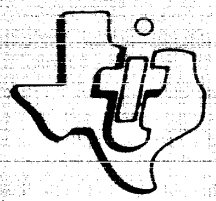
The Engineering Staff of
TEXAS INSTRUMENTS INCORPORATED
Semiconductor Group



TMS 9902 ASYNCHRONOUS COMMUNICATION CONTROLLER

JANUARY 1977

TEXAS INSTRUMENTS
INCORPORATED

PRELIMINARY DATA SHEET:
Supplementary data may be published at a later date.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

1. INTRODUCTION

The TMS 9902 Asynchronous Communication Controller (ACC) is a peripheral device for the TMS 9900 family of microprocessors. The ACC provides an interface between the microprocessor and a serial asynchronous communication channel, performing the timing and data serialization and deserialization, thus facilitating the control of the asynchronous channel by the microprocessor. Key features of the TMS 9902 ACC are as follows:

- 5- to 8-bit character length
- 1, 1 1/2, or 2 stop bits
- Even, odd, or no parity
- Fully programmable data rate generation
- Interval timer with resolution from 64 to 16,320 μs
- Fully TTL compatible, including single power supply.

2. DEVICE INTERFACE

The relationship of the ACC to other components in the system is shown in Figures 1 and 2. The ACC is connected to the asynchronous channel through level shifters which translate the TTL inputs and outputs to the appropriate levels (e.g., RS-232C, TTY current loop, etc.). The microprocessor transfers data to and from the ACC via the Communication Register Unit (CRU).

2.1 CPU INTERFACE

The ACC interfaces to the CPU through the Communication Register Unit (CRU). The CRU interface consists of five address-select lines (S0–S4), chip enable ($\overline{CE}$), and three CRU control lines (CRUIN, CRUOUT, and CRUCLK). When $\overline{CE}$ becomes active (low), the five select lines address the CRU bit being accessed. When data is being transferred to the ACC from the CPU, CRUOUT contains the valid datum which is strobed by CRUCLK. When ACC data is being read, CRUIN is the datum output by the ACC.

2.2 ASYNCHRONOUS COMMUNICATION CHANNEL INTERFACE

The interface to the asynchronous communication channel consists of an output control line ($\overline{RTS}$), two input status lines ($\overline{DSR}$ and $\overline{CTS}$), and serial transmit (XOUT) and receive (RIN) data lines. The request-to-send line ($\overline{RTS}$) is active (low) whenever the transmitter is activated. However, before data transmission begins, the clear-to-send ($\overline{CTS}$) input must be active. The data set ready ($\overline{DSR}$) input does not affect the receiver or transmitter. When $\overline{DSR}$ or $\overline{CTS}$ changes level, an interrupt is generated.

2.3 INTERRUPT OUTPUT

The interrupt output ($\overline{INT}$) is active (low) when any of the following conditions occur and the corresponding interrupt has been enabled by the CPU:

