes) a 16-bit data word from some master device to the KG11-A (slave).

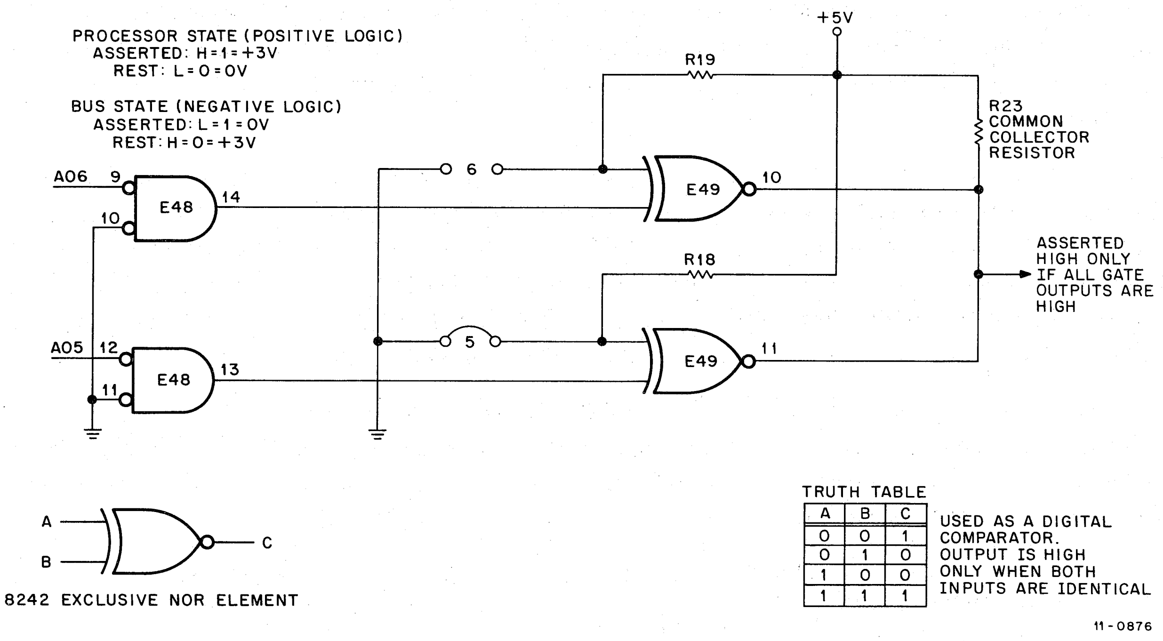
Control lines C01 and C00 and address line A00 determine the type of bus-data transfer to be implemented. In the case of the KG11-A, only DATI and DATO are used, so the state of A00 is irrelevant. Bit A00 is the byte control bit and it has an effect only when C01 and C00 both equal 1. The state of A00 when C01 = C00 = 1 determines which byte (low or high) is accessed during a data-out byte (DATOB) transaction. The DATOB transaction is not used in operating the KG11-A. Control bits C01 and C00 and address bit A00 are inputs to the address selection logic (drawing M7251, sheet 4). They generate three gating signals: IN H, OUT LOW H, and OUT HIGH H. Table 3-1 shows the gating signals generated by the two bus transactions (DATO and DATI) used by the KG11-A.

Table 3-1
KG11-A Gating Signals

Mode Control C ⟨01:00⟩	Byte Control A00	Bus Data Transfer	Gating Signal True (+3V)
00	X	DATI	IN H
01	X	DATO	OUT LOW H
			OUT HIGH H (not used)

The first five digits of the address indicate which KG11-A in the system has been addressed. Using the example in Figure 3-1, the first five digits, 77070, indicate that the first KG11-A in the system has been addressed. Octal 77070 represents bits A ⟨17:03⟩ which are inputs to the address selection logic (drawing M7251, sheet 4). To conform to the established address bounds for device registers, bits A ⟨17:13⟩ must all be 1s. Decoding of bits A ⟨12:03⟩ is determined by jumpers. If a line contains a jumper, the decoding logic searches for a 0 on that line; if there is no jumper, it searches for a 1. In this example, jumpers would be installed in bit positions 11, 10, 9, 5, 4 and 3. When the processor asserts the address that matches the jumper configuration, the decoder associated with bits A ⟨12:03⟩ produces a high. This signal is used in the bus control and SELECT signal portions of the address selection logic. The decoder is a unique circuit and is discussed below.

Unibus address lines A ⟨17:03⟩ are connected to Unibus receivers which are type 380A 2-input NOR gates (bottom row of gates on drawing M7251, sheet 4). These gates are shown as negative-input ANDs which are logically equivalent to positive NORs. Thirteen receivers are used for 15 address signals: A16 and A15 go to one gate and A14 and A13 go to one gate; all others, A03–A11 and A17, go to individual gates. The output of the receiver associated with A17 (E38 pin 14) is sent to E42 pin 10. Each output of the other 12 receivers is sent to an 8242 exclusive-NOR. There are four gates per device and they are designated E40, E45, and E49. These gates have a common output connection which is the other input of E38 (pin 9). Jumpers are provided for bits A ⟨12:03⟩. A typical example using bits A06 and A05 is shown in Figure 3-2. If the first five octal digits of the address are 77070, bit A06 = 1 and A05 = 0 (Figure 3-1). In this example, bit A06 is low and bit A05 is high. The other input of each receiver is held low (logical 0); therefore, the A06 receiver output is high (1) and the A05 receiver output is low (0). There is no jumper associated with A06 so +5V passes through R19 to hold one input of the A06 exclusive-NOR high; both inputs are high so its output is high. There is a jumper associated with A05 so the +5V is dropped to ground via R18 and the jumper to hold one input of the A05 exclusive-NOR low; both inputs are low so its output is high (truth table in Figure 3-2). This process is repeated with address bits A03, A04 and A07–A012. Bits A ⟨16:13⟩ are all high (logic 0 at the bus) and they produce high outputs at exclusive-NOR gates E40 pin 4 and E40 pin 10. All 12 exclusive-NOR gate outputs are high and their common output connection puts a high on pin 9 of E42. The other input to this gate is not asserted because it represents the ANDing of bit A17 and master sync (MSYN L), which has not yet been asserted by the processor. Coincidentally, address bits A00, A01, and A02 have been asserted. Bit A00 is ignored because its state is irrelevant to the operation of the M7251 address selector. Bits A01 and A02 are discussed separately because they are decoded to indicate which KG11-A register has been addressed (Table 3-2).



A	B	C
0	0	1
0	1	0
1	0	0
1	1	1

Figure 3-2 Typical Jumper Connections in Address Decoder, Logic Diagram

Table 3-2
KG11-A Select Signals

Device Address	Lines A ⟨02:01⟩	Select Signal True (+3V)	Register Selected
770700	00	SELECT 0	status
770702	01	SELECT 2	BCC
770704	10	SELECT 4	data

Bits A02 and A01 are sent to separate Unibus receivers designated E37. The outputs of these receivers and their complements (via two gates E42) are sent to four 4-input type 8815 NOR gates E46 (2) and E47 (2). The gates are arranged so that only one has three low inputs for a particular combination of bits A02 and A01 (four combinations possible). The fourth input must be low to assert the gate; this fourth input is common to all four gates and is the output (pin 8) of E42. If the bus is clear, SSYN L is clear and the processor asserts MSYN L to the address selection logic which produces a high at the output (pin 14) of receiver E38. This high is ANDed with the high signal from the common connection of the exclusive-NOR gates to enable E42 pin 8 low. This signal enables one of four SELECT signals, depending on the states of A02 and A01. This low signal is also inverted by E41 pin 1 and asserts SSYN L from pin 4 of E41.

3.2.3 Gating Control Logic

The select signals and gating signals from the address selection logic are combined in a control logic network to provide signals to implement the following functions: read/write CSR; read the BCC register; and load the data register. Figure 3-3 is a simplified block diagram showing the interrelation of the address selection logic and the gating control logic. The gating control logic is simple but it is spread over sheets 2 and 3 of engineering drawing M7251. Supplemental simplified logic diagrams are used to assist in understanding the operation of the gating control logic.

Figure 3-4 shows the control signals generated during a DATI (read) operation (drawing M7251, sheet 3). A DATI bus transaction asserts IN H which is ANDed with SELECT 0 H or SELECT 2 H, depending on whether the CSR or the BCC register has been addressed. If SELECT 0 is asserted, a high signal is produced at E29 pin 8 to enable the Unibus drivers associated with the seven bits of the CSR that can be read. Enabling the drivers places the information on the bus data lines D00–D03 and D06–D08. If SELECT 2 is asserted, a high signal is produced at E29 pin 10 to enable the Unibus drivers associated with the 16-bit BCC register.

Figure 3-5 shows the control signals generated during a DATO (write) operation (drawing M7251, sheet 2). A DATO bus transaction asserts OUT LOW H which is ANDed with SELECT 0 H or SELECT 4 H, depending on whether the CSR or the data register has been addressed. If SELECT 0 H is asserted, a low signal is produced at the output (pin 11) of NAND gate E31. This signal initiates several control signals.

a. It is sent directly to the CLOCK input (pin 6) of function register E27 which allows bits D00–D03 to be written in the CSR.

b. Inversion of this signal by E33 pin 6 provides the clocking signal for the SEN flip-flop E36 pin 3 (SEN is bit D06 of the CSR).

c. This signal is sent to pin 10 of Unibus receiver E30 and is ANDed with the STEP bit (D05 L) to enable the STEP control logic. This logic is discussed in a subsequent paragraph.

(continued on next page)

d. This signal is sent to pin 12 of Unibus receiver E30 and is ANDed with the CLEAR bit (D04 L) to enable the CLEAR logic for the DONE flip-flop. This logic is discussed in a subsequent paragraph. The output (pin 13) of E30 is also inverted by E29 pin 12 and is sent to the reset (R_D) inputs of the BCC register (E4, E10, E16, and E22) to clear it.

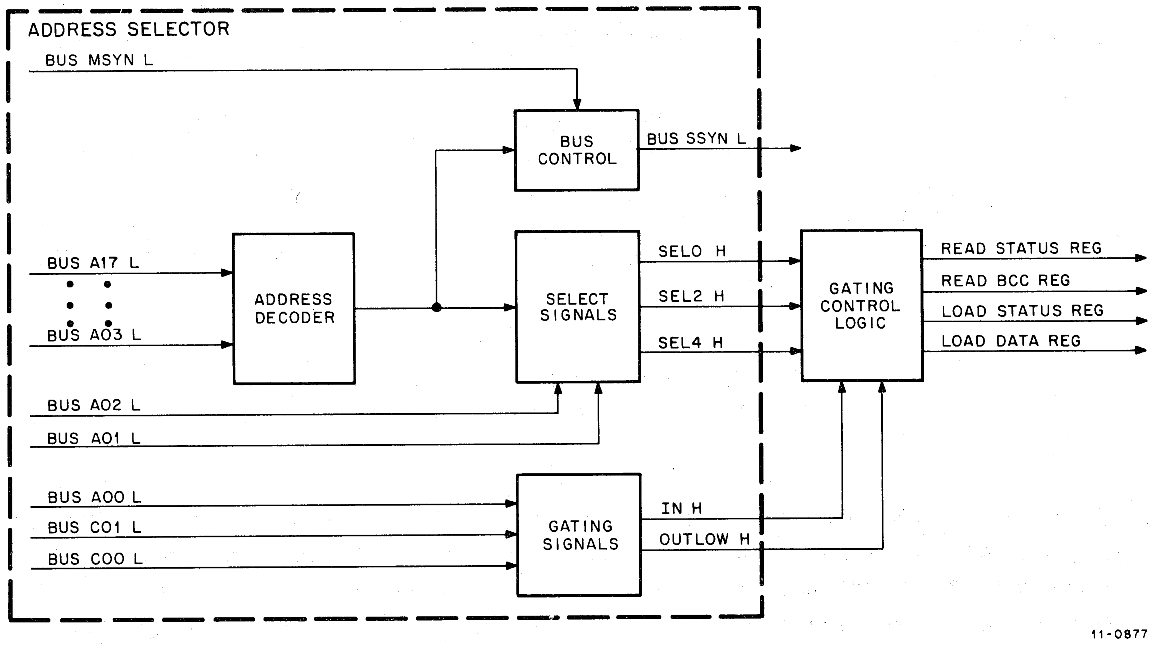
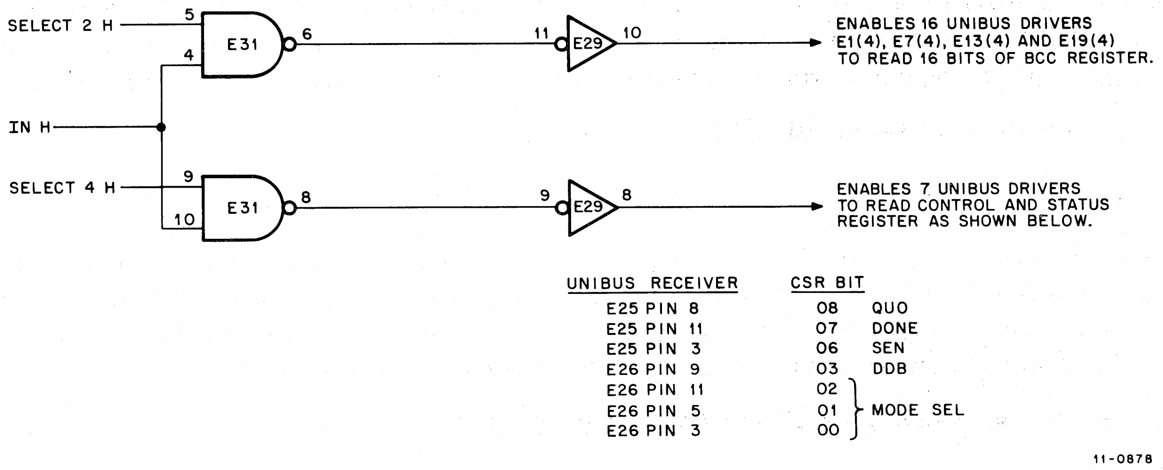


Figure 3-3 Address Selection Logic and Gating Control Logic, Block Diagram


SELECT 2 H
5
4
E31
6
11
E29
10
ENABLES 16 UNIBUS DRIVERS E1(4), E7(4), E13(4) AND E19(4) TO READ 16 BITS OF BCC REGISTER.
IN H
SELECT 4 H
9
10
E31
8
9
E29
8
ENABLES 7 UNIBUS DRIVERS TO READ CONTROL AND STATUS REGISTER AS SHOWN BELOW.
11-0878

UNIBUS RECEIVER	CSR BIT	
E25 PIN 8	08	QUO
E25 PIN 11	07	DONE
E25 PIN 3	06	SEN
E26 PIN 9	03	DDB
E26 PIN 11	02	MODE SEL
E26 PIN 5	01	
E26 PIN 3	00	

Figure 3-4 Control Signals Generated During DATI (Read) Operation, Logic Diagram

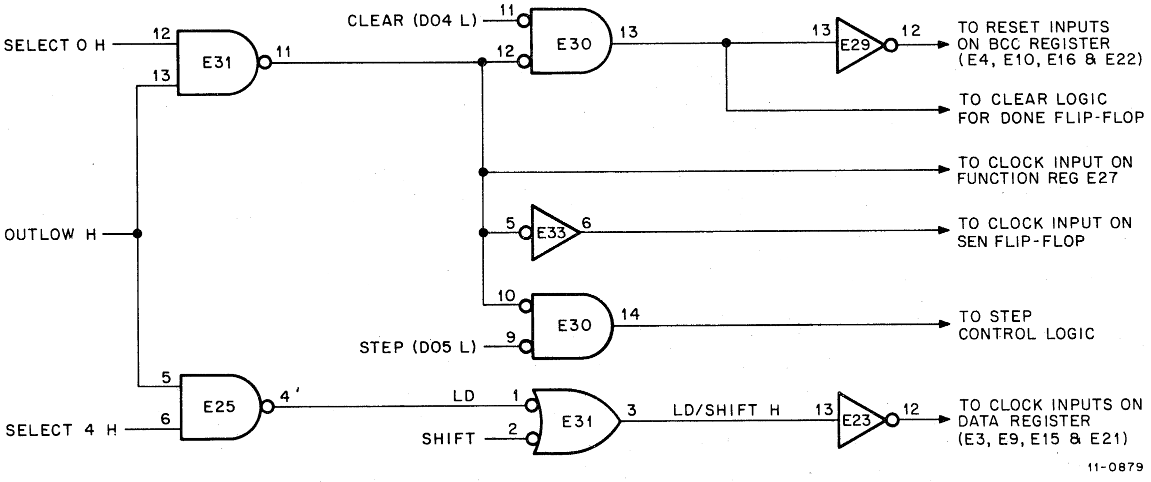

SELECT 0 H
12
13
E31
11
CLEAR (D04 L)
11
12
E30
13
13
E29
12
TO RESET INPUTS ON BCC REGISTER (E4, E10, E16 & E22)
TO CLEAR LOGIC FOR DONE FLIP-FLOP
TO CLOCK INPUT ON FUNCTION REG E27
OUTLOW H
5
E33
6
TO CLOCK INPUT ON SEN FLIP-FLOP
10
9
E30
14
STEP (D05 L)
TO STEP CONTROL LOGIC
5
6
SELECT 4 H
E25
4'
LD
1
SHIFT
2
E31
3
LD/SHIFT H
13
E23
12
TO CLOCK INPUTS ON DATA REGISTER (E3, E9, E15 & E21)
11-0879

Figure 3-5 Control Signals Generated During DATO (Write) Operation, Logic Diagram

If SELECT 4 is asserted, a low signal (LD) is sent to pin 1 of high-speed NAND gate E31. This gate is shown as a negative-input OR which is logically equivalent to a NAND; however, it identifies the asserted state as a high, which results when either or both inputs are low. The other input is the SHIFT signal from the shift control logic, which is discussed later. The output (pin 3) of E31 is inverted by E23 pin 12 and is the clock signal for the data register (E3, E9, E15, and E21).