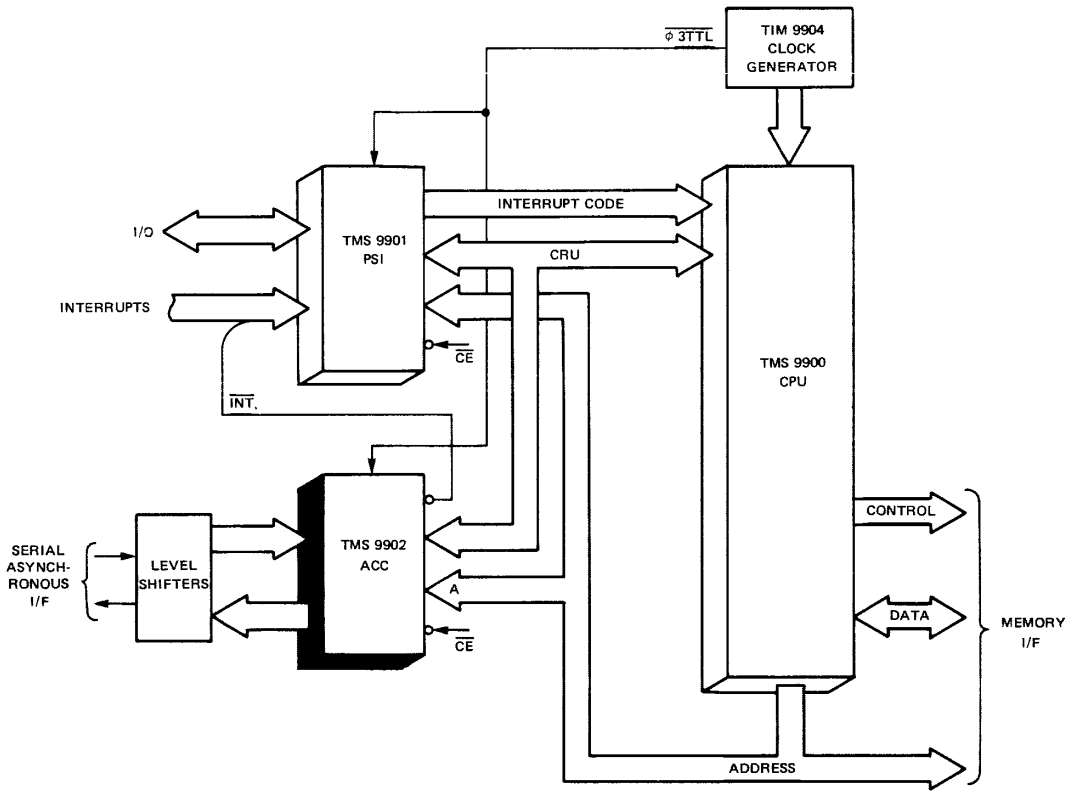


FIGURE 1 – TMS 9902 ACC IN A TMS 9900 SYSTEM

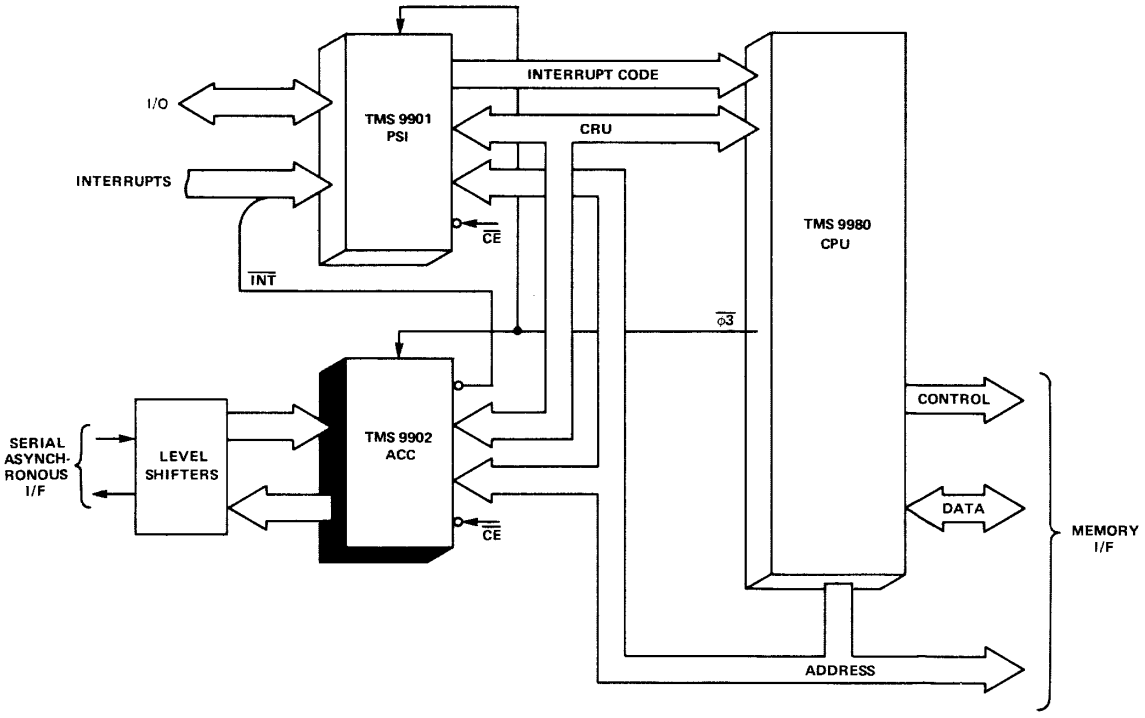


FIGURE 2 – TMS 9902 ACC IN A TMS 9980 SYSTEM

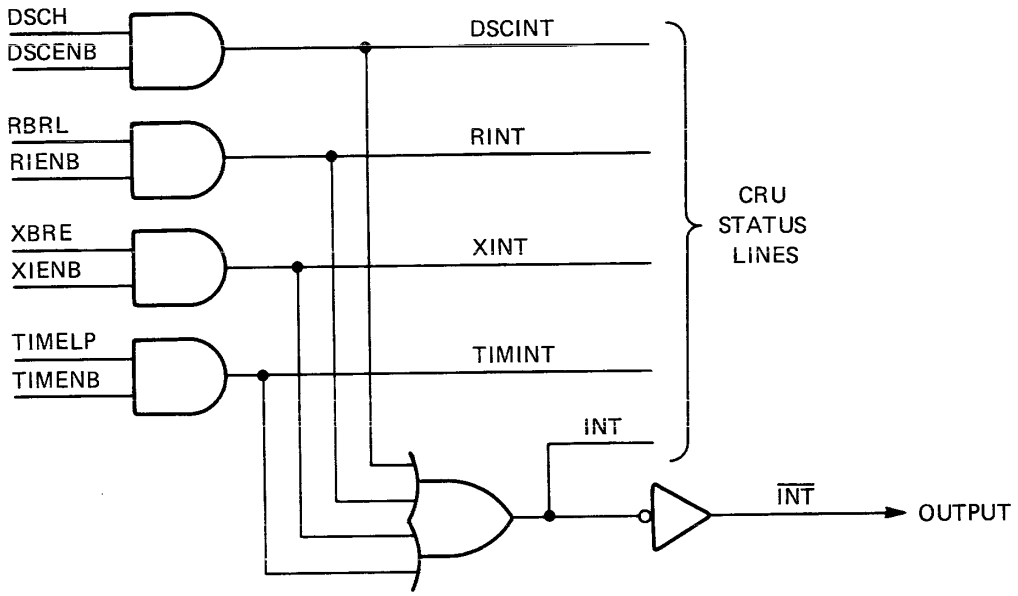
(1) $\overline{DSR}$ or $\overline{CTS}$ changes levels (DSCH = 1);

(2) a character has been received and stored in the Receive Buffer Register (RBRL = 1);

(3) the Transmit Buffer Register is empty (XBRE = 1); or

(4) the selected time interval has elapsed (TIMELP = 1).

The logical relationship of the interrupt output is shown below.