Figure 3-6 shows the CSR read/write functions with the control logic and associated Unibus receivers and drivers.

3.3 CONTROL AND STATUS REGISTER

The control and status register (CSR) is a 16-bit register that provides various control and status signals related to the operation of the KG11-A. Only nine bits are used (0–8): of these, five are read/write (0–3 and 6); two are read-only (7 and 8); and two are write-only (4 and 5). The bit assignments for the CSR are shown in Figure 3-7.

Table 3-3 contains a detailed discussion of the function of each CSR bit.

3.4 DATA REGISTER

The data register is a 16-bit write-only register that is loaded with the data on which the BCC is calculated. The number of data bits, right-justified, shifted out of the register is a function of the polynomial selected and the state of the double-data-byte (DDB) bit: one character is shifted out when DDB=0 and two characters are shifted out when DDB=1. Figure 3-8 shows the data register format for the various polynomials.

Note that 2-character loading with CRC-12 (DDB=1) is not illustrated.

The data register is composed of four 4-bit shift registers (type 8271). They are connected to provide synchronous clocking, synchronous clearing, parallel loading and serial right shifting (drawing M7251, sheet 3). The four devices are designated E3, E9, E15, and E21. The parallel input data terminals of each device are designated D_A, D_B, D_C, and D_D. Input D_A on E3 is the register MSB; input D_D on E21 is the register LSB. The corresponding outputs are A_O, B_O, C_O, D_O, and $\bar{D}_O$ which is the complement of output D_O. The only device output used is D_O and it is connected to the serial data input D_S of the succeeding device. The data register output is the LSB (D00)

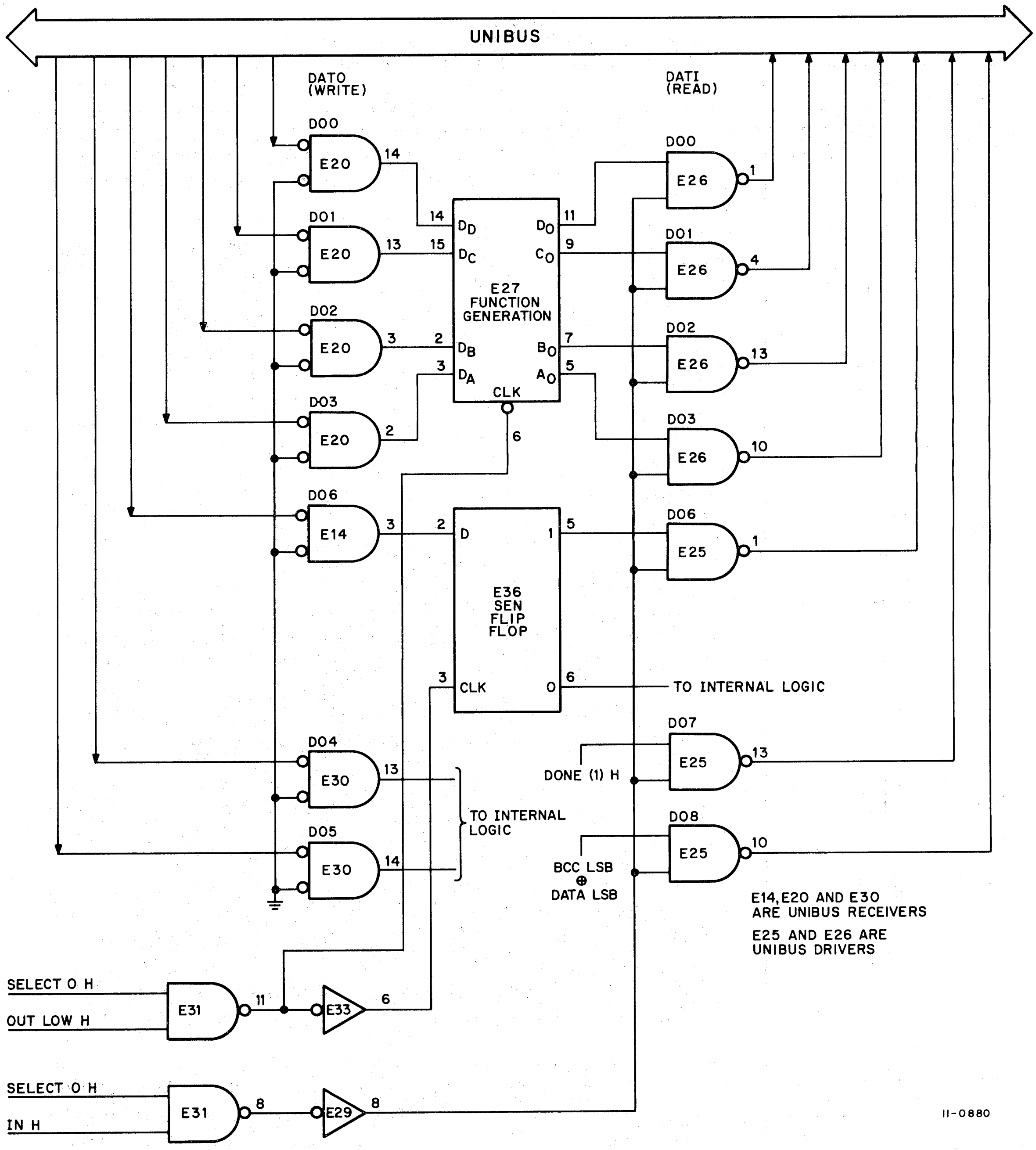


Figure 3-6 Control and Status Read/Write Functions, Logic Diagram

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
NOT USED							QUO	DONE	SEN	STEP	CLR	DDB	MODE SEL		
							R	R	R/W	W	W	R/W	R/W	R/W	R/W

11-0881

Figure 3-7 Control and Status Register Bit Assignments

Table 3-3

Control and Status Register Bit Functions

<table>
<tr><th colspan="2">Bit</th><th>Function</th></tr>
<tr><td>02:00</td><td>MODE SEL</td><td>Selects the CRC or LRC polynomial to be generated. Eight combinations are possible but only five are used. The other three generate undefined results.
<table>
<tr><th>Polynomial
Generated</th><th>Bit 02
CRC I/C</th><th>Bit 01
LRC</th><th>Bit 00
16 BCC</th></tr>
<tr><td>CRC-12</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>CRC-16</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>LRC-8</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>LRC-16</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>Undefined</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>CRC-CCITT</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>Undefined</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>Undefined</td><td>1</td><td>1</td><td>1</td></tr>
</table>
Set by the program. Cleared by INIT. Read/write.</td></tr>
<tr><td>03</td><td>DDB</td><td>Allows one or two characters (16 bits maximum) to be loaded into the data register. When cleared (0), the KG11-A accepts one character; when set (1), two characters are accepted. The KG11-A must receive characters in the same order as the communications line; therefore, in the double-data-byte mode (DDB=1) the first character to/from the communications line must be right-justified. Set by the program. Cleared by INIT. Read/write.</td></tr>
<tr><td>04</td><td>CLR</td><td>When set, clears the BCC register and presets the DONE flip-flop. This ensures that a new BCC accumulation starts with all 0s in the BCC register. Set by program. Write-only.</td></tr>
<tr><td>05</td><td>STEP</td><td>When set, provided the SEN bit (06) is cleared, the KG11-A performs one shift and exclusive-OR; the BCC register and data register are single-stepped by the program. The step mode is a diagnostic tool which allows the operation of the KG11-A to be checked by reading the QUO bit (08) and the BCC register after each step operation. Set by the program. Write-only.</td></tr>
<tr><td>06</td><td>SEN</td><td>When set, the KG11-A performs a complete cycle each time the data register is loaded. When cleared, provided the STEP bit (05) is set, the KG11-A is placed in the step mode. Set by the program. Cleared by INIT. Read/write.</td></tr>
<tr><td>07</td><td>DONE</td><td>Controls the operating mode of the data register. When set, the data register can be loaded (16 bits in parallel). DONE is set when the data register has shifted the required number of times; when the CLR bit (04) is set; when INIT is asserted. When cleared, the data register can be shifted right. DONE is cleared by the selection gating logic after the data register is loaded but prior to removal of the selection signal. Read-only.</td></tr>
<tr><td>08</td><td>QUO</td><td>This bit, called the Quotient, is the result of the exclusive-OR of the LSB of the data register and the LSB of the BCC register. It is used for diagnostic purposes during the step mode. This bit is cleared by INIT. Read-only.</td></tr>
</table>

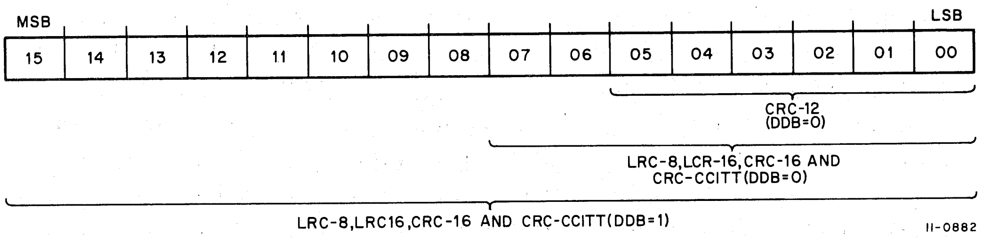


Figure 3-8 Data Register Format

from output D_O of E21. The register is cleared by INIT L applied to the direct reset ($\bar{R}_D$) input of each device. The register is clocked on the negative-going transition of the clock pulse which is the inverted LD/SHIFT H signal from pin 12 of high-speed inverter E23. The clock pulses originate in the logic shown in drawing M7251, sheet 2. This logic is discussed in detail in subsequent paragraphs. Briefly, a clock pulse is generated when the data register is addressed (SELECT 4 H and OUT LOW H are asserted). This pulse loads the register with the 16 data bits on the data lines (D ⟨15:00⟩) via Unibus drivers. After the register is loaded, clock pulses are generated to shift the data register a specific number of times. The mode of operation of the data register is controlled by the load (LD) and shift (SFT) inputs. Type 8271 shift registers can be operated in four modes, depending on the status of LD and SFT. In this application, only two modes of operation are used and the controlling factor is the shift (SFT) input. The load (LD) input of each device is connected to +3V which holds it high (1). The shift (SFT) inputs are all tied to DONE (1) L which is the 0 output of the DONE flip-flop. When the DONE flip-flop is set, DONE (1) L = 0 and the parallel load mode is selected; the serial shift right mode is selected when the DONE flip-flop is cleared and DONE (1) L = 1.

3.5 BLOCK CHECK CHARACTER (BCC) REGISTER

The block check character (BCC) register is a 16-bit read-only register which maintains the accumulation (block check character) derived from the shifting operations. The format of the data in the BCC register is a function of the polynomial used to perform the accumulation. Figure 3-9 shows the BCC register format for the various polynomials.

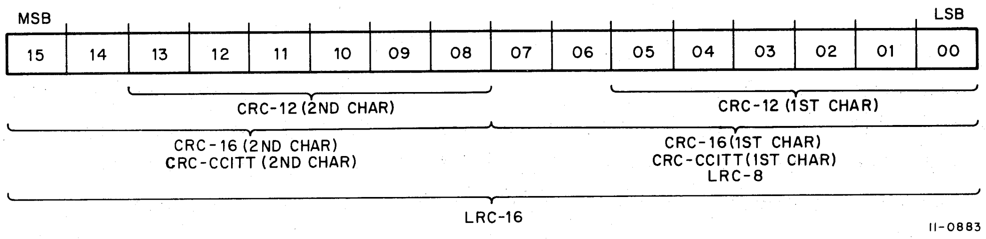


Figure 3-9 BCC Register Format

The BCC register is composed of four 4-bit shift registers (type 8271) designated E4, E10, E16, and E22. They are connected to provide synchronous clocking and synchronous clearing (drawing M7251, sheet 3). The register is cleared by CLEAR BCC L which is generated when the CLEAR bit (D04) is set or INIT is asserted. The CLEAR BCC L signal is sent to the direct reset ($\bar{R}_D$) input of each device. The register is clocked on the negative-going transition of the clock pulse which is the output (pin 6) of inverter E17 (drawing M7251, sheet 2). This clock pulse originates in logic that is discussed in detail in subsequent paragraphs. The BCC register is operated in two modes only and the controlling factor is the shift (SFT) input; the load (LD) input of each device is connected to +3V which holds it high (1). The shift inputs of E10 (bits D11 through D08) and E22 (bits D03 through D00) are connected to LRC (1) H which is mode-select bit D01 from E27. When LRC (1) H = 1, the serial shift right mode is selected; the parallel load mode is selected when LRC (1) H = 0. The shift input of E16 (bits D07 through D04) is connected to 16 BCC (1) H. When 16 BCC (1) H = 1, the serial shift right mode is selected; the parallel load mode is selected when 16 BCC (1) H = 0. Signal LRC (1) H is inverted and applied to pin 2 of NAND gate E18 along with CRC I/C L which is mode-select bit D02 from E27. The output (pin 3) of E18 is sent to the shift input of E4 (bits D15 through D12). When either or both inputs of E18 are low, its output is high and E4 is placed in the serial shift right mode. Both inputs must be high to select the parallel load mode.

A different feedback connection is used to accumulate the BCC for each polynomial. Sample accumulations and hardware configurations are discussed in detail in Appendix A.

3.6 CONTROL SIGNALS GENERATED BY THE CSR

3.6.1 Introduction

The first seven bits (D06–D00) of the CSR can be written into to perform certain functions (Table 3-3). The following paragraphs discuss the logic that generates the signals required to perform the functions.

3.6.2 Mode Selection Bits D02:00 and DDB Bit D03

Bits D ⟨03:00⟩ are sent to the inputs of E27 which is a 4-bit shift register (type 8271). The shift (SFT) input is connected to ground (logical 0) and the load (LD) input is connected to +3V (logical 1). As a result of these connections, E27 operates only in the parallel load mode (drawing M7251, sheet 2). The input data is transferred to the output on the negative-going clock transition. The device is clocked when the CSR is addressed during a DATO operation (SEL 0 H and OUT LOW H are both high). Bit-to-signal correspondence is:

Bit	Signal
D00	16 BCC (1)
D01	LRC
D02	CRC I/C
D03	DDB

A simplified logic diagram of E27 and associated gates is shown in Figure 3-10. DDB H (E27 pin 5) goes to the shift control logic where it and its complement (DDB L) are used (E34 pins 3 and 4). Signal 16 BCC (1) H goes to the shift input of BCC register E16 (bits D07 through D04) and pin 12 of NAND gate E18, which is in the line to the serial data input (D_S) of BCC register E4 (bits D15 through D12). Along with its complement, output $\bar{D}_O$ (pin 12) of E27, 16 BCC (1) H also goes to the feedback selection logic. LRC H goes to the shift inputs of BCC register E10 (bits D11 through D08) and E22 (bits D03 through D00). It is inverted by E23 pin 10 and is sent as LRC (1) L to the feedback selection logic. LRC (1) L is also applied to NAND gate E18 pin 2. The other input (pin 1) of E18 is CRC I/C H. When either or both inputs are low, E18 is enabled high to the shift input of BCC register E4 (bits D15 through D12). CRC I/C H is also sent to the feedback selection logic.