$\overline{INT}$ OUTPUT GENERATION

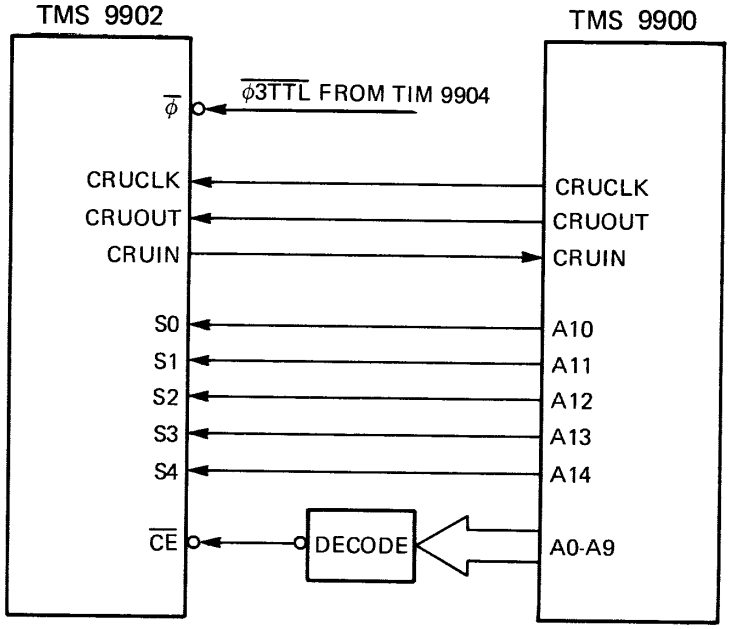
2.4 CLOCK INPUT

The clock input to the ACC ($\overline{\phi}$) is normally provided by the $\overline{\phi}3$ output of the TIM 9904 (9900 systems) or the TMS 9980 (9980 systems). This clock input is used to generate the internal device clock, which provides the time base for the transmitter, receiver, and interval timer of the ACC.

3. DEVICE OPERATION

3.1 CONTROL AND DATA OUTPUT

Data and control information is transferred to the ACC using $\overline{CE}$, S0–S4, CRUOUT, and CRUCLK. The diagrams below show the connection of the ACC to the TMS 9900 and TMS 9980 CPUs. The high-order CPU address lines are used to decode the $\overline{CE}$ signal when the device is being selected. The low-order address lines are connected to the five address-select lines (S0–S4). Table 1 describes the output bit address assignments for the ACC.


TMS 9902
TMS 9900
$\overline{\phi}$
$\overline{\phi 3TTL}$ FROM TIM 9904
CRUCLK
CRUOUT
CRUIN
S0
S1
S2
S3
S4
$\overline{CE}$
DECODE
CRUCLK
CRUOUT
CRUIN
A10
A11
A12
A13
A14
A0-A9

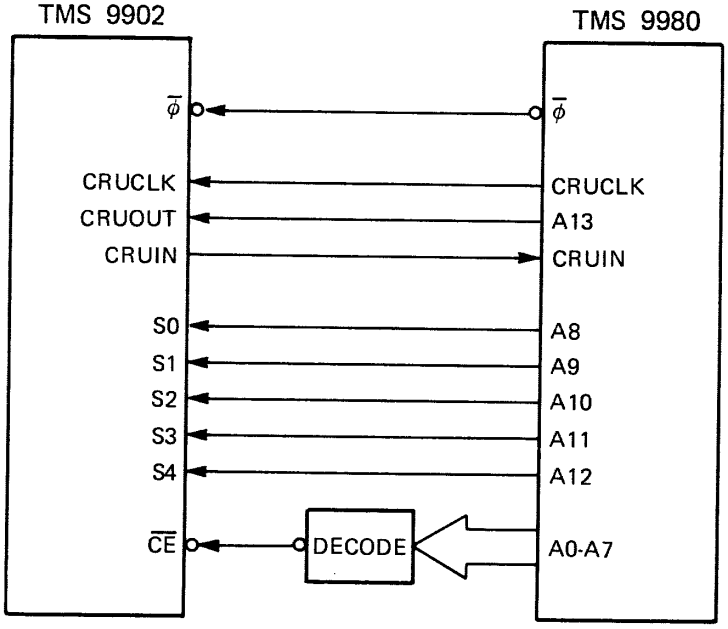

TMS 9902
TMS 9980
$\overline{\phi}$
CRUCLK
CRUOUT
CRUIN
S0
S1
S2
S3
S4
$\overline{CE}$
DECODE
$\overline{\phi}$
CRUCLK
A13
CRUIN
A8
A9
A10
A11
A12
A0-A7

TABLE 1

TMS 9902 ACC OUTPUT BIT ADDRESS ASSIGNMENTS

$ADDRESS_2$ S0 S1 S2 S3 S4	$ADDRESS_{10}$	NAME	DESCRIPTION
1 1 1 1 1	31	RESET	Reset device.
	30-22		Not used.
1 0 1 0 1	21	DSCENB	Data Set Status Change Interrupt Enable.
1 0 1 0 0	20	TIMENB	Timer Interrupt Enable
1 0 0 1 1	19	XBIENB	Transmitter Interrupt Enable
1 0 0 1 0	18	RIENB	Receiver Interrupt Enable
1 0 0 0 1	17	BRKON	Break On
1 0 0 0 0	16	RTSON	Request to Send On
0 1 1 1 1	15	TSTMD	Test Mode
0 1 1 1 0	14	LDCTRL	Load Control Register
0 1 1 0 1	13	LDIR	Load Interval Register
0 1 1 0 0	12	LRDR	Load Receiver Data Rate Register
0 1 0 1 1	11	LXDR	Load Transmit Data Rate Register
	10-0		Control, Interval, Receive Data Rate, Transmit Data Rate, and Transmit Buffer Registers

Bit 31 (RESET) – Writing a one or zero to Bit 31 causes the device to be reset, disabling all interrupts, initializing the transmitter and receiver, setting $\overline{RTS}$ inactive (high), setting all register load control flags (LDCTRL, LDIR, LRDR, and LXDR) to a logic one level, and resetting the BREAK flag. No other input or output operations should be performed for 11$\overline{\phi}$ clock cycles after issuing the RESET command.

Bit 30–Bit 22 – Not used.

Bit 21 (DSCENB) – Data Set Change Interrupt Enable. Writing a one to Bit 21 causes the $\overline{INT}$ output to be active (low) whenever DSCH (Data Set Status Change) is a logic one. Writing a zero to Bit 21 causes DSCH interrupts to be disabled. Writing either a one or zero to Bit 21 causes DSCH to be reset.