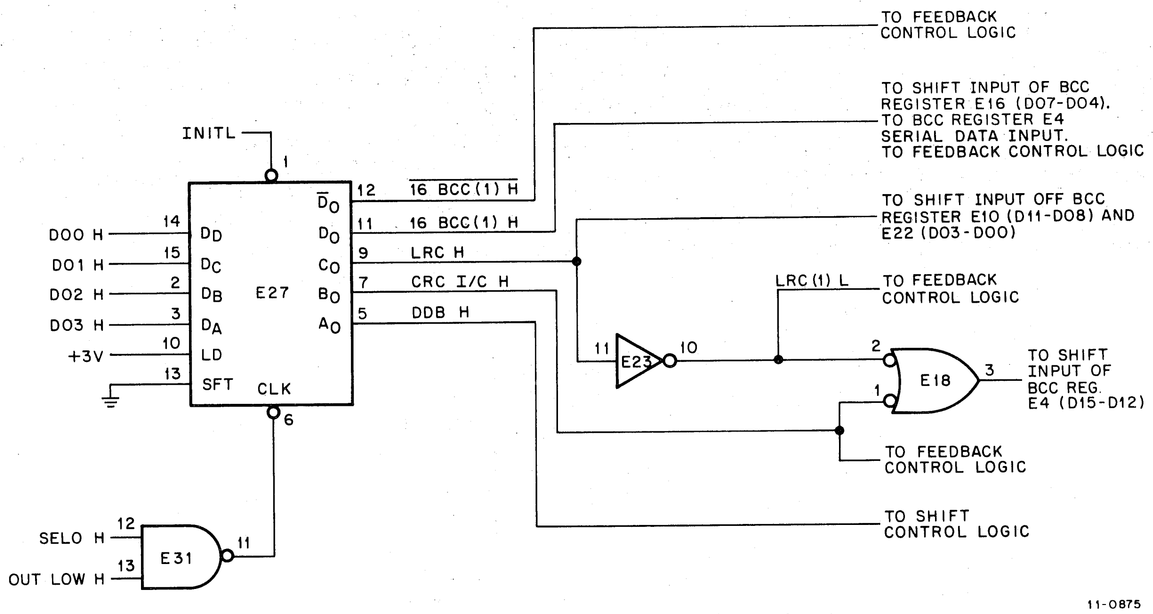

TO FEEDBACK CONTROL LOGIC
TO SHIFT INPUT OF BCC REGISTER E16 (D07-D04).
TO BCC REGISTER E4 SERIAL DATA INPUT.
TO FEEDBACK CONTROL LOGIC
INITL
1
TO SHIFT INPUT OFF BCC REGISTER E10 (D11-D08) AND E22 (D03-D00)
12
16 BCC (1) H
11
16 BCC (1) H
9
LRC H
7
CRC I/C H
5
DDB H
D00 H
D01 H
D02 H
D03 H
+3V
14
15
2
3
10
13
DD
DC
DB
DA
LD
SFT
E27
CLK
DO
DO
CO
BO
AO
6
LRC (1) L
TO FEEDBACK CONTROL LOGIC
11
E23
10
2
1
E18
3
TO SHIFT INPUT OF BCC REG. E4 (D15-D12)
TO FEEDBACK CONTROL LOGIC
TO SHIFT CONTROL LOGIC
SELO H
OUT LOW H
12
13
E31
11
11-0875

Figure 3-10 Mode Selection Control Signals, Logic Diagram

For a simple example, assume that polynomial LRC-16 has been selected to provide the following status bit states (Table 3-3):

CRC I/C H (D02) = 0
LRC H (D01) = 1
16 BCC (1) H (D00) = 1

Polynomial LRC-16 requires that all 16 bits in the BCC register be shifted. This means that E4, E10, E16, and E22 must all be in the serial shift right mode (shift input = 1). Signal 16 BCC (1) H = 1 which puts E16 (bits D07 through D04) in the shift mode. LRC H = 1 performs the same function for E10 (bits D11 through D08) and E22 (bits D03 through D00). LRC H is inverted and sent to pin 2 of E18 as a low (logical 0) signal which enables the gate high to put E4 (bits D15 through D12) in the shift mode. These status signals are also sent to the feedback control logic to set up the correct feedback path for the BCC register.

3.6.3 CLEAR Bit D04

CLEAR bit D04, a write-only bit, is set by the program to clear the BCC register and preset the DONE flip-flop. This ensures that a new BCC accumulation starts with all 0s in the BCC register and that the data register is in the parallel load mode. The initialize (INIT) signal from the processor performs the same functions: INIT is generated during power-up and when the processor console START switch is activated. The logic that implements the clear and initialize functions is shown in drawing M7251, sheet 2. A simplified logic diagram is shown in Figure 3-11.

When the program sets the CLEAR bit, D04 L = 0 at the input (pin 11) of Unibus receiver E30. The other input (pin 12) of this gate is also 0 because it is connected to the output (pin 11) of NAND gate E31. Both inputs of this gate are 1s (SELECT 0 H and OUT LOW H) because the CSR has been addressed during a write (DATO) operation. The high output (pin 13) of E30 is inverted by E29 pin 12 to produce CLEAR BCC L which clears the BCC register. The high output (pin 13) of E30 also goes to pin 6 of NOR gate E30 and enables it. The low output (pin 2) of E30 presets the DONE flip-flop E36 via its preset input (pin 10). These two functions (clearing BCC register and presetting DONE flip-flop) are also performed by the initialization (INIT) signal from the processor. When asserted, INIT = 0 at pin 5 of Unibus receiver E30. It produces a high at output pin 3 of E30 which enables E30 pin 2 low and presets the DONE flip-flop. This high signal is inverted by E29 pin 2 to produce CLEAR BCC L which clears the BCC register. The outputs of both inverters (E29 pin 12 and E29 pin 2) are connected together; this common point is CLEAR BCC L. These inverters are 7416 open-collector type and are connected in a wired-OR configuration. The output is 0 when one or both inputs equal 1. CLEAR BCC L = 0 when either CLEAR or INIT is asserted. The high output (pin 3) of E30, when INIT is asserted, is inverted by E29 pin 4 to generate INIT L = 0. This signal clears the data register, SEN flip-flop E36, and E27.

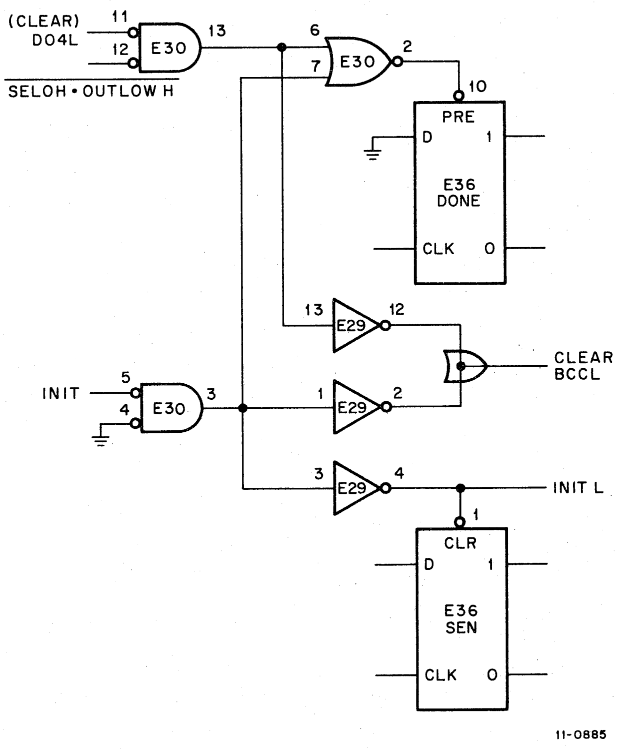


Figure 3-11 Clear and Initialize Control Signals, Logic Diagram

3.6.4 STEP Bit D05 and Shift Enable (SEN) Bit D06

The STEP bit and SEN bit are used to place the KG11-A in the single-step mode, which is a diagnostic tool. Normally, the KG11-A performs a complete cycle each time the data register is loaded. If the SEN bit (D06) is cleared, the KG11-A performs one shift and exclusive-OR operation each time the STEP bit (D05) is set. This single-step mode allows the performance of the KG11-A to be checked by examination of the contents of the BCC register and the state of the quotient (QUO) bit D08 after each step.

The STEP and SEN bit logic interacts with the shift clock and shift control logic. In the single-step mode, the shift clock output is inhibited. This discussion covers primarily the single-step mode; details of the shift clock and shift control logic are discussed in detail in Paragraph 3.6.5.

The STEP and SEN bit logic is shown in drawing M7251, sheet 2; however, a simplified logic diagram is shown in Figure 3-12. The two modes of operation established by the STEP and SEN bits are listed below.

Operating Mode	SEN (D06)	STEP (D05)
cycle	1	X
single step	0	1

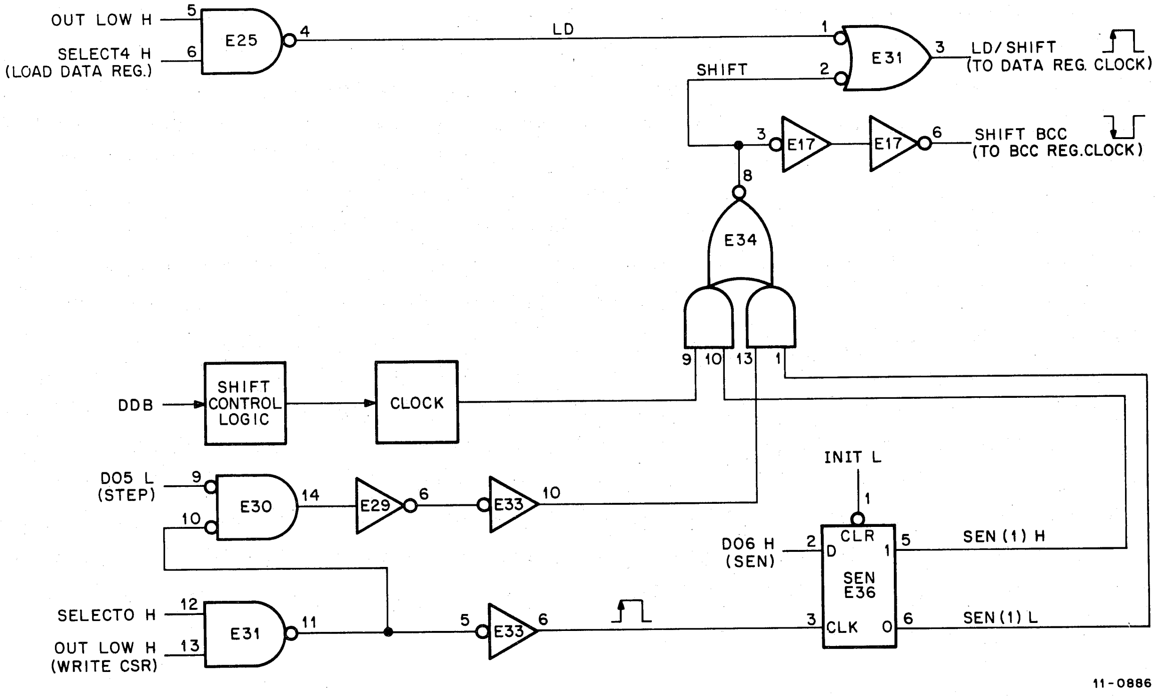


Figure 3-12 STEP and SEN Control Signals, Logic Diagram

First, assume that it is desired to operate the KG11-A in the normal cycle mode. The program addresses the CSR and sets the SEN bit, which provides a high signal (D06 = 1) on the D-input (pin 2) of SEN flip-flop E36. A write operation for the CSR asserts SELECT 0 H and OUT LOW H at the inputs of NAND gate E31 pins 12 and 13. The low output (pin 11) of E31 is inverted by E33 pin 5 and the positive-going edge of the level transition clocks SEN flip-flop E36 which sets it. The 1-output of E36 [SEN (1) H] is high and is sent to input 10 of AND-OR-invert gate E34 which qualifies this AND section. The 0 output of E36 (pin 6) is low and is sent to input 1 of AND-OR-invert gate E34, which disqualifies this AND section. The other input (pin 13) of this AND section is connected to the STEP bit (D05) via inverters E33 and E29 and receiver E30. The state of STEP is irrelevant in this case.

The program drops the CSR address and now specifies the loading of the data register by asserting SELECT 4 H and OUT LOW H. These signals are sent to the inputs (pin 5 and pin 6) of E25. The low output (pin 4) of E25, identified as LD, is sent to pin 1 of E31 which enables this gate (LD/SHIFT H = 1). The positive-going edge of this signal clocks the data register which loads it. The low signal from E25 pin 4 clears the DONE flip-flop which puts the data register in the serial shift right mode. This low signal also inhibits the shift clock. The program now drops the data register address. This activates the clock whose output is connected to input 9 of AND-OR-invert gate E34. The repetitive clock pulses are propagated through E34 pin 8 to pin 2 of E31. The output of this gate is LD/SHIFT H and its positive-going transitions clock the data register. The clock pulses from E34 pin 8 are double-inverted by E17 pin 4 and E17 pin 6 whose output is SHIFT BCC. The negative-going transitions of SHIFT BCC clock the BCC register. This process continues until the shifting ceases. The cycle is completed and the new block check character is in the BCC register.

To operate the KG11-A in the single-step mode, the program addresses the CSR and clears the SEN bit, which places a low signal (D06 = 0) on the D-input of SEN flip-flop E36. As in the previous example (cycle mode), the SEN flip-flop is clocked, which clears it. The 1 output of E36 [SEN (1) H] is low and is sent to input 10 of AND-OR-invert gate E34 which disqualifies this AND section and blocks the passage of shift clock pulses through E34 pin 8. The 0 output of E36 (pin 6) is high and is sent to input 1 of AND-OR-invert gate E34 which qualifies this AND section. The program drops the CSR address and now loads the data register as described in the previous example of the cycle mode. The data register is placed in the serial shift right mode. The program now drops the data register address; the shift clock is activated but its output is inhibited by SEN (1) H = 0. To implement the single-step mode, the program addresses the CSR and sets the STEP bit which places a low (D06L = 0) on pin 9 of Unibus receiver E30. The other input (pin 10) of the gate is connected to the output (pin 11) of E31. This output goes low when a CSR write operation is specified; SELECT 0 H and OUT LOW H are asserted (logical 1). With both inputs (pins 9 and 10) of E30 low, its output (pin 14) is high. This signal is double-inverted by E29 pin 6 and E33 pin 11, and sent to input 13 of AND-OR-invert gate E34. This AND section of E34 is enabled and a low is produced at the output (pin 8) of E34, which enables LD/SHIFT H at pin 3 of E31 and SHIFT BCC at pin 6 of E17. These signals clock the data register and BCC register; actually, one shift and exclusive-OR operation has been performed. Each time the STEP bit is set by the program, a single step is performed. After each step, the program can read the contents of the BCC register and read the quotient (QUO) bit (D08) in the CSR to check the operation of the KG11-A. The QUO bit is the result of the exclusive-OR of the LSB of the data register and the LSB of the BCC register. This operation is performed at exclusive-OR gate E28 pin 6 (drawing M7251, sheet 3). The QUO bit can be read at Unibus driver E25 pin 10.

3.6.5 Shift Clock and Shift Control Logic

In performing three types of cyclic redundancy checks (CRC) and two forms of longitudinal redundancy checks (LRC), the KG11-A deals with 6-bit, 8-bit, and 16-bit characters. It has the capability of shifting the data register and BCC register 6, 8, or 16 times, depending on the function being processed. A shift clock and shift control logic provide the correct number of shifts. The control logic includes the shift counter E32; DONE flip-flop E36; gating logic E33, E34, and E35; and counter preset gates E17 and E18 (drawing M7251, sheet 2, lower left section). A simplified logic diagram is provided in Figure 3-13.

The shift clock is a simple 10-MHz RC type. It consists of NAND gate E35, inverter E33, and feedback capacitor C54 and resistor R7. The clock output, E33 pin 12, is fed back via C54 to input pin 9 of E35. The other two inputs to this 3-input NAND gate are the 0 output [DONE (1) L] of the DONE flip-flop and the output of E25 (pin 4), which is the ANDing of SELECT 4 H and OUT LOW H. The qualifying conditions for starting the clock are: SELECT 4 H = 0 (data register not selected), and DONE (1) L = 1 (DONE flip-flop cleared). The clock is stopped after the correct number of shift pulses by the DONE flip-flop, which is set by the overflow pulse from the shift counter.

For example, assume that polynomial LRC-8 is selected and that single 8-bit characters are to be loaded (DDB = 0). The SEN bit (D06) is set which places the KG11-A in the normal cycle mode. The program determines the mode selection bits for LRC-8: D00H = 0, D01H = 1, D02 H = 0, and D03H = 0 (single 8-bit character selection). The program selects a write CSR operation (SELECT 0 H and OUT LOW H are asserted), which clocks E27 and transfers the input data (D00H – D03H) to the output. Output pin 9 of E27 is LRC H (bit D01) and is

1. It is inverted by E23 pin 10 (sheet 3) and is sent as LRC (1) L = 0 to input 5 of NAND gate E18, which disqualifies the gate. The 1 output (pin 6) of E18 is inverted by E17 pin 12 and is sent, as a 0, to input D_C of shift counter E32. The other E18 input (pin 4) comes from the $\overline{D}_O$ output (pin 12) of E27 and is irrelevant in this case.

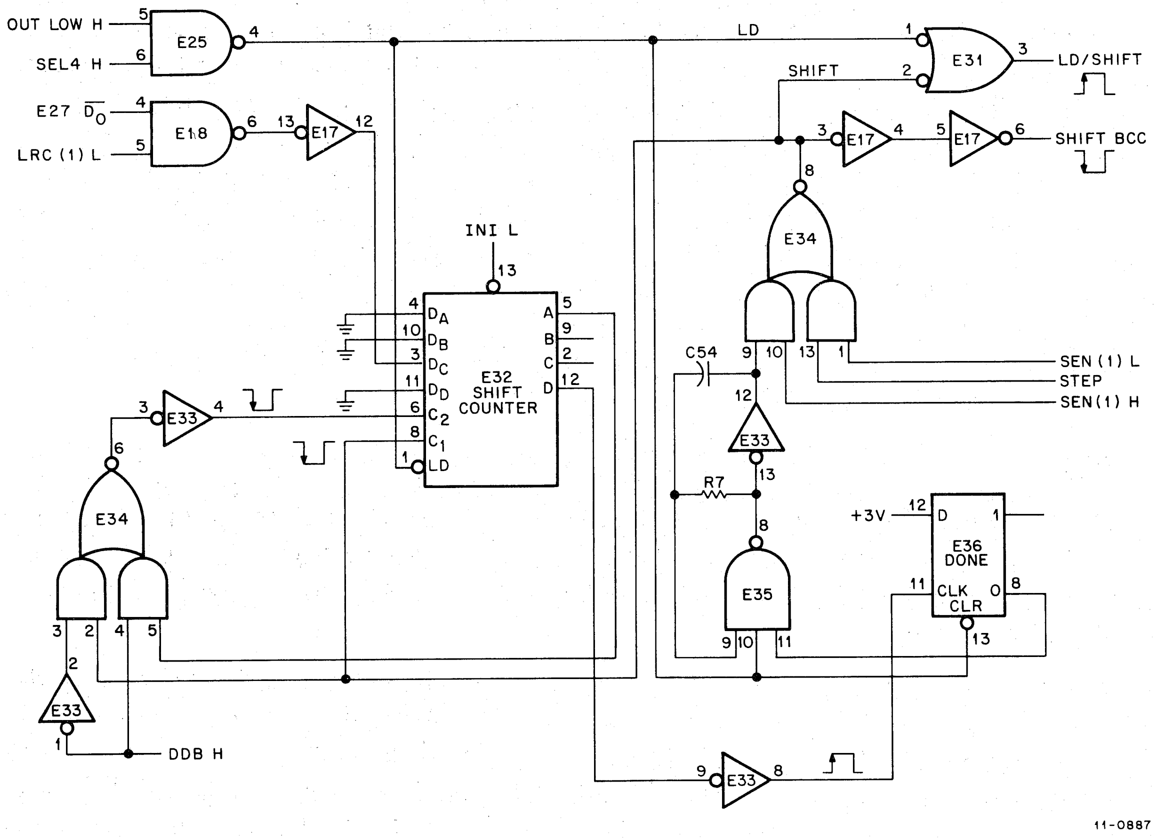


Figure 3-13 Shift Clock and Shift Control, Logic Diagram

At this point, the discussion diverges to explain the operation of shift counter E32. This device is a presettable counter/latch (type 74197). It provides a modulo 2 counter and a modulo 8 counter. Pin 4 (D_A) is the preset input of the modulo 2 counter which is clocked by negative-going pulses to clock input C_1 (pin 8); the output is A (pin 5). D_B, D_C, and D_D are the preset inputs to the modulo 8 counter which is clocked by negative-going pulses to clock input C_2 (pin 6); the outputs are B, C, and D. The modulo 2 counter cannot be preset because D_A is connected to ground (logical 0); its output has two states, 0 or 1. Inputs D_B and D_D of the modulo 8 counter are grounded and D_C is connected to control gating; therefore, it can be preset to 0 for a count of 8 or it can be preset to decimal 2 (D_C = logical 1) for a count of 6. Only outputs A and D are used in the control logic. When the count/load input (pin 1) is strobed low, the input data is transferred to the output (counter preset); the outputs will remain unchanged as long as the count/load input is high and the clocks are inactive. Pin 13 is a direct clear input which, when low (INIT L = 0), sets all outputs low regardless of the states of the clocks.

The discussion now returns to the point at which the modulo 8 counter input D_C is a 0 (via E17 pin 12). The program drops the write CSR operation and selects the load data register operation. SELECT 4 H and OUT LOW H are asserted (both = 1) which produces a low at the output (pin 4) of E25. This low enables LD/SHIFT H = 1 which clocks the information on the data lines D ⟨07:00⟩ to the output of the data register. It also clears the DONE flip-flop, which puts the data register in the serial shift right mode; there is enough delay in the circuit to ensure loading of the data register prior to its change to the shift mode. The low signal from E25 pin 4 is sent to the count/load input (pin 1) of the shift counter E32 which strobes the input data to the output; in this case, the counter is preset to 0. With the DONE flip-flop cleared, DONE (1) L = 1 and is sent to the shift clock input (E35 pin 11). Input 10 is still low, so the output (pin 8) of E35 is high and is fed back to input 9 via resistor R7. When the program drops the load data register operation, input 10 goes high and the clock starts. The oscillations are sustained by the RC network as long as inputs 10 and 11 remain high. The clock pulses are produced at the output (pin 12) of E33 at a rate of 10 MHz. The clock pulses are propagated through AND-OR-invert gate E34 via input 9. This AND section is qualified by a high at input 10 from the SEN flip-flop 1 output (pin 5) which is high because the flip-flop is set (normal cycle mode of KG11-A). The 0 output (pin 6) of the SEN flip-flop is low and is sent to input 1 of E34, which disqualifies this AND section. The clock pulses from E34 pin 8 enable LD/SHIFT H and SHIFT BCC to shift the data register and the BCC register. The modulo 2 counter is incremented by the clock pulses received at clock input C_1 (pin 8 of E32).

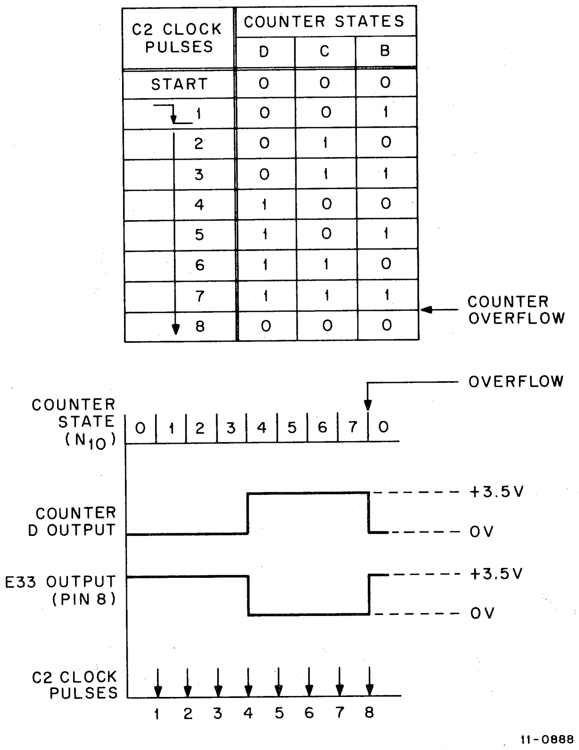
The clock pulses are also sent to input 2 of AND-OR-invert gate E34 which is qualified by a high from E33 pin 2. The input to this inverter is DDB = 0 which specifies that 8-bit characters are to be processed. The clock pulses are propagated through E34 pin 6, inverted by E33 pin 4, and sent to clock input C_2 (pin 6 of E32). The modulo 8 counter is incremented. Output A alternates between 0 and 1 and is sent to input 5 of the other AND section of E34. The state of the modulo 2 counter (output A) is irrelevant because this AND section is disqualified by a low on input 4 (DDB = 0).

The shift clock is running and each pulse causes the data register and BCC register to shift once; each pulse also increments the modulo 8 counter by 1. The counter started at 0 and the eighth clock pulse causes it to overflow (go to all 0s). This produces a negative-going edge at the D output (E32 pin 12) which is inverted by E33 pin 8; the resulting positive-going edge clocks the DONE flip-flop, which sets it. The 0 output of the DONE flip-flop [DONE (1) L], which is connected to the clock input (E35 pin 11), goes low and stops the clock. This signal also places the data register in the parallel load mode to await the next character. After the required eight shifts, the clock is stopped and is ready to begin the next cycle. Figure 3-14 shows the counter states and the generation of the overflow signal that turns off the clock.

If the example had selected polynomial CRC-12 with 6-bit characters to be handled (DDB = 0), the operating cycle would be the same except that the counter would be preset to 2_{10} and the clock would be stopped after six shift pulses.

To explain a 16-shift operation, assume that polynomial CRC-16 is selected and that two 8-bit characters are to be loaded (DDB = 1). The mode selection is made for CRC-16: D00H = 1, D01H = 0, D02H = 0, and D03H = 1 (two 8-bit character selection). The data register is loaded with 16 bits, the DONE flip-flop is cleared, the modulo 8 counter is preset to 0, and the shift clock is started. The clock pulses from E34 pin 8 enable LD/SHIFT H and SHIFT BCC to shift the data register and the BCC register. DDB = 1 and is sent to input 4 of AND-OR-invert gate E34, which qualifies this AND section; the other AND section of this gate is disqualified by the inverted DDB signal (0) applied to input 3. The clock pulses from E34 pin 8 are sent to clock input C_1 (pin 8 of E32). Each negative-going transition of the shift clock changes the state of the modulo 2 counter (output A). The output of this counter (pin 5 of E32) is sent to input 5 of AND-OR-invert gate E34. The pulses formed by the alternating state changes are propagated through E34 and E33 pin 4 to clock input C_2 (pin 6 of E32). Each negative-going transition increments the modulo 8 counter by 1. After eight C_2 clock pulses, the counter overflows (output

D = 0) and the shift clock is stopped. There is one negative-going transition of C_2 for two negative-going transitions of the shift clock; therefore, 16 shift operations have taken place during the eight counts initiated by clock C_2. Figure 3-15 shows the counter states and the generation of the overflow signal.



C2 CLOCK PULSES	COUNTER STATES		
	D	C	B
START	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0

Figure 3-14 Counter States and Overflow for Eight Shift Pulses

3.6.6 Exclusive-OR Operators and Feedback Logic

The KG11-A performs three types of cyclic redundancy checks (CRC) and two forms of longitudinal redundancy checks (LRC). The BCC register is used to accumulate the BCCs, but each of the five checks requires a specific configuration of exclusive-OR (X-OR) operators and feedback paths. A logic network provides the proper configuration for the check mode selected (drawing M7251, sheet 2). This paragraph describes the general arrangement of the logic; it does not describe detailed data flow. Notes in the upper-right corner of sheet 2 identify each type of check with a circled number (1 through 5). These circled numbers are placed at various points in the logic in sheet 2, and in the BCC register in sheet 3, to define the data paths. This convention, along with the BCC block diagrams and tabular shift patterns presented in Appendix A, should allow the reader to trace the data flow during a BCC accumulation.

The outputs of the X-OR and feedback logic are applied to the data inputs of the BCC register. A typical output signal is identified as BCC 5 IN; this signal goes to the bit 05 input of the BCC register. The majority of input signals to this logic come from the BCC register data outputs; for example, BCC 14 OUT. The mode select signals from E27 are used as inputs: BCC (1) H and its complement from the $\overline{D}_O$ output of E27; LRC H and its complement from E23 pin 10 (sheet 3); and CRC I/C H and its complement from E23 pin 2 (sheet 2). Another input is the exclusive-OR of the LSB of the data register and the LSB of the BCC register (BCC LSB ⊕ DATA REG) from E28 pin 6 (sheet 3).

C1 CLOCK PULSES	C2 CLOCK PULSES	COUNTER STATES		
		D	C	B
START	START	0	0	0
1		0	0	0
2	1	0	0	1
3		0	0	1
4	2	0	1	0
5		0	1	0
6	3	0	1	1
7		0	1	1
8	4	1	0	0
9		1	0	0
10	5	1	0	1
11		1	0	1
12	6	1	1	0
13		1	1	0
14	7	1	1	1
15		1	1	1
16	8	0	0	0

COUNTER OVERFLOW

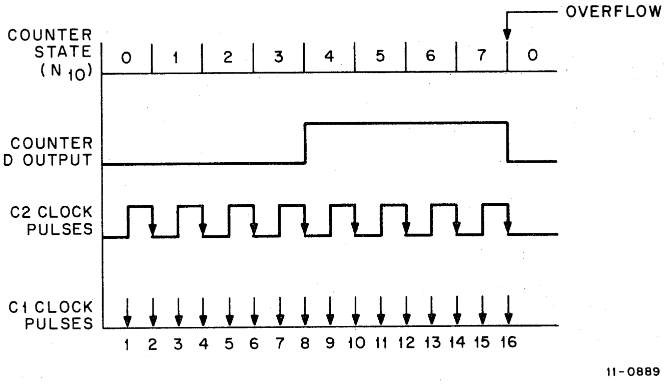
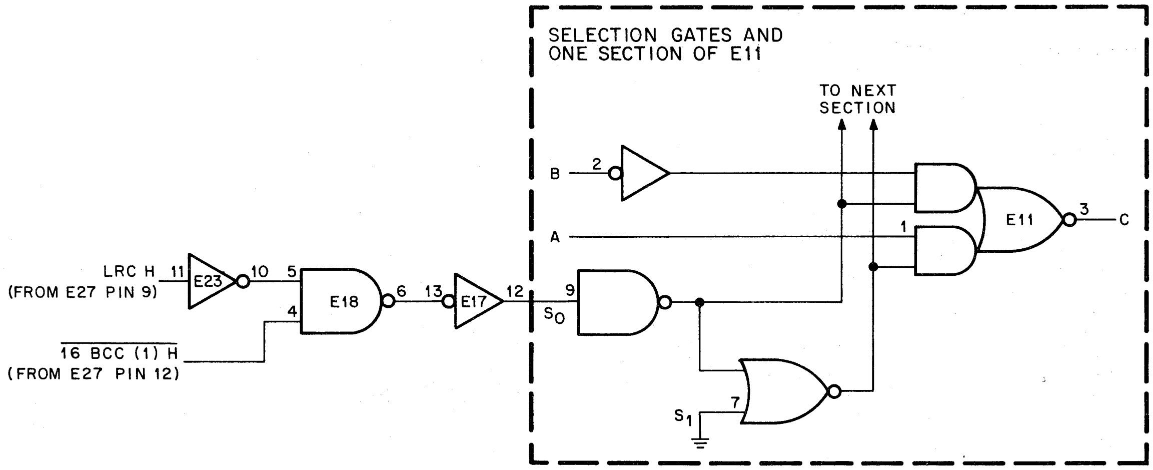


Figure 3-15 Counter States and Overflow for 16 Shift Pulses

Most of the IC devices used in the X-OR and feedback logic are self-explanatory (NAND gates, inverters, exclusive-OR gates, and AND-OR-invert gates); however, one MSI device (E11) requires a brief description. E11 is an 8266 2-Input 4-Bit Multiplexer (sheet 2, top center section). The multiplexer chooses from two different 4-bit inputs (pins 14, 11, 5, and 2, or pins 15, 10, 6, and 1); selection is controlled by inputs 7 and 9. A simplified logic diagram of one section of the multiplexer, along with the control signals, is shown in Figure 3-16. E11 contains four identical sections and two selection gates. Each section is an AND-OR-invert gate with an inverter in one input leg (Figure 3-16). Input A (pin 1) goes directly to one input of a 2-input AND gate; input B (pin 2) is inverted and then goes to one input of the other 2-input AND gate. The other input of each AND gate is connected to the selection gates so that, for a given state of S_0 (pin 9), their states are complementary. In this manner, only one AND gate is qualified by the selection logic at one time; the A leg is qualified when S_0 = 1 and the B leg is qualified when S_0 = 0. The B input is transferred to the output C (pin 3) in true form; the A input is transferred in complemented form. Selection input S_1 is grounded, so control is provided by the state of S_0 which is determined by mode selection bits D00 (16 BCC H) and D01 (LRC H). LRC H from output C_0 of E27 is inverted by E23 pin 10 (sheet 3) and ANDed with the complement of BCC (1) H (output $\bar{D}_0$ of E27) at E18 pins 1 and 4. The output (pin 6) of E18 is inverted by E17 pin 13 and applied to pin 9 (S_0) of E11.



FUNCTION	16 BCC (1) H	$\overline{\text{16 BCC (1) H}}$	LRC H	S_0	S_1	C (OUT)
① CRC-CCITT	1	0	0	0	0	B
② CRC-16	1	0	0	0	0	B
③ CRC-12	0	1	0	1	0	$\overline{A}$
④ LRC-8	0	1	1	0	0	B

11-0890

Figure 3-16 Control of 8266 Multiplexer in Feedback Circuit, Logic Diagram

CHAPTER 4
PROGRAMMING INFORMATION

4.1 INTRODUCTION

This chapter provides general programming information for software control of the KG11-A. For detailed PDP-11 programming information, refer to the *Paper-Tape Software Programming Handbook,* DEC-11-GGPA-D. This chapter discusses the various LRC and CRC codes implemented by the KG11-A and describes in detail a flow chart of the recommended programming practices.

4.2 LRC AND CRC CODES

4.2.1 Introduction

The KG11-A provides two forms of error detection: longitudinal redundancy checking (LRC) and cyclic redundancy checking (CRC). Two LRC codes are used (LRC-8 and LRC-16), and three CRC codes are used (CRC-12, CRC-16, and CRC-CCITT).