Bit 20 (TIMENB) – Timer Interrupt Enable. Writing a one to Bit 20 causes the $\overline{INT}$ output to be active whenever TIMELP (Timer Elapsed) is a logic one. Writing a zero to Bit 20 causes TIMELP interrupts to be disabled. Writing either a one or zero to Bit 20 causes TIMELP and TIMERR (Timer Error) to be reset.

Bit 19 (XBIENB) – Transmit Buffer Interrupt Enable. Writing a one to Bit 19 causes the $\overline{INT}$ output to be active whenever XBRE (Transmit Buffer Register Empty) is a logic one. Writing a zero to Bit 19 causes XBRE interrupts to be disabled. The state of XBRE is not affected by writing to Bit 19.

Bit 18 (RIENB) – Receiver Interrupt Enable. Writing a one to Bit 18 causes the $\overline{INT}$ output to be active whenever RBRL (Receive Buffer Register Loaded) is a logic one. Writing a zero to Bit 18 disables RBRL interrupts. Writing either a one or zero to Bit 18 causes RBRL to be reset.

Bit 17 (BRKON) – Break On. Writing a one to Bit 17 causes the XOUT (Transmitter Serial Data Output) to go to a logic zero whenever the transmitter is active and the Transmit Buffer Register (XBR) and the Transmit Shift Register (XSR) are empty. While BRKON is set, loading of characters into the XBR is inhibited. Writing a zero to Bit 17 causes BRKON to be reset and the transmitter to resume normal operation.

Bit 16 (RTSON) – Request-to-Send On. Writing a one to Bit 16 causes the $\overline{RTS}$ output to be active (low). Writing a zero to Bit 16 causes $\overline{RTS}$ to go to a logic one after the XSR and XBR are empty, and BRKON is reset. Thus, the $\overline{RTS}$ output does not become inactive (high) until after character transmission has been completed.

Bit 15 (TSTMD) – Test Mode. Writing a one to Bit 15 causes $\overline{RTS}$ to be internally connected to $\overline{CTS}$, XOUT to be internally connected to RIN, $\overline{DSR}$ to be internally held low, and the Interval Timer to operate at 32 times its normal rate. Writing a zero to Bit 15 re-enables normal device operation.

Bits 14–11 – Register Load Control Flags. Output Bits 14-11 control which of the five registers will be loaded by writing to Bits 10-0. The flags are prioritized as shown in Table 2.

TABLE 2
TMS 9902 ACC REGISTER LOAD SELECTION

REGISTER LOAD CONTROL FLAG STATUS				REGISTER ENABLED
LDCTRL	LDIR	LRDR	LXDR	
1	X	X	X	Control Register
0	1	X	X	Interval Register
0	0	1	X	Receive Data Rate Register
0	0	X	1	Transmit Data Rate Register
0	0	0	0	Transmit Buffer Register

Bit 14 (LDCTRL) – Load Control Register. Writing a one to Bit 14 causes LDCTRL to be set to a logic one. When LDCTRL = 1, any data written to bits 0-7 are directed to the Control Register. Note that LDCTRL is also set to a logic one when a one or zero is written to Bit 31 (RESET). Writing a zero to Bit 14 causes LDCTRL to be reset to a logic zero, disabling loading of the Control Register. LDCTRL is also automatically reset to a logic zero when a datum is written to Bit 7 of the Control Register which normally occurs as the last bit written when loading the Control Register with a LDCR instruction.

Bit 13 (LDIR) – Load Interval Register. Writing a one to Bit 13 causes LDIR to be set to a logic one. When LDIR = 1 and LDCTRL = 0, any data written to Bits 0-7 are directed to the Interval Register. Note that LDIR is also set to a logic one when a datum is written to Bit 31 (RESET); however, Interval Register loading is not enabled until LDCTRL is set to a logic zero. Writing a zero to Bit 13 causes LDIR to be reset to logic zero, disabling loading of the Internal Register. LDIR is also automatically reset to logic zero when a datum is written to Bit 7 of the Interval Register, which normally occurs as the last bit written when loading the Interval Register with a LDCR instruction.

Bit 12 (LRDR) – Load Receive Data Rate Register. Writing a one to Bit 12 causes LRDR to be set to a logic one. When LRDR = 1, LDIR = 0, and LDCTRL = 0, any data written to Bits 0-10 are directed to the Receive Data Rate Register. Note that LRDR is also set to a logic one when a datum is written to Bit 31 (RESET); however, Receive Data Rate Register loading is not enabled until LDCTRL and LDIR have been set to a logic zero. Writing a zero to Bit 12 causes LRDR to be reset to a logic zero, disabling loading of the Receive Data Rate Register. LRDR is also automatically reset to logic zero when a datum is written to Bit 10 of the Receive Data Rate Register, which normally occurs as the last bit written when loading the Receive Data Rate Register with a LDCR instruction.

Bit 11 (LXDR) – Load Transmit Data Rate Register. Writing a one to Bit 11 causes LXDR to be set to a logic one. When LXDR = 1, LDIR = 0, and LDCTRL = 0, any data written to Bits 0-10 are directed to the Transmit Data Rate Register. Note that loading of both the Receive and Transmit Data Rate Registers is enabled when LDCTRL = 0, LDIR = 0, LRDR = 1, and LXDR = 1; thus these two registers may be loaded simultaneously when data are received and transmitted at the same rate. LXDR is also set to a logic one when a datum is written to Bit 31 (RESET); however, Transmit Data Rate Register loading is not enabled until LDCTRL and LDIR have been reset to logic zero. Writing a zero to Bit 11 causes LXDR to be reset to logic zero, disabling loading of the Transmit Data Rate Register. Since Bit 11 is the next bit addressed after loading the Transmit Data Rate Register, the register may be loaded and the LXDR flag reset with a single LDCR instruction where 12 bits (Bits 0-11) are written, with a zero written to Bit 11.