Longitudinal redundancy checking is the modulo 2 sum of the bits in each bit level of all characters in a message block to produce a check sum.

Cyclic redundancy checking is the serial binary division of a message by a constant which is the polynomial for the selected code. The quotient is discarded and the remainder is the check sum.

4.2.2 CRC-16

CRC-16 is applied to synchronous systems that use 8-bit characters. It is compatible with IBM Binary Synchronous Communications (BSC) when the transmission is extended binary-coded decimal interchange code (EBCDIC) or an 8-bit transparent code.

The following examples show the message format for an IBM-compatible system using normal transmission. Each character represents eight bits. The BCC accumulation is 16 bits; therefore, two BCC characters (8 bits each) are shown together. The first one represents the least-significant 8 bits of the BCC accumulation in the KG11-A.

Example 1:

```
S        EBB
T–text–TCC
X        BCC
```

Example 2:

```
S        EBB
T–text–TCC
X        XCC
```

Example 3:

```
S        IBB        EBB
T–text–TCC–text–TCC
X        BCC        BCC
```

Example 4:

```
S        IBB        EBB
T–text–TCC–text–TCC
X        BCC        XCC
```

In all four examples, the STX is not included in the BCC accumulation. In examples 1 and 2, the BCC includes the first text character through the ETB or ETX. In examples 3 and 4, the first BCC includes the first text character through the ITB. The second BCC starts with the character following the first BCC, even if it is an STX or DLE.

The following example shows the message format for an IBM-compatible system using transparent transmission.

```
**        *          *    |        *
DS        DD         DIBB | DS      DEBB
LT–text–LL–text– LTCC     | LT–text–LTCC
EX        EE         EBCC | EX      EBCC
                          |
                          |<—— New BCC computation
                          |    starts here
```

*characters not included in BCC computation

The first DLE STX sequence puts the system in the transparent mode. The DLE DLE sequence indicates that the second DLE is data rather than a control character; therefore, it is included in the BCC accumulation. That sequence may appear in text as often as the 8-bit DLE representation is required. The DLE ITB sequence takes the system out of the transparent mode; therefore, the DLE STX sequence following the BCC is included in the next BCC accumulation and it also puts the system back in the transparent mode.

4.2.3 CRC-12

CRC-12 is applied to synchronous systems that use 6-bit characters. The BCC accumulation is 12 bits. It is compatible with IBM Binary Synchronous Communications (BSC) when the transmission is 6-bit transcode. The characters included/excluded during the BCC accumulation are the same as for CRC-16; the only difference is in the character length which is 6 bits instead of 8 bits.

4.2.4 CRC-CCITT

CRC-CCITT is the standard used to compute a BCC for European systems. The characters included/excluded during the BCC accumulation depend on the line protocol used for the system in which the KG11-A is installed. When operating with 8-bit characters, the BCC accumulation is 16 bits.

4.2.5 LRC-8

LRC-8 is applied to systems that use characters of eight bits or less. Normally, LRC is used in combination with a parity bit check on each character. The parity check is generally called vertical redundancy checking (VRC). VRC requires 1 bit; therefore, it can only be used with a data character length of seven bits or less. The KG11-A applies LRC-8 to 8-bit characters only. The BCC is computed by performing an exclusive-OR (modulo 2 addition) of the bits in each bit level of all characters. LRC with VRC is compatible with IBM BSC when the transmission code is USASCII. Odd parity (VRC) is used with IBM systems.

4.2.6 LRC-16

LRC-16 is applied to systems that use characters of 16 bits or less. The KG11-A applies LRC-16 to 16-bit characters only.

4.3 RECOMMENDED PROGRAMMING PRACTICES

4.3.1 Introduction

It is recommended that the KG11-A compute the BCC on a message basis. The message is passed through the KG11-A in a continuous loop after being received or prior to the start of transmission. The BCC is computed and added to the end of the message to be transmitted; or the BCC is computed for the received data and compared to the received BCC.

The KG11-A can compute a partial BCC; i.e., on a character-by-character basis. This discussion deals primarily with the recommended practice (message basis); however, the partial BCC technique is described briefly. Details of the KG11-A addressing requirements and CSR bit assignments are not discussed; they are covered in detail in Chapter 3.

4.3.2 BCC Computation on a Message Basis

Figure 4-1 is a flow chart showing the recommended procedure for computing a BCC on a complete message. The steps are numbered on the flow chart to correspond to the discussion.

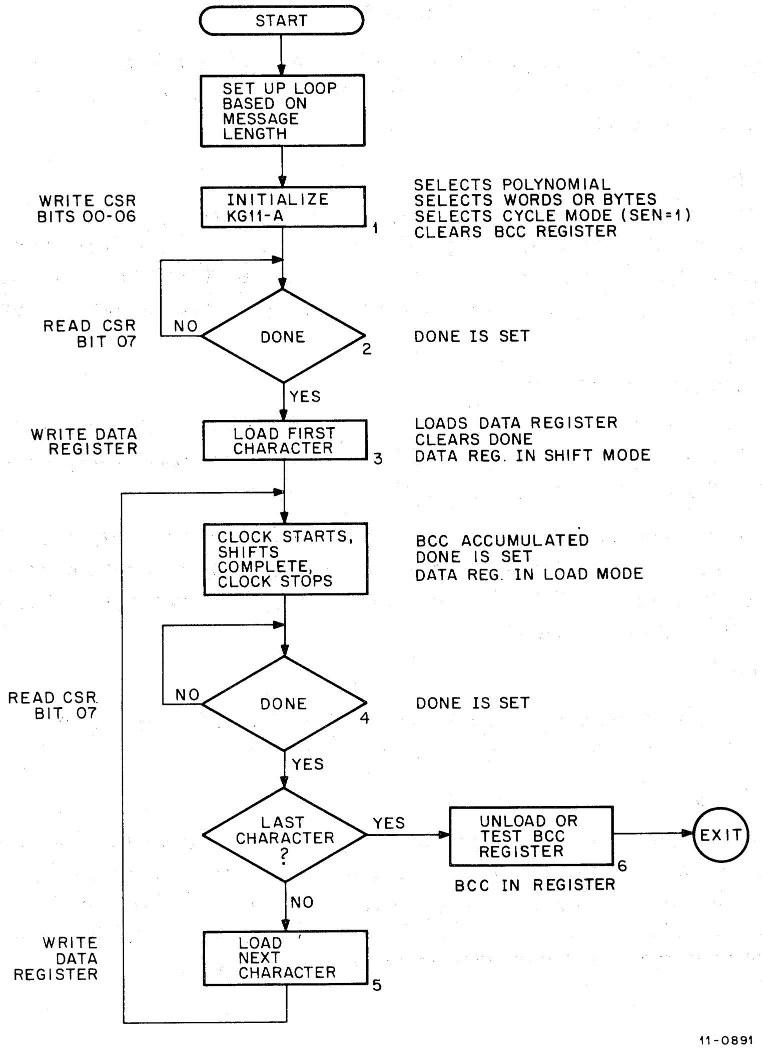


Figure 4-1 BCC Computation on a Message Basis, Flow Chart

1 **Initialization**

The initialization process consists of sending a control word to the CSR. The CSR is addressed and a DATO (write) operation is performed. The seven control bits of the CSR are chosen to select the desired conditions.

a. Bits 16 BCC (D00), LRC (D01), and CRC I/C (D02) select the polynomial to be used in computing the BCC. LRC and 16 BCC also preset the shift counter to 0 or 2, as required, to generate the correct number of shifts.

b. Bit DDB (03) determines whether bytes (8 bits) or words (16 bits) are to be loaded into the data register. Even if bytes are accumulated, the program loop is shorter if they are presented two at a time (word option).

NOTE

The KG11-A must receive characters in the same order that they are received or are to be transmitted via the communication line. If the message is formed in the byte mode, it can be moved to the data register in the word mode but the first byte must be right-justified (LSB to the right). The message must be stored in memory in ascending order of byte address.

c. Bits STEP (D05) and SEN (D06) select the mode of operation: cycle or single step. In this case, the cycle mode is selected; SEN = 1 and STEP is ignored.

d. Bit CLR (D04) clears the BCC register to ensure that a new BCC accumulation starts with all 0s in the BCC register. The CLR bit also presets the DONE flip-flop which places the data register in the parallel load mode.

2 **Test DONE Bit**

The read-only DONE bit is provided primarily for diagnostic testing purposes. It can be tested at this point to check on the completion of the initialization process. Testing is performed by addressing the CSR and performing a DATAI (read) operation. With current PDP-11 memory cycle times, the KG11-A is fast enough to complete its cycle while the program is testing to determine if more characters are to be accumulated.

3 **Load First Character into Data Register**

The program addresses the data register and a DATO (write) operation is performed. Information from the Unibus data lines is strobed in parallel to the output of the data register. Then the DONE flip-flop is cleared, which puts the data register in the shift mode.

When the address lines and DATO operation are dropped by the program, the clock starts. The cycle begins, and after the correct number of shifts, the clock is stopped and the block check character is in the BCC register. Also, the DONE flip-flop is set which puts the data register back in the parallel mode for the next character to be loaded.

4 **Test DONE Bit**

The DONE bit can be tested at this point to check that the shifting operation is complete, which means that the character has been added to the BCC accumulation.

5 **Load Next Character**

The process for loading additional characters is the same as that described for loading the first one (item 3). Each load operation results in the character being processed and added to the BCC accumulation in the BCC register. The loading process continues until all characters have been loaded. The complete message has been passed through the KG11-A and the message BCC is in the BCC register.

(continued on next page)

6 **Unload or Test the BCC Register**

If the BCC is for a message to be transmitted, the contents of the BCC register can be moved to the message buffer for subsequent transmittal.

If the BCC accumulated is the value of the BCC of a received message, it can be compared to the received one in the register. Alternatively, the BCC may be included in the accumulation, in which case a good BCC results in a 0 accumulation in the BCC register.

If the ITB character is used in the message (examples 3 and 4, Paragraph 4.2.2), the BCC that follows can be included in the accumulation and the result should be 0. If the rest of the message is accumulated without testing for 0, the final BCC will compare only if the intermediate BCC caused the accumulation to go to 0. (Data register and BCC register formats for the various polynomials are shown in Figures 3-8 and 3-9.)

4.3.3 Partial BCC Computation (Character-by-Character)

A partial BCC computation (character-by-character) is accomplished by adding a character to the accumulation as it is received. This can be done efficiently for a half-duplex system because the BCC can be left in the KG11-A until all the characters have been processed. For systems using full-duplex and for multiple lines, the BCC accumulation for a specific line cannot be left in the KG11-A because another line may be serviced before the next character appears on the original line. The partial BCC must be saved and reloaded when the next character appears. The following sequence is used to load a partial BCC, add a character, and store the new partial BCC.

Step	Procedure
1	Set mode to LRC-16 and clear the BCC register. The BCC register cannot be written into; the partial BCC must be loaded through the data register in the LRC-16 mode.
2	Load the partial BCC into the data register.
3	Test the DONE bit. This step is not required but it can be performed to check for completion of the loading operation. The KG11-A does not generate interrupts so there is no automatic completion indication.
4	Set mode to the selected polynomial. Do not clear the BCC register.
5	Load the character into the data register.
6	Test the DONE bit.
7	Store partial BCC accumulation.

4.3.4 Single-Step Operation

Figure 4-2 is a flow chart showing the procedure for operating the KG11-A in the single-step mode for testing purposes. The steps are numbered on the flow chart to correspond to the discussion.

1 **Initialization**

The initialization procedure is the same as that described in normal operating mode (Paragraph 4.3.2) except that the SEN bit is cleared to place the KG11-A in the step mode.

2 **Load Character into Data Register**

The character is loaded into the data register and the DONE flip-flop is cleared, which puts the data register in the shift mode. These operations are the same as those performed in the normal cycle mode. When the data register address is dropped, the shift clock starts but its output is inhibited by the SEN flip-flop. The operating cycle does not start. The data register is loaded but it and the BCC register cannot be clocked in the normal way.

(continued on next page)

3 **Set STEP Bit**

The program sets the STEP bit in the CSR and one pulse is generated that clocks the data register and BCC register. Actually, the KG11-A performs one shift and exclusive-OR operation. Each time the STEP bit is set, a single step is performed.

4 **Test QUO Bit**

The QUO bit is tested to check on the KG11-A operation. It is the result of the exclusive-OR of the LSB of the data register and the LSB of the BCC register.

5 **Test BCC Register**

The contents of the BCC register are tested to check on the KG11-A operation.

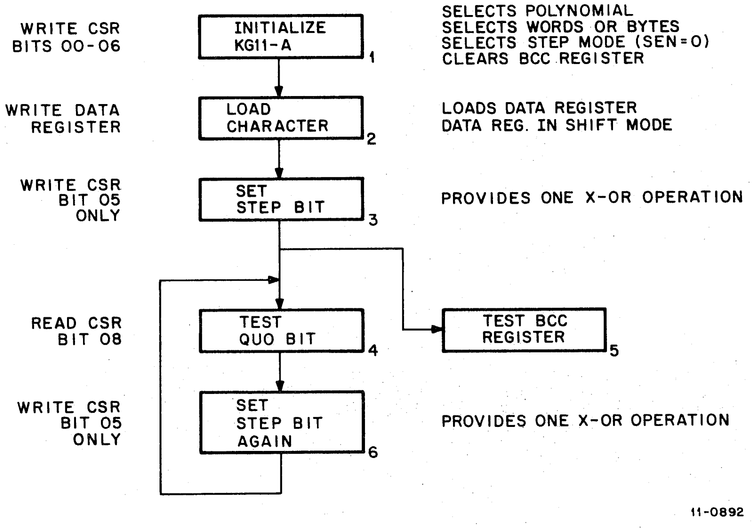


Figure 4-2 Step Mode Operation, Flow Chart

CHAPTER 5
MAINTENANCE

5.1 INTRODUCTION

As a maintenance tool, this manual is to be used as a combination training guide and reference document. This manual and the engineering logic drawings should provide the user with information necessary to analyze and correct equipment malfunctions.

This chapter references a diagnostic program and includes a description of test points that are available on the KG11-A.

5.2 TESTING AND DIAGNOSTICS

The minimum hardware required to perform an operational test on the KG11-A includes a PDP-11 processor with at least 4K of core memory, and one KG11-A module (M7251). The module can be installed in one of the two PDP-11 processor small peripheral controller slots, or in a DD11-A peripheral mounting unit.

Each KG11-A contains three registers that must be assigned consecutively as follows:

Address	Register
7707X0	status
7707X2	BCC
7707X4	data

Addresses are assigned for eight KG11-As per system. In the address designation X = 0 for the first KG11-A, X = 1 for the second, and X = 7 for the eighth.

After the hardware is assembled, testing is accomplished by loading and operating MainDEC-11-D8KA. This diagnostic program requires approximately 2 minutes per pass (for a PDP-11/20 processor). Five successful passes are considered to be a valid test.

Single-step operation of the KG11-A is a convenient diagnostic tool. It is described in Paragraphs 3.6.4 and 4.3.4.

5.3 TEST POINTS

Eight test points are provided on the KG11-A to assist in troubleshooting malfunctions. Each test point is brought out to a module contact pin that can be accessed at the corresponding wire-wrapped pin on the system unit into which the module is plugged.

Table 5-1 lists the location of the test point on the drawing, the associated module contact pin, and the function.

Table 5-1
KG11-A Test Points

Drawing Coordinates	Module Pin No.	Function
sheet 3 C-8	BA1	This signal is the output (pin 10) of inverter E29. It is the inverted output (pin 6) of NAND gate E31. The inputs (pins 4 and 5) of E31 are IN H and SELECT 2 H from the address selection logic. Both signals are high during a read operation on the BCC register; therefore, this test point is also high during the operation. This test point is used to verify IN H and SELECT 2, and to check out E31 and E29.
sheet 3 C-8	BB1	This signal is the output (pin 8) of inverter E29. It is the inverted output (pin 8) of NAND gate E31. The inputs (pins 9 and 10) of E31 are IN H and SELECT 0 H from the address selection logic. Both signals are high during a read operation on the CSR; therefore, this test point is also high during the operation. It is used to verify IN H and SELECT 0 H, and to check out E31 and E29.
sheet 2 B-7	BR1	This signal is the output (pin 11) of NAND gate E31. The inputs (pins 12 and 13) of this gate are SELECT 0 H and OUT LOW H from the address selection logic. Both signals are high during a write operation on the CSR; therefore, this test point is low during the operation. This test point is used to verify SELECT 0 H and OUT LOW H, and to check out E31.
sheet 2 B-7	AE1	This signal is the output (pin 4) of NAND gate E25. The inputs (pins 5 and 6) of this gate are OUT LOW H and SELECT 4 H from the address selection logic. Both signals are high during a write (load) operation on the data register; therefore, this test point is low during the operation. This test point is used to verify OUT LOW H and SELECT 4 H, and to check out E25.
sheet 2 B-7	AF1	This signal is the output (pin 8) of AND-OR-invert gate E34. It is the clock signal for the data register and the BCC register. During normal cycle operation, when the shift clock is running, a 10-MHz pulse train can be observed at this point. During single-step operation, a single negative pulse can be observed here when the STEP bit is set.
sheet 2 C-4	AM1	This signal is the output of the wired-OR connection of E29 pin 12 and E29 pin 2. It is the clear signal (CLEAR BCC L) for the BCC register. It is asserted low to clear the register when the program sets the CLEAR bit or the processor asserts INIT.
sheet 2 B-4	BS1	This signal is the output (pin 13) of bus receiver E30 which handles the CLEAR bit (D04L). It presets the DONE flip-flop when the program sets the CLEAR bit.
sheet 2 A-2	AK1	This signal is the output (pin 6) of inverter E29. It is the inverted output (pin 14) of bus receiver E30 (shown as negative-input AND). The inputs to this gate are the STEP bit (D05L) and the signal that is generated when a write operation is performed on the CSR. When the STEP bit is set by the program, this test point is low.

APPENDIX A
BCC COMPUTATION FOR LRC AND CRC

A.1 INTRODUCTION

This appendix contains a general discussion of longitudinal redundancy checking (LRC) and cyclic redundancy checking (CRC). It also contains specific examples of BCC computations using LRC and CRC methods. The examples are primarily graphic, using binary 1s and 0s to show the step-by-step shift pattern of the BCC register.

A.2 LONGITUDINAL REDUNDANCY CHECKING

LRC is the modulo 2 sum (exclusive-OR) of the bits in each bit position of all characters in a message block to produce a BCC. The following example shows the BCC computation for four 8-bit characters using LRC. Each character contains seven data bits and an odd-parity bit.

Bit Position

	P	6	5	4	3	2	1	0
Character 1	0	1	1	1	1	0	0	1
Character 2	1	0	0	1	1	0	0	0
Character 3	0	0	0	0	0	1	1	1
Character 4	0	1	1	1	0	0	0	0
BCC	1	0	0	1	0	1	1	0

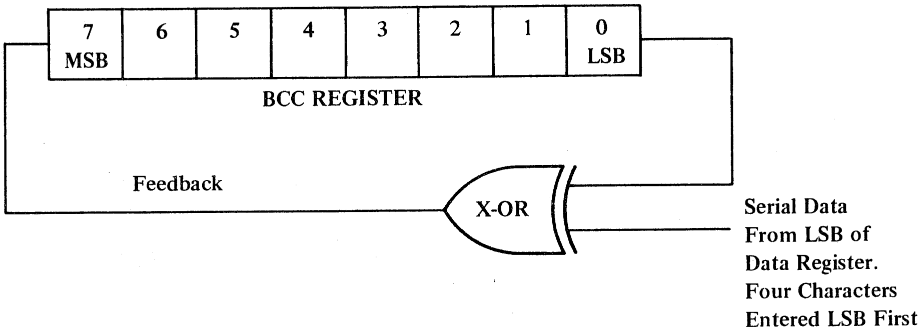
This is an example of LRC-8. It is implemented by the KG11-A and has a generating polynomial of $X^8 + 1$. The hardware implementation is cyclic and is illustrated in Figure A-1 along with a state chart for the BCC register during the BCC accumulation.

Four 8-bit characters are to be encoded. They are loaded into the data register in a right-justified format (LSB to the right). The LSB output of the data register is connected to one input of the exclusive-OR gate (X-OR); the other exclusive-OR input is connected to the LSB of the BCC register. With each clock pulse, the LSB of the data register is exclusive-ORed with the LSB of the BCC register and the result is fed back to the MSB of the BCC register. A 1 is fed back only if the exclusive-OR inputs are complementary; two 1s or two 0s at the inputs produce a 0. The chart in Figure A-1 shows the BCC register starting at all 0s and finishing after 32 shift pulses with the BCC accumulator for the four characters shown.

A.3 CYCLIC REDUNDANCY CHECKING

A.3.1 Introduction

The KG11-A generates a cyclic code called a BCC. In the case of a message to be transmitted, the BCC is sent after the data bits as part of the message. The receiving station, using a KG11-A or similar device, verifies the BCC. Verification assumes that the message is without error and the station accepts it as received. Non-verification results in the original message being retransmitted. Both stations must be using the same CRC code.



CHARACTER/BIT LOADED	SHIFT NO.	BCC REGISTER BITS							
		7	6	5	4	3	2	1	0
	Start	0	0	0	0	0	0	0	0
Char 1 Bit 1	1	1	0	0	0	0	0	0	0
Char 1 Bit 2	2	0	1	0	0	0	0	0	0
Char 1 Bit 3	3	0	0	1	0	0	0	0	0
Char 1 Bit 4	4	1	0	0	1	0	0	0	0
● ●	–	–	–	–	–	–	–	–	–
Char 1 Bit 8	8	0	1	1	1	1	0	0	1
Char 2 Bit 1	9	1	0	1	1	1	1	0	0
Char 2 Bit 2	10	0	1	0	1	1	1	1	0
● ●	–	–	–	–	–	–	–	–	–
Char 2 Bit 8	16	1	1	1	0	0	0	0	1
Char 3 Bit 1	17	0	1	1	1	0	0	0	0
Char 3 Bit 2	18	1	0	1	1	1	0	0	0
● ●	–	–	–	–	–	–	–	–	–
Char 3 Bit 8	24	1	1	1	0	0	1	1	0
Char 4 Bit 1	25	0	1	1	1	0	0	1	1
Char 4 Bit 2	26	1	0	1	1	1	0	0	1
● ●	–	–	–	–	–	–	–	–	–
Char 4 Bit 8	32	1	0	0	1	0	1	1	0

←BCC Accumulation

BIT POSITION

	P	6	5	4	3	2	1	0
Char 1	0	1	1	1	1	0	0	1
Char 2	1	0	0	1	1	0	0	0
Char 3	0	0	0	0	0	1	1	1
Char 4	0	1	1	1	0	0	0	0

LSB

Figure A-1 BCC Computation for Four 8-Bit Characters Using LRC-8

There are several CRC codes; selection is based on the code's error detection pattern and the ease of hardware implementation. The KG11-A implements three CRC codes: CRC-12, CRC-16, and CRC-CCITT.

A.3.2 Mathematical Background

A cyclic code message consists of a specific number of data bits and a BCC that is computed by the KG11-A. Let n equal the total number of bits in the message and k equal the number of data bits; then n-k equals the number of bits in the BCC.

The code message is derived from two polynomials which are algebraic representations of two binary words, the generator polynomial P(X) and the message polynomial G(X). The generator polynomial is the type of code used (CRC-12, CRC-16, and CRC-CCITT); the message polynomial is the string of serial data bits. The polynomials are usually represented algebraically by a string of terms in powers of X such as $X^n \ldots + X^3 + X^2 + X + X^0$ (or 1). In binary form, a 1 is placed in each position that contains a term; absence of a term is indicated by a 0. The convention followed in this manual is to place the least significant bit (X^0) at the right. For example, if a polynomial is given as $X^4 + X + 1$ its binary representation is 10011 (3rd and 2nd degree terms are not present).

Given a message polynomial G(X) and a generator polynomial P(X), the objective is to construct a code message polynomial F(X) that is evenly divisible by P(X). It is accomplished as follows:

a. Multiply the message G(X) by X^{n-k} where n-k is the number of bits in the BCC.

b. The resulting product X^{n-k} [G(X)] is divided by the generator polynomial P(X).

c. The quotient is disregarded and the remainder C(X) is added to the product to yield the code message polynomial F(X), which is represented as X^{n-k} [G(X)] + C(X).

The division is performed in binary without carries or borrows. In this case, the remainder is always one bit less than the divisor. The remainder is the BCC and the divisor is the generator polynomial; therefore, the bit length of the BCC is always one less than the number of bits in the generator polynomial.

A simple example is explained below.

1. Given:
   Message polynomial G(X) = 110011 ($X^5 + X^4 + X + X^0$)
   Generator polynomial P(X) = 11001 ($X^4 + X^3 + 1$)
   G(X) contains 6 data bits
   P(X) contains 5 bits and will yield a BCC with 4 bits; therefore, n-k = 4.

2. Multiplying the message G(X) by X^{n-k} gives:

   $$X^{n-k}\ [G(X)] = X^4\ (X^5 + X^4 + X + X^0) = X^9 + X^8 + X^5 + X^4$$

   The binary equivalent of this product contains 10 bits and is 1100110000.

3. This product is divided by P(X)

```
                 100001 ←—— quotient
P(X)→11001 | 1100110000 ←—— X^(n-k) [G(X)]
             11001
                 10000
                 11001
                  1001 ←—— remainder = C(X) = BCC
```

4. The remainder C(X) is added to X^{n-k} [G(X)] to give F(X) = 1100111001.

The code message polynomial is transmitted. The receiving station divides it by the same generator polynomial. If there is no error, the division will produce no remainder and it is assumed that the message is correct. A remainder indicates an error. The division is shown below.

```
                 100001
P(X)→11001 | 1100111001 ←—— F(X)
             11001
                 11001
                 11001
                 00000 ←—— no remainder
```

A more practical example of generating a BCC by long division is shown in Figure A-2.

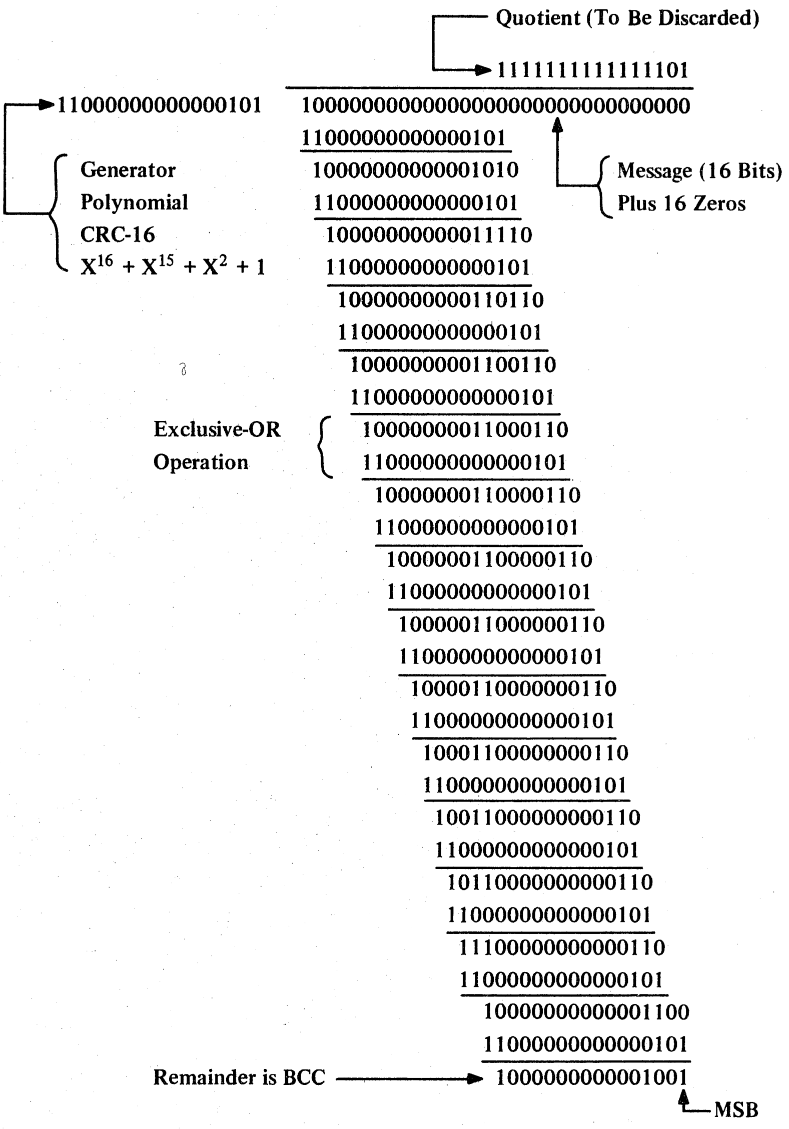


Figure A-2 BCC Computation Using Long Division Method
With CRC-16 as Generator

A.3.3 Hardware Implementation of CRC

The BCC is computed and accumulated in a shift register. The configuration of the register is based on the CRC code to be implemented. The number of stages in the register is equal to the degree of the generating polynomial; the number of exclusive-OR elements is a function of the polynomial also. In the subsequent examples, a unique register configuration is shown for each CRC code (CRC-12, CRC-16, and CRC-CCITT). In the KG11-A, only one BCC register is used. Logic is used to configure the register for the selected CRC code.

1. **CRC-12**

CRC-12 is applied to synchronous systems that use 6-bit characters. The BCC accumulation is 12 bits. The generator polynomial is $X^{12} + X^{11} + X^3 + X^2 + X + 1$ with prime factors of (X+1) and $(X^{11} + X^2 + 1)$. It provides error detection of bursts up to 12 bits in length. Additionally, 99.955% of error bursts greater than 12 bits can be detected.

Figure A-3 shows a block diagram configuration of a BCC register for use with CRC-12. A step-by-step shift pattern is shown as the data is serially applied to the register. Initially, the register contains all 0s. A 12-bit data word (a 1 followed by eleven 0s is the input to the register). Prior to the first shift, the first data bit (LSB), which is a 1, is exclusive-ORed with the 0 from the LSB of the register. The result on the serial quotient line is a 1 which is sent via the feedback paths to the following places: bit 11 and the exclusive-ORs between bits 11 and 10, bits 10 and 9, bits 9 and 8, and bits 1 and 0. When the feedback settles down, the first shift takes place and produces the register states in the line labeled 1 under the SHIFT NO. column. The shift also presents the next data bit to the input. The process repeats until all 12 data bits are encoded. The BCC is the contents of the register after shift number 12. The most important fact to remember is that the exclusive-OR of the LSB of the BCC register and the input data bit set up the feedback path prior to the shifting operation. The result of this operation is shown under the column labeled FEEDBACK BEFORE SHIFT. When the shift takes place, the results of the exclusive-OR operations are shifted into the register.

The subsequent examples, which show the BCC accumulation for CRC-CCITT and CRC-16, can be analyzed the same way; the register length, data word length, and feedback paths are different but the process is the same.

2. **CRC-CCITT**

CRC-CCITT is the standard used to compute a BCC for European systems. When operating with 8-bit characters, the BCC accumulation is 16 bits. The generator polynomial is $X^{16} + X^{12} + X^5 + 1$. It provides error detection of bursts up to 16 bits in length.

Figure A-4 shows a BCC accumulation using a 16-bit data word (a 1 followed by fifteen 0s).

3. **CRC-16**

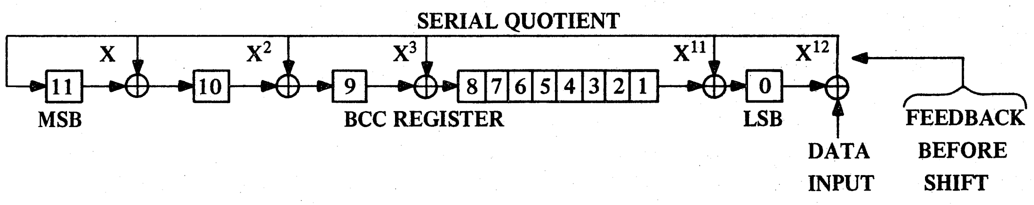
CRC-16 is applied to synchronous systems that use 8-bit characters. The BCC accumulation is 16 bits. The generator polynomial is $X^{16} + X^{15} + X^2 + 1$ with prime factors of (X+1) and $(X^{15} + X + 1)$. It provides error detection of bursts up to 16 bits in length. Additionally, 99.955% of error bursts greater than 16 bits can be detected.

Figure A-5 shows a BCC accumulation using a 16-bit data word (a 1 followed by fifteen 0s).

The three CRC examples (Figures A-3, A-4, and A-5) show BCC accumulations that are to be used in a transmission sequence. Figure A-6 shows the BCC accumulation performed on a message that has been received along with a BCC. The message and the computed BCC have been taken from Figure A-5; i.e., a BCC accumulation using CRC-16. In Figure A-6, the accumulation process is the same as that shown in Figure A-5 through shift number 16. Starting with shift number 17, the BCC is sent to the data input, LSB first. A correct transmitted BCC results in all 0s in the BCC register at the end of the computation. In effect, this is a comparison of the transmitted BCC with the one computed at the receiving station. A correct comparison yields a 0 remainder, or all 0s in the BCC register.

A.4 HARDWARE CONFIGURATION OF BCC REGISTER

The KG11-A uses one register to accumulate the BCC for two LRC modes and three CRC modes. It is a 16-bit register that uses a logic network to provide the proper feedback path and exclusive-OR operators for the LRC or CRC mode selected. Figure A-7 is a block diagram of the BCC register that shows the configuration for the two LRC modes and three CRC modes.