3.1.1 Control Register

The Control Register is loaded to select character length, device clock operation, parity, and the number of stop bits for the transmitter. Table 3 shows the bit address assignments for the Control Register.

TABLE 3
CONTROL REGISTER BIT ADDRESS ASSIGNMENTS

$ADDRESS_{10}$	NAME	DESCRIPTION
7	SBS1	← Stop Bit Select
6	SBS2	
5	PENB	Parity Enable
4	PODD	Odd Parity Select
3	CLK4M	$\overline{\phi}$ Input Divide Select
2	—	Not Used
1	RCL1	← Character Length Select
0	RCL0	

7	6	5	4	3	2	1	0
SBSI	SBS2	PENB	PODD	CLK4M	NOT USED	RCL1	RCL0
MSB							LSB

Bits 7 and 6 (SBS1 and SBS2) – Stop Bit Selection. The number of stop bits to be appended to each transmitter character is selected by Bits 7 and 6 of the Control Register as shown below. The receiver only tests for a single stop bit, regardless of the status of Bits 7 and 6.

SBS1 BIT 7	SBS2 BIT 6	NUMBER OF TRANSMITTED STOP BITS
0	0	1½
0	1	2
1	0	1
1	1	1

STOP BIT SELECTION

Bits 5 and 4
(PENB and PODD) –

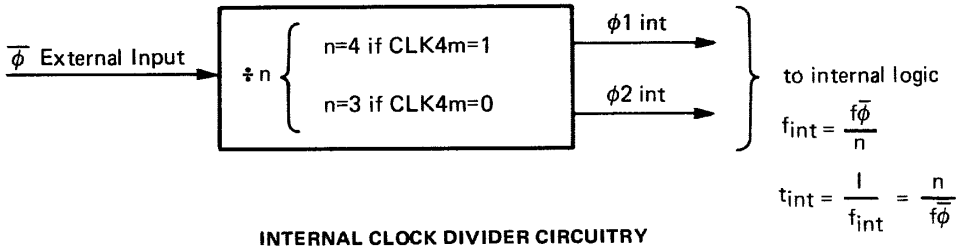
Parity Selection. The type of parity to be generated for transmission and detected for reception is selected by Bits 5 and 4 of the Control Register as shown below. When parity is enabled (PENB = 1), the parity bit is transmitted and received in addition to the number of bits selected for the character length. Odd parity is such that the total number of ones in the character and parity bit, exclusive of stop bit(s), will be odd. For even parity, the total number of ones will be even.

PENB BIT 5	PODD BIT 4	PARITY
0	0	None
0	1	None
1	0	Even
1	1	Odd

PARITY SELECTION

Bit 3 (CLK4M) –

$\overline{\phi}$ Input Divide Select. The $\overline{\phi}$ input to the TMS 9902 ACC is used to generate internal dynamic logic clocking and to establish the time base for the Interval Timer, Transmitter, and Receiver. The $\overline{\phi}$ input is internally divided by either 3 or 4 to generate the two-phase internal clocks required for MOS logic, and to establish the basic internal operating frequency (f_{int}) and internal clock period (t_{int}). When Bit 3 of the Control Register is set to a logic one (CLK4M = 1), $\overline{\phi}$ is internally divided by 4, and when CLK4M = 0, $\overline{\phi}$ is divided by 3. For example, when $f\overline{\phi}$ = 3 MHz, as in a standard 3 MHz TMS 9900 system, and CLK4M = 0, $\overline{\phi}$ is internally divided by 3 to generate an internal clock period t_{int} of 1 μs. The figure below shows the operation of the internal clock divider circuitry. The internal clock frequency should be no greater than 1.1 MHz; thus, when $f\overline{\phi} > 3.3$ MHz, CLK4M should be set to a logic one.



INTERNAL CLOCK DIVIDER CIRCUITRY

Bits 1 and 0
(RCL1 and RCL0) –

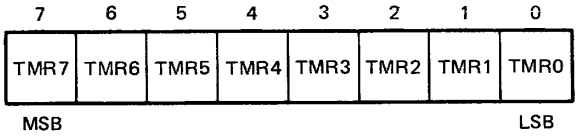
Character Length Select. The number of data bits in each transmitted and received character is determined by Bits 1 and 0 of the Control Register as shown below.

RCL1 BIT 1	RCL0 BIT 0	CHARACTER LENGTH
0	0	5 Bits
0	1	6 Bits
1	0	7 Bits
1	1	8 Bits

CHARACTER LENGTH SELECTION

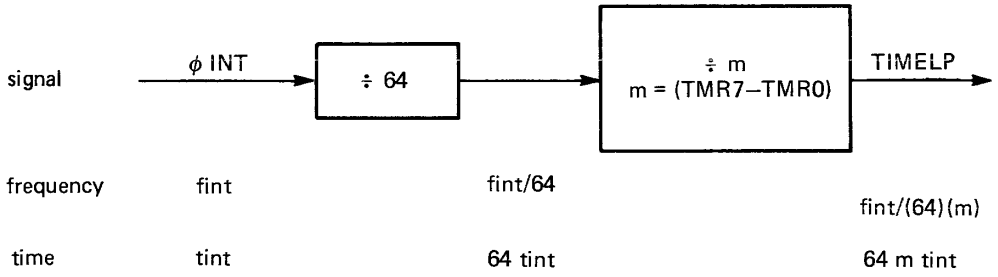
3.1.2 Interval Register

The Interval Register is enabled for loading whenever LDCTRL = 0 and LDIR = 1. The Interval Register is used for selecting the rate at which interrupts are generated by the Interval Timer of the ACC. The figure below shows the bit address assignments for the Interval Register when enabled for loading.



INTERVAL REGISTER BIT ADDRESS ASSIGNMENTS

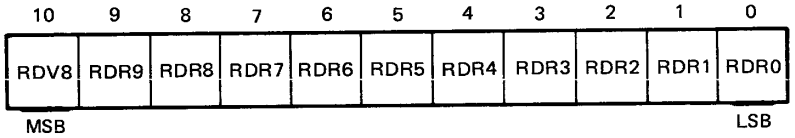
The figure below illustrates the establishment of the interval for the Interval Timer. As an example, if the Interval Register is loaded with a value of 80_{16} (128_{10}) the interval at which Timer Interrupts are generated is $t_{ITVL} = t_{int} \cdot 64 \circ M = (1\ \mu s)\ (\cdot\ 64)(\cdot\ 128) = 8.192$ ms. when $t_{int} = 1\ \mu s$.



TIME INTERAL SELECTION

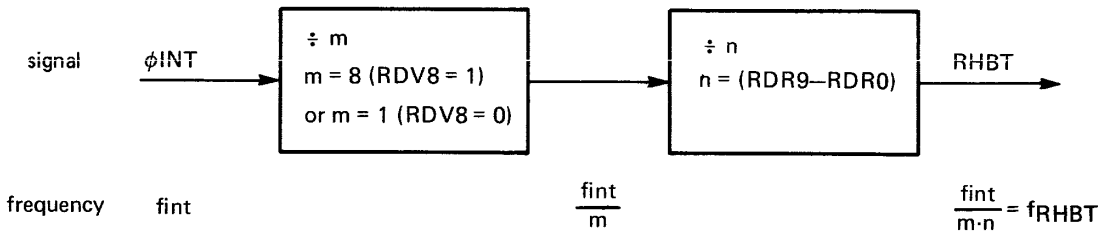
3.1.3 Receive Data Rate Register

The Receive Data Rate Register is enabled for loading whenver LDCTRL = 0, LDIR = 0, and LRDR = 1. The Receive Data Rate Register is used for selecting the bit rate at which data is received. The diagram shows the bit address assignments for the Receive Data Rate Register when enabled for loading.



RECEIVE DATA RATE REGISTER BIT ADDRESS ASSIGNMENTS

The following diagram describes the manner in which the receive data rate is established. Basically, two programmable counters are used to determine the interval for one-half the bit period of receive data. The first counter either divides the internal system clock frequency (f_{int}) by either 8 (RDV8 = 1) or 1 (RDV8 = 0). The second counter has ten stages and may be programmed to divide its input signal by any value from 1 (RDR9–RDR0 = 0000000001) to 1023 (RDR8–RDR0 = 1111111111). The frequency of the output of the second counter (f_{RHBT}) is double the receive-data rate. Register is loaded with a value of 11000111000, RDV8 = 1, and RDR9–RDR0 = $1000111000 = 238_{16} = 568\ 10$. Thus, for f_{int} = 1 MHz, the receive-data rate = $1 \times 10^6 \div 8 \div 568 \div 2 = 110.04$ bits per second.



RECEIVE DATA RATE SELECTION

Quantitatively, the receive-data rate f_{RCV} may be described by the following algebraic expression:

$$f_{RCV} = \frac{f_{RHBT}}{2} = \frac{f_{int}}{2mn} = \frac{f_{int}}{(2)\ (8^{RDV8})\ (RDR9–RDR0)}$$

3.1.4 Transmit Data Rate Register

The Transmit Data Rate Register is enabled for loading whenver LDCTRL = 0, LDIR = 0, and LXDR = 1. The Transmit Data Rate Register is used for selecting the data rate for the transmitter. The figure below shows the bit address assignments for the Transmit Data Rate Register.

10	9	8	7	6	5	4	3	2	1	0
XDV8	XDR9	XDR8	XDR7	XDR6	XDR5	XDR4	XDR3	XDR2	XDR1	XDR0
MSB										LSB

Selection of transmit data rate is accomplished with the Transmit Data Rate Register in the same way that the receive data rate is selected with the Receive Data Rate Register. The algebraic expression for the Transmit Data Rate f_{XMT} is:

$$f_{XMT} = \frac{f_{XHBT}}{2} = \frac{f_{int}}{(2)\ (8^{XDV8})\ (XDR9\text{-}XDR0)}$$

For example, if the Transmit Data Rate Register is loaded with a value of 00110100001, XDV8 = 0, and XDR9–XDR0= $1A1_{16}$ = 417, the transmit data rate = 1 X $10^6 \div 2 \div 1 \div 417$ = 1199.04 bits per second.

3.1.5 Transmit Buffer Register

The Transmit Buffer Register is enabled for loading when LDCTRL = 0, LDIR = 0, LRDR = 0, LXDR = 0, and BRKON = 0. The Transmit Buffer Register is used for storage of the next character to be transmitted. When the transmitter is active, the contents of the Transmit Buffer Register are transferred to the Transmit Shift Register each time the previous character has been completely transmitted. The bit address assignments for the Transmit Buffer Register are shown below:

7	6	5	4	3	2	1	0
XBR7	XBR6	XBR5	XBR4	XBR3	XBR2	XBR1	XBR0
MSB							LSB

TRANSMIT BUFFER REGISTER BIT ADDRESS ASSIGNMENTS

All 8 bits should be transferred into the register, regardless of the selected character length. The extraneous high-order bits will be ignored for transmission purposes; however, loading of bit 7 is internally detected to cause the Transmit Buffer Register Empty (XBRE) status flag to be reset.

3.2 STATUS AND DATA INPUT

Status and data information is read from the ACC using $\overline{CE}$, S0–S4, and CRUIN. The following figure illustrates the relationship of the signals used to access data from the ACC. Table 5 describes the input bit address assignments for the ACC.