SHIFT NO.							
Start	0	0	0	0 0 0 0 0 0 0 0	0	–	1
1	1	1	1	1 0 0 0 0 0 0 0	1	1 LSB	1
2	1	0	0	0 1 0 0 0 0 0 0	1	0	1
3	1	0	1	1 0 1 0 0 0 0 0	1	0	1
4	1	0	1	0 1 0 1 0 0 0 0	1	0	1
5	1	0	1	0 0 1 0 1 0 0 0	1	0	1
6	1	0	1	0 0 0 1 0 1 0 0	1	0	1
7	1	0	1	0 0 0 0 1 0 1 0	1	0	1
8	1	0	1	0 0 0 0 0 1 0 1	1	0	1
9	1	0	1	0 0 0 0 0 0 1 0	0	0	0
10	0	1	0	1 0 0 0 0 0 0 1	0	0	0
11	0	0	1	0 1 0 0 0 0 0 0	1	0	1
12	1	1	1	0 0 1 0 0 0 0 0	1	0 MSB	1

Arrows Indicate Exclusive-OR of Data Bit and LSB of BCC Register Prior to Shift

Block Check Character (BCC)

12 Bit Data Word

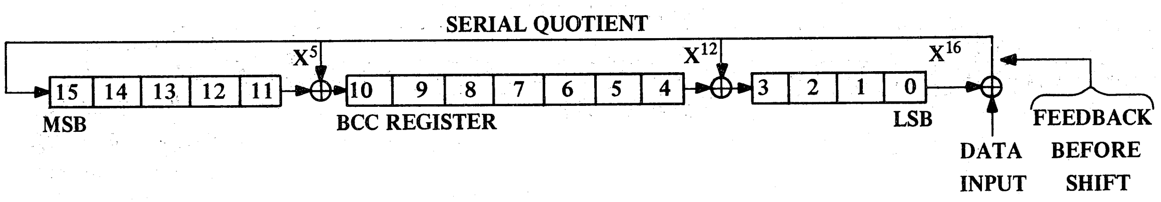
NOTES

□ = BCC Register Stage

⊕ = Exclusive-OR

CRC-12 Polynomial = $X^{12} + X^{11} + X^3 + X^2 + X + 1$

Figure A-3 BCC Accumulation Using CRC-12, Transmit Sequence



SHIFT NO.						
Start	0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0	–		1
1	1 0 0 0 0	1 0 0 0 0 0 0	1 0 0 0	1	LSB	0
2	0 1 0 0 0	0 1 0 0 0 0 0	0 1 0 0	0		0
3	0 0 1 0 0	0 0 1 0 0 0 0	0 0 1 0	0		0
4	0 0 0 1 0	0 0 0 1 0 0 0	0 0 0 1	0		1
5	1 0 0 0 1	1 0 0 0 1 0 0	1 0 0 0	0		0
6	0 1 0 0 0	1 1 0 0 0 1 0	0 1 0 0	0		0
7	0 0 1 0 0	0 1 1 0 0 0 1	0 0 1 0	0		0
8	0 0 0 1 0	0 0 1 1 0 0 0	1 0 0 1	0		1
9	1 0 0 0 1	1 0 0 1 1 0 0	1 1 0 0	0		0
10	0 1 0 0 0	1 1 0 0 1 1 0	0 1 1 0	0		0
11	0 0 1 0 0	0 1 1 0 0 1 1	0 0 1 1	0		1
12	1 0 0 1 0	1 0 1 1 0 0 1	0 0 0 1	0		1
13	1 1 0 0 1	1 1 0 1 1 0 0	0 0 0 0	0		0
14	0 1 1 0 0	1 1 1 0 1 1 0	0 0 0 0	0		0
15	0 0 1 1 0	0 1 1 1 0 1 1	0 0 0 0	0		0
16	0 0 0 1 1	0 0 1 1 1 0 1	1 0 0 0	0	MSB	0

Block Check Character (BCC)

16 Bit Data Word

Arrows Indicate Exclusive-OR of Data Bit and LSB of BCC Register Prior to Shift

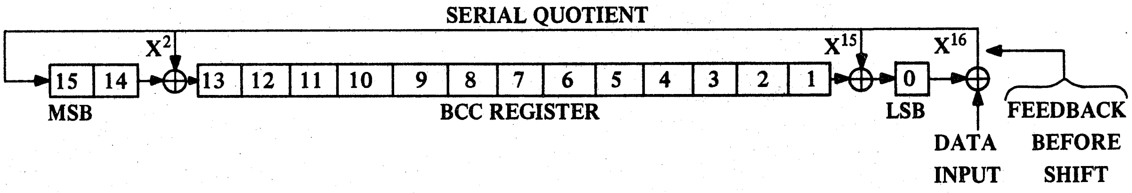
NOTES

□ = BCC Register Stage

⊕ = Exclusive-OR

CRC–CCITT Polynomial = $X^{16} + X^{12} + X^5 + 1$

Figure A-4 BCC Accumulation Using CRC-CCITT, Transmit Sequence



SHIFT NO.					
Start	0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0	0	–	1
1	1 0	1 0 0 0 0 0 0 0 0 0 0 0 0 0	1	1	1
2	1 1	1 1 0 0 0 0 0 0 0 0 0 0 0 0	1	0	1
3	1 1	0 1 1 0 0 0 0 0 0 0 0 0 0 0	1	0	1
4	1 1	0 0 1 1 0 0 0 0 0 0 0 0 0 0	1	0	1
5	1 1	0 0 0 1 1 0 0 0 0 0 0 0 0 0	1	0	1
6	1 1	0 0 0 0 1 1 0 0 0 0 0 0 0 0	1	0	1
7	1 1	0 0 0 0 0 1 1 0 0 0 0 0 0 0	1	0	1
8	1 1	0 0 0 0 0 0 1 1 0 0 0 0 0 0	1	0	1
9	1 1	0 0 0 0 0 0 0 1 1 0 0 0 0 0	1	0	1
10	1 1	0 0 0 0 0 0 0 0 1 1 0 0 0 0	1	0	1
11	1 1	0 0 0 0 0 0 0 0 0 1 1 0 0 0	1	0	1
12	1 1	0 0 0 0 0 0 0 0 0 0 1 1 0 0	1	0	1
13	1 1	0 0 0 0 0 0 0 0 0 0 0 1 1 0	1	0	1
14	1 1	0 0 0 0 0 0 0 0 0 0 0 0 0 1	0	0	0
15	0 1	1 0 0 0 0 0 0 0 0 0 0 0 0 0	1	0	1
16	1 0	0 1 0 0 0 0 0 0 0 0 0 0 0 0	1	0	1

LSB

MSB

Arrows Indicate Exclusive-OR of Data Bit and LSB of BCC Register Prior to Shift

Block Check Character (BCC)

16 Bit Data Word

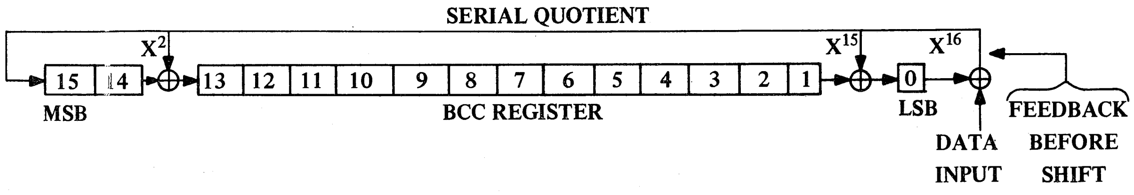
NOTES

□ = BCC Register Stage

⊕ = Exclusive-OR

CRC-16 Polynomial = $X^{16} + X^{15} + X^2 + 1$

Figure A-5 BCC Accumulation Using CRC-16, Transmit Sequence



SHIFT NO.						
Start	0 0	0 0 0 0 0 0 0 0 0 0 0 0 0	0	–		1
1	1 0	1 0 0 0 0 0 0 0 0 0 0 0 0	1	1	LSB	1
2	1 1	1 1 0 0 0 0 0 0 0 0 0 0 0	1	0		1
3	1 1	0 1 1 0 0 0 0 0 0 0 0 0 0	1	0		1
4	1 1	0 0 1 1 0 0 0 0 0 0 0 0 0	1	0		1
●	● ●	● ●	●	●		●
●	● ●	● Same as CRC-16 Transmit ●	●	●		●
●	● ●	● Sequence (See Figure A-5) ●	●	●	DATA	●
●	● ●	● ●	●	●		●
13	1 1	0 0 0 0 0 0 0 0 0 0 0 1 1	1	0		1
14	1 1	0 0 0 0 0 0 0 0 0 0 0 0 1	0	0		0
15	0 1	1 0 0 0 0 0 0 0 0 0 0 0 0	1	0		1
16	1 0	0 1 0 0 0 0 0 0 0 0 0 0 0	1	0	MSB	1
17	0 1	0 0 1 0 0 0 0 0 0 0 0 0 0	0	1	LSB	0
18	0 0	1 0 0 1 0 0 0 0 0 0 0 0 0	0	0		0
19	0 0	0 1 0 0 1 0 0 0 0 0 0 0 0	0	0		0
20	0 0	0 0 1 0 0 1 0 0 0 0 0 0 0	0	0		0
21	0 0	0 0 0 1 0 0 1 0 0 0 0 0 0	0	0		0
22	0 0	0 0 0 0 1 0 0 1 0 0 0 0 0	0	0		0
23	0 0	0 0 0 0 0 1 0 0 1 0 0 0 0	0	0		0
24	0 0	0 0 0 0 0 0 1 0 0 1 0 0 0	0	0	BCC	0
25	0 0	0 0 0 0 0 0 0 1 0 0 1 0 0	0	0		0
26	0 0	0 0 0 0 0 0 0 0 1 0 0 1 0	0	0		0
27	0 0	0 0 0 0 0 0 0 0 0 1 0 0 1	0	0		0
28	0 0	0 0 0 0 0 0 0 0 0 0 1 0 0	1	0		0
29	0 0	0 0 0 0 0 0 0 0 0 0 0 1 0	0	1		0
30	0 0	0 0 0 0 0 0 0 0 0 0 0 0 1	0	0		0
31	0 0	0 0 0 0 0 0 0 0 0 0 0 0 0	1	0		0
32	0 0	0 0 0 0 0 0 0 0 0 0 0 0 0	0	1	MSB	0

Arrows Indicate Exclusive-OR of Data Bit and LSB of BCC Register Prior to Shift

BCC Register ALL 0s Received Message Assumed To Be Correct

16 Bit Data Word and BCC Received

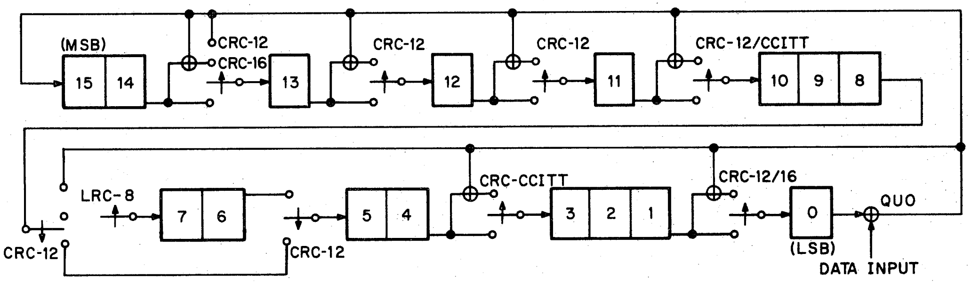
NOTES

□ = BCC Register Stage

⊕ = Exclusive-OR

CRC-16 Polynomial = $X^{16} + X^{15} + X^2 + 1$

Figure A-6 BCC Accumulation Using CRC-16, Receive Sequence



NOTES:

1. Each BCC Register stage (Flip Flop) shown as a numbered box.
2. ⊕ = Exclusive OR.
3. Data Flow is Left to Right.

11-0923

Figure A-7 BCC Register, Block Diagram

APPENDIX B
ABBREVIATIONS

This appendix contains two abbreviation lists. List 1 contains abbreviations that are used throughout the manual. List 2 contains the USASCII control characters used in the message examples in Chapter 4.

List 1 – Abbreviations

BCC	block check character
BG	bus grant
CCITT	Consultative Committee for International Telephone and Telegraph
CLK	clock
CLR	clear
CRC	cyclic redundancy check
CSR	control and status register
DATI	data in
DATO	data out
DATOB	data out, byte
DDB	double data byte
H	high
INIT	initialize
L	low
LRC	longitudinal redundancy check
LSB	least-significant bit
MSB	most-significant bit
QUO	quotient
REG	register
SEL	select
SEN	shift-enable
USASCII	USA Standard Code for Information Interchange
VRC	vertical redundancy check
X-OR	exclusive-OR

List 2 – USASCII Control Characters

DLE	Data Link Escape
ETB	End of Transmission Block
ETX	End of Text
ITB	Intermediate Block Check Character
STX	Start of Text

APPENDIX C
INTEGRATED CIRCUIT DESCRIPTIONS

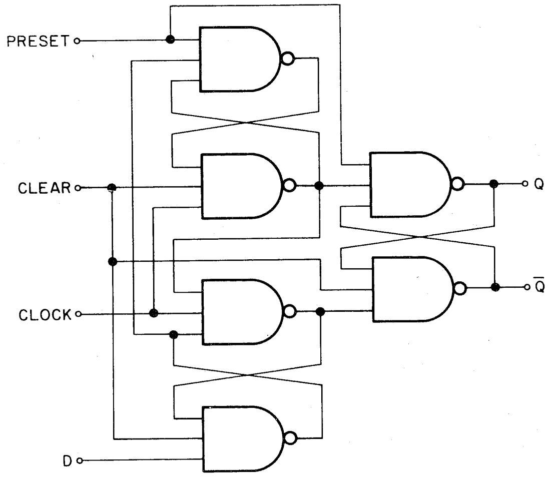
This appendix provides a description, pin designation diagram, logic diagram, and truth table for the more complex integrated circuits (ICs) used in the KG11-A. They are listed below.

7474 Dual D-Type Edge-Triggered Flip-Flops
8242 4-Bit Digital Comparator
8266 2-Input 4-Bit Digital Multiplexer
8271 4-Bit Shift Register
74197 Presettable Binary Counter/Latch

All but the 7474 are categorized as medium scale integration (MSI) devices.

7474 DUAL D-TYPE EDGE-TRIGGERED FLIP-FLOPS

The 7474 D-type flip-flops are triggered by the positive edge of the clock pulse. They feature direct-clear and preset inputs and complementary Q and $\overline{Q}$ outputs.

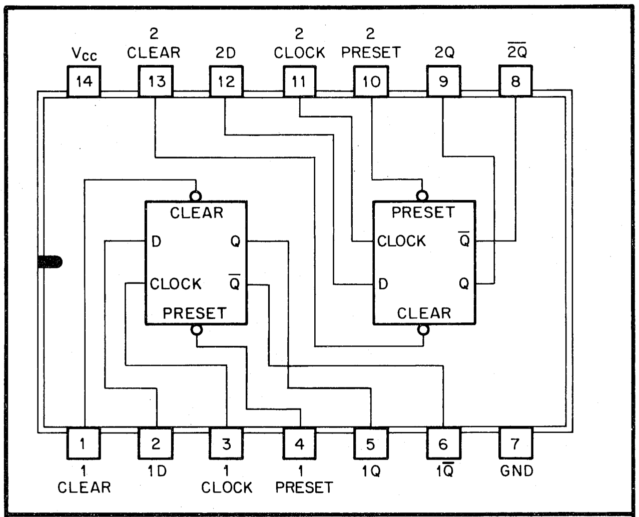


11-0767

TRUTH TABLE (Each Flip-Flop)

t_n	t_{n+1}	
Input D	**Output Q**	**Output $\overline{Q}$**
0	0	1
1	1	0

Notes: 1. t_n = bit time before clock pulse.
2. t_{n+1} = bit time after clock pulse.

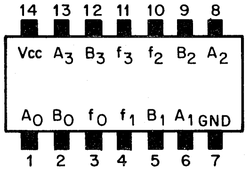


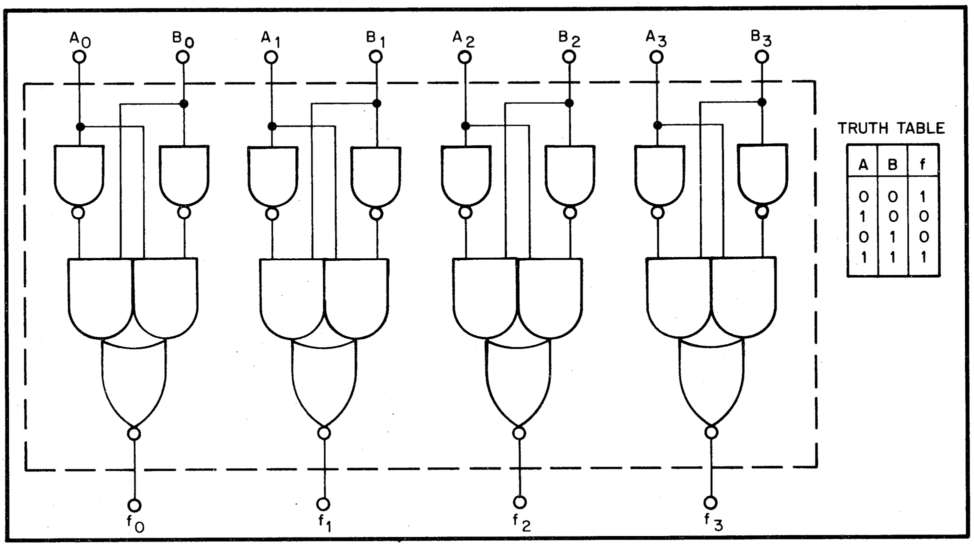
POSITIVE LOGIC: LOW INPUT TO PRESET SETS Q TO LOGICAL 1
LOW INPUT TO CLEAR SETS Q TO LOGICAL 0
PRESET AND CLEAR ARE INDEPENDENT OF CLOCK

11-0766

8242 4-BIT DIGITAL COMPARATOR (QUAD EXCLUSIVE-NOR)

The 8242 contains four independent exclusive-NOR gates which may be used to implement digital comparison functions. The 8242 outputs are bare collector to facilitate implementation of multiple-bit comparisons; a 4-bit comparison is made by connecting the outputs of the four independent gates together. This common output connection is high only when all input pairs are matched (both 0s or both 1s).