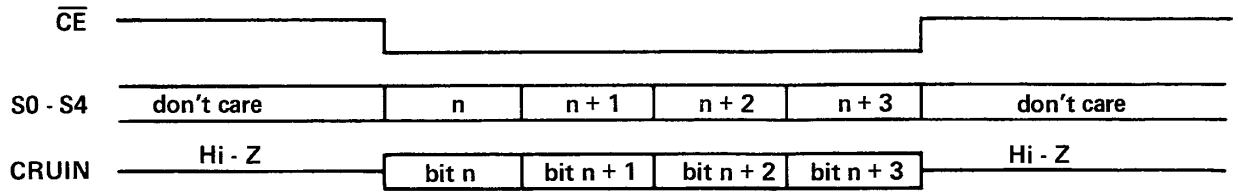
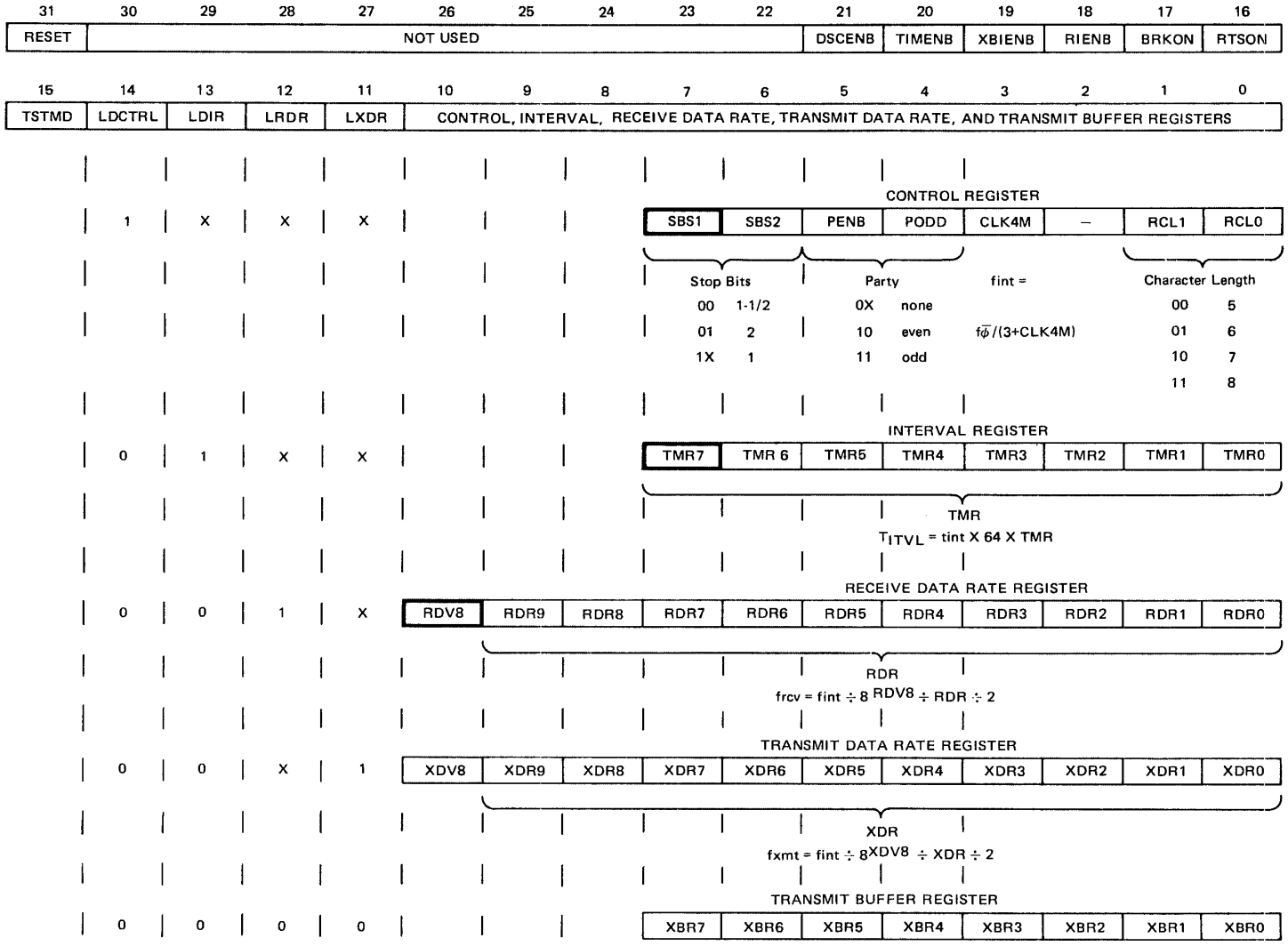


TABLE 4. CRU OUTPUT BIT ADDRESS ASSIGNMENTS



NOTE 1: LOADING OF THE BIT INDICATED BY ☐ CAUSES THE LOAD CONTROL FLAG FOR THAT REGISTER TO BE AUTOMATICALLY RESET.

TABLE 5
TMS 9902 ACC INPUT BIT ADDRESS ASSIGNMENTS

$ADDRESS_9$ S0	S1	S2	S3	S4	$ADDRESS_{10}$	NAME	DESCRIPTION
1	1	1	1	1	31	INT	Interrupt
1	1	1	1	0	30	FLAG	Register Load Control Flag Set
1	1	1	0	1	29	DSCH	Data Set Status Change
1	1	1	0	0	28	CTS	Clear to Send
1	1	0	1	1	27	DSR	Data Set Ready
1	1	0	1	0	26	RTS	Request to Send
1	1	0	0	1	25	TIMELP	Timer Elapsed
1	1	0	0	0	24	TIMERR	Timer Error
1	0	1	1	1	23	XSRE	Transmit Shift Register Empty
1	0	1	1	0	22	XBRE	Transmit Buffer Register Empty
1	0	1	0	1	21	RBRL	Receive Buffer Register Loaded
1	0	1	0	0	20	DSCINT	Data Set Status Charge Interrupt (DSCH • DSCENB)
1	0	0	1	1	19	TIMINT	Timer Interrupt (TIMELP • TIMENB)
1	0	0	1	0	18	—	Not used (always = 0)
1	0	0	0	1	17	XBINT	Transmitter Interrupt (XBRE • XBIENB)
1	0	0	0	0	16	RBINT	Receiver Interrupt (RBRL • RIENB)
0	1	1	1	1	15	RIN	Receive Input
0	1	1	1	0	14	RSBD	Receive Start Bit Detect
0	1	1	0	1	13	RFBD	Receive Full Bit Detect
0	1	1	0	0	12	RFER	Receive Framing Error
0	1	0	1	1	11	ROVER	Receive Overrun Error
0	1	0	1	0	10	RPER	Receive Parity Error
0	1	0	0	1	9	RCVERR	Receive Error
0	1	0	0	0	8	—	Not used (always = 0)
					7-0	RBR7-RBR0	Receive Buffer Register (Received Data)