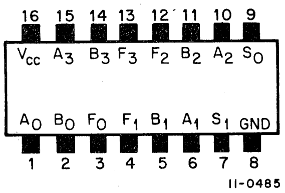
TRUTH TABLE

A	B	f
0	0	1
1	0	0
0	1	0
1	1	1

11-0472

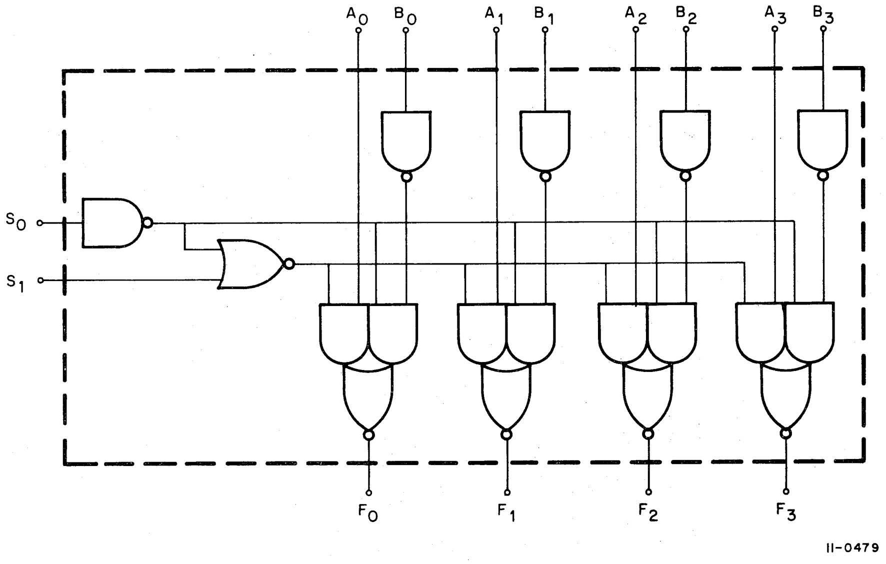
8266 2-INPUT 4-BIT DIGITAL MULTIPLEXER

The 8266 multiplexer is able to choose from two different input sources, each containing 4 bits: $A = (A_0, A_1, A_2, A_3)$, $B = (B_0, B_1, B_2, B_3)$. The selection is controlled by input S_0, while the second control input, S_1, is held at 0.


16 15 14 13 12 11 10 9
V_{cc} A_3 B_3 F_3 F_2 B_2 A_2 S_0
A_0 B_0 F_0 F_1 B_1 A_1 S_1 GND
1 2 3 4 5 6 7 8
11-0485

TRUTH TABLE

Select Lines S_0	 S_1	Output f_n (0,1,2,3)
0	0	B_n
0	1	B_n
1	0	$\overline{A}_n$
1	1	1



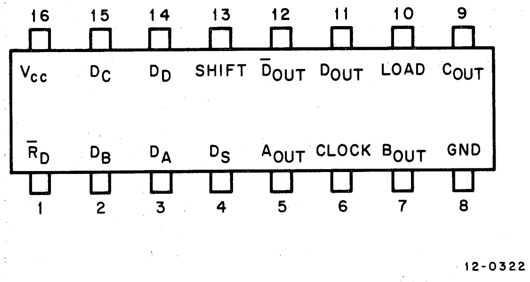
8271 4-BIT SHIFT REGISTER

The 8271 is a 4-bit shift register with both serial and parallel data entry capability. The data input lines are single-ended true input data lines which condition their specific register bit location after an enabled clocking transition. Since data transfer is synchronous with the clock, data maybe transferred in any serial/parallel input/output relationship.

The internal design uses stored charge binaries to effect a desirable speed/power relationship. These binaries respond to the negative-going clock transition. A buffer clock driver has been included to minimize input clock loading and reduce the input clock transition requirements.

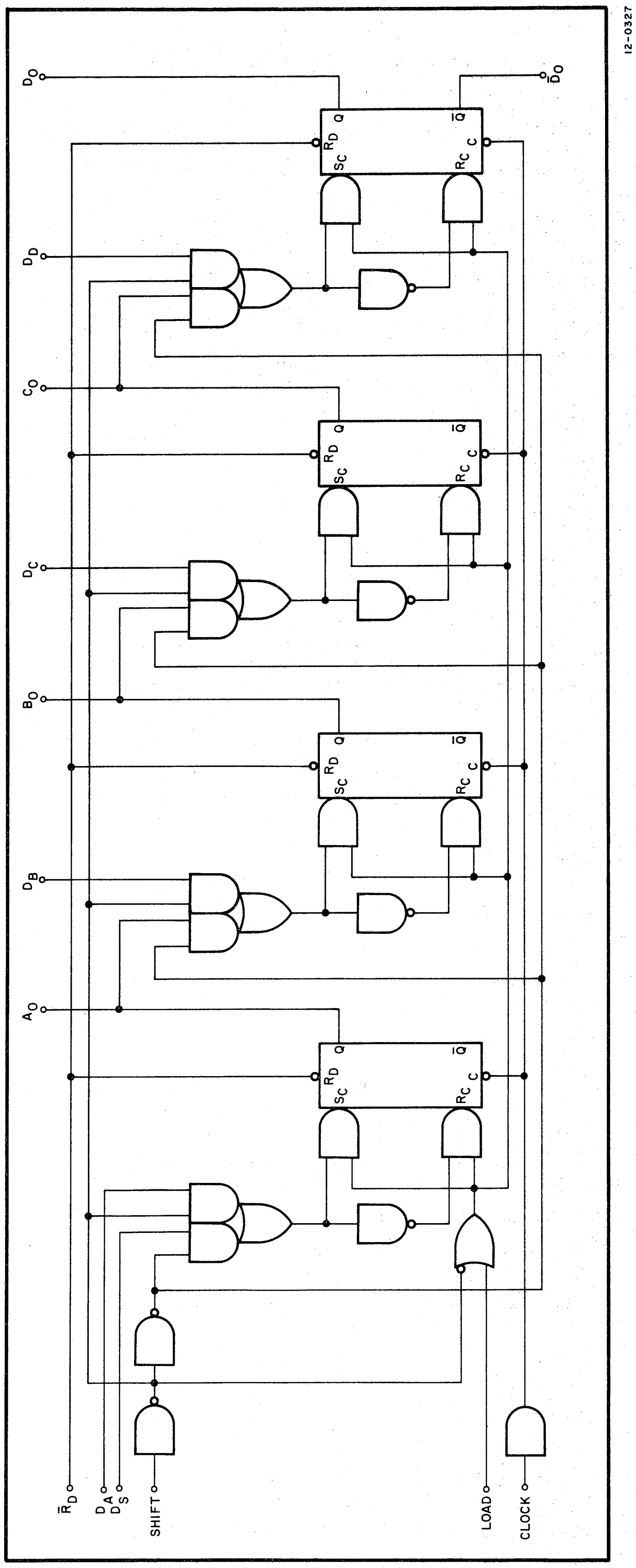
Mode control logic is available to determine three possible control states: serial shift right mode, parallel enter mode, and no change or hold mode. These states accomplish logical decoding for system control. The truth table for the control modes is shown below. For applications not requiring the hold mode, the load input may be tied high and the shift input used as the mode control.

The 8271 provides a direct reset $(\overline{R}_D)$, and a $\overline{D}_{out}$ line.



TRUTH TABLE

Control State	Load	Shift
Hold	0	0
Parallel Entry	1	0
Shift Right	0	1
Shift Right	1	1


12-0327
D_O
$\bar{D}_O$
Q
$\bar{Q}$
R_D
S_C
R_C
C
D_D
C_O
Q
$\bar{Q}$
R_D
S_C
R_C
C
D_C
B_O
Q
$\bar{Q}$
R_D
S_C
R_C
C
D_B
A_O
Q
$\bar{Q}$
R_D
S_C
R_C
C
$\bar{R}_D$
D_A
D_S
SHIFT
LOAD
CLOCK

74197 PRESETTABLE BINARY COUNTER/LATCH

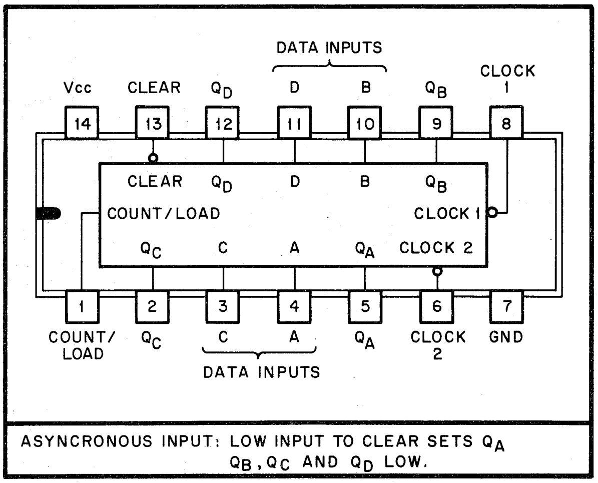
This high-speed monolithic counter consists of four dc-coupled master-slave flip-flops which are internally interconnected to provide a divide-by-2 counter and a divide-by-8 counter. This counter is fully programmable; i.e., the outputs may be preset to any state by placing a low on the count/load input and entering the desired data at the data inputs. The outputs change to agree with the data inputs independent of the state of the clocks.

The counter may also be used as a 4-bit latch by using the count/load input as the strobe and entering data at the data inputs. The outputs directly follow the data inputs when the count/load input is low, but remain unchanged when the count/load input is high and the clock inputs are inactive.

As a high-speed counter, the 74197 accepts count frequencies of 0 to 50 MHz at the clock-1 input and 0 to 25 MHz at the clock-2 input. During the count operation, transfer of information to the outputs occurs on the negative-going edge of the clock pulse. This counter features a direct-clear which, when taken low, sets all outputs low regardless of the states of the clocks.

The output of flip-flop A is not internally connected to the succeeding flip-flops; therefore, the counter may be operated in two independent modes:

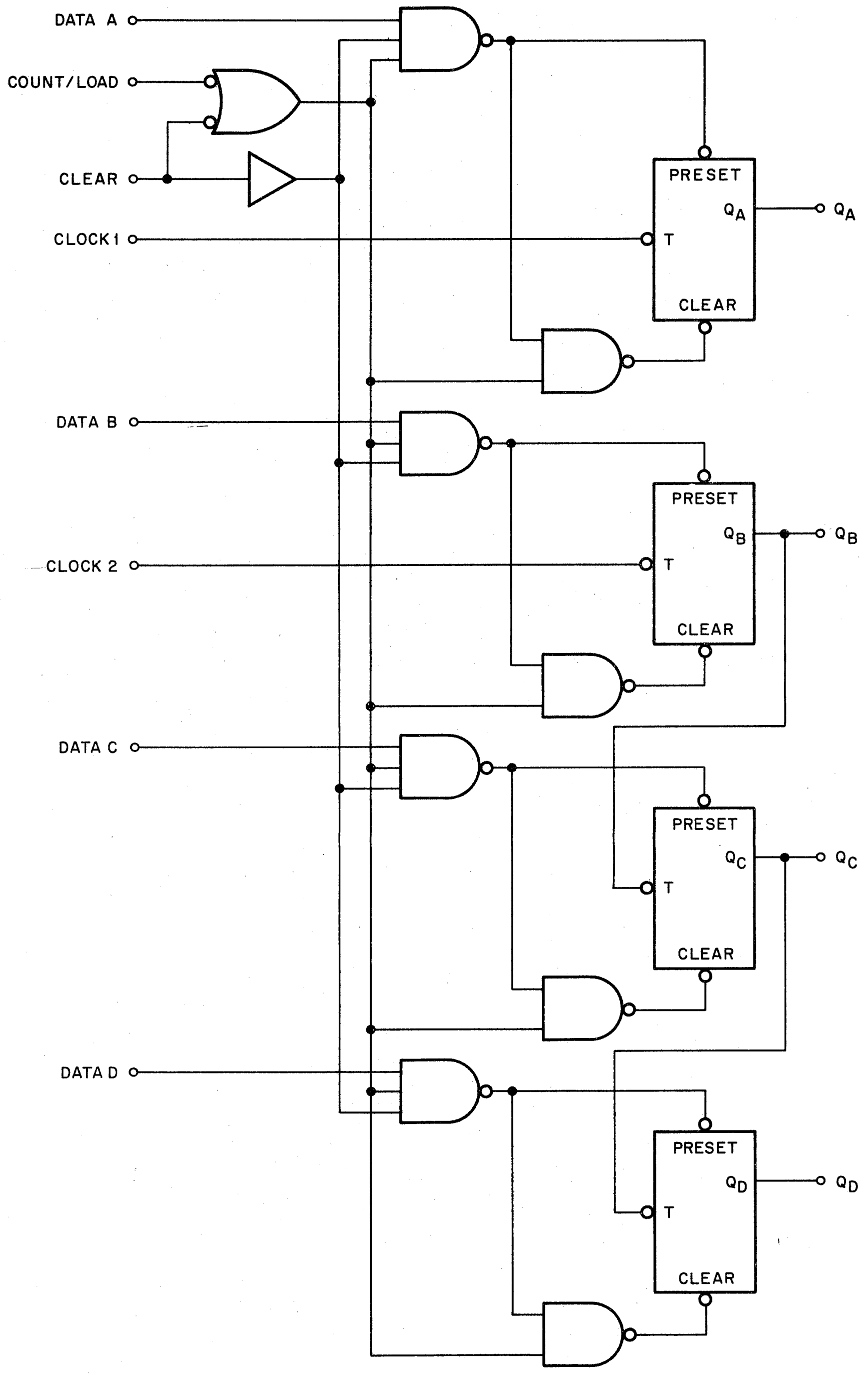
1. When used as a high-speed 4-bit ripple-through counter, output Q_A must be externally connected to the clock-2 input. The input count pulses are applied to the clock-1 input. Simultaneous divisions by 2, 4, 8, and 16 are performed at the Q_A, Q_B, and Q_D outputs.
2. When used as a 3-bit ripple-through counter, the input count pulses are applied to the clock-2 input. Simultaneous frequency divisions by 2, 4, and 8 are available at the Q_B, Q_C, and Q_D outputs. Independent use of flip-flop A is available if the load and clear functions coincide with those of the 3-bit ripple-through counter.



ASYNCRONOUS INPUT: LOW INPUT TO CLEAR SETS Q_A Q_B, Q_C AND Q_D LOW.

*Pin assignments for these circuits are the same for all packages.

11-0482


DATA A
COUNT/LOAD
CLEAR
CLOCK 1
PRESET
Q_A
T
CLEAR
Q_A
DATA B
PRESET
Q_B
CLOCK 2
T
CLEAR
Q_B
DATA C
PRESET
Q_C
T
CLEAR
Q_C
DATA D
PRESET
Q_D
T
CLEAR
Q_D
11-0481

READER'S COMMENTS

KG11-A EXCLUSIVE-OR and
CRC BLOCK CHECK MANUAL
DEC-11-HKGAB-A-D

Your comments and suggestions will help us in our continuous effort to improve the quality and usefulness of our publications.

What is your general reaction to this manual? In your judgment is it complete, accurate, well organized, well written, etc.? Is it easy to use? ____________________

____

____

____

What features are most useful? ____________________

____

____

____

What faults do you find with the manual? ____________________

____

____

____

Does this manual satisfy the need you think it was intended to satisfy? ____________________

Does it satisfy *your* needs? ____________ Why? ____________________

____

____

____

Would you please indicate any factual errors you have found. ____________________

____

____

____

Please describe your position. ____________________

Name ____________________ Organization ____________________

Street ____________________ Department ____________________

City ____________ State ____________ Zip or Country ____________

Fold Here

Do Not Tear - Fold Here and Staple

digital

Digital Equipment Corporation
Technical Documentation Department
146 Main Street
Maynard, Massachusetts 01754

digital

DIGITAL EQUIPMENT CORPORATION
MAYNARD, MASSACHUSETTS 01754