Bit 31 (INT) – INT = DSCINT + TIMINT + XBINT + RBINT. The interrupt output ($\overline{INT}$) is active when this status signal is a logic 1.

Bit 30 (FLAG) – FLAG = LDCTRL + LDIR + LRDR + LXDR + BRKON. When any of the register load control flags or BRKON is set, FLAG = 1.

Bit 29 (DSCH) – Data Set Status Change Enable. DSCH is set when the $\overline{DSR}$ or $\overline{CTS}$ input changes state. To ensure recognition of the state change, $\overline{DSR}$ or $\overline{CTS}$ must remain stable in its new state for a minimum of two internal clock cycles. DSCH is reset by an output to bit 21 (DSCENB).

Bit 28 (CTS) – Clear to Send. The CTS signal indicates the inverted status of the $\overline{CTS}$ device input.

Bit 27 (DSR) – Data Set Ready. The DSR signal indicates the inverted status of the $\overline{DSR}$ device input.

Bit 26 (RTS) – Request to Send. The RTS signal indicates the inverted status of the $\overline{RTS}$ device output.

Bit 25 (TIMELP) – Timer Elapsed. TIMELP is set each time the Interval Timer decrements to 0. TIMELP is reset by an output to bit 20 (TIMENB).

Bit 24 (TIMERR) – Timer Error. TIMERR is set whenever the Interval timer decrements to 0 and TIMELP is already set, indicating that the occurrence of TIMELP was not recognized and cleared by the CPU before subsequent intervals elapsed. TIMERR is reset by an output to bit 20 (TIMENB).

Bit 23 (XSRE) – Transmit Shift Register Empty. When XSRE = 1, no data is currently being transmitted and the XOUT output is at logic 1 unless BRKON is set. When XSRE = 0, transmission of data is in progress.

Bit 22 (XBRE) – Transmit Buffer Register Empty. When XBRE = 1, the transmit buffer register does not contain the next character to be transmitted. XBRE is set each time the contents of the transmit buffer register are transferred to the transmit shift register, XBRE is reset by an output to bit 7 of the transmit buffer register (XBR7), indicating that a character has been loaded.

Bit 21 (RBRL) – Receive Buffer Register Loaded. RBRL is set when a complete character has been assembled in the receive shift register and the character is transferred to the receive buffer register. RBRL is reset by an output to bit 18 (RIENB).

Bit 20 (DSCINT) – Data Set Status Change Interrupt. DSCINT = DSCH (input bit 29) • DSCENB (output bit 21). DSCINT indicates the presence of an enabled interrupt caused by the changing of state of $\overline{\text{DSR}}$ or $\overline{\text{CTS}}$.

Bit 19 (TIMINT) – Timer Interrupt. TIMINT = TIMELP (input bit 25) • TIMENB (output bit 20). TIMINT indicates the presence of an enabled interrupt caused by the interval timer.

Bit 17 (XBINT) – Transmitter Interrupt. XBINT = XBRE (input bit 22) • XBIENB (output bit 19). XBINT indicates the presence of an enabled interrupt caused by the transmitter.

Bit 16 (RBINT) – Receiver Interrupt. RBINT = RBRL (input bit 21) • RIENB (output bit 18). RBINT indicates the presence of an enabled interrupt caused by the receiver.

Bit 15 (RIN) – Receive Input. RIN indicates the status of the RIN input to the device.

Bit 14 (RSBD) – Receive Start Bit Detect. RSBD is set one-half bit time after the 1-to-0 transition of RIN indicating the start bit of a character. If RIN is not still 0 at this point in time, RSBD is reset. Otherwise, RSBD remains true until the complete character has been received. This bit is normally used for testing purposes.

Bit 13 (RFBD) – Receive Full Bit Detect. RFBD is set one bit time after RSBD is set to indicate the sample point for the first data bit of the received character. RSBD is reset when the character has been completely received. This bit is normally used for testing purposes.

Bit 12 (RFER) – Receive Framing Error. RFER is set when a character is received in which the stop bit, which should be a logic 1, is a logic 0. RFER should only be read when RBRL (input bit 21) is a 1. RFER is reset when a character with the correct stop bit is received.

Bit 11 (ROVER) – Receive Overrun Error. ROVER is set when a new character is received before the RBRL flag (input bit 21) is reset, indicating that the CPU failed to read the previous character and reset RBRL before the present character is completely received. ROVER is reset when a character is received and RBRL is 0 when the character is transferred to the receive buffer register.

Bit 10 (RPER) – Receive Parity Error. RPER is set when a character is received in which the parity is incorrect. RPER is reset when a character with correct parity is received.

Bit 9 (RCVERR) – Receive Error. RCVERR = RFER + ROVER + RPER. RCVERR indicates the presence of an error in the most recently received character.

Bit 7–Bit 0 (RBR7–RBR0) – Receive Buffer Register. The receive buffer register contains the most recently received character. For character lengths of fewer than 8 bits the character is right justified, with unused most significant bit(s) all zero(es). The presence of valid data in the receive buffer register is indicated when RBRL is a logic 1.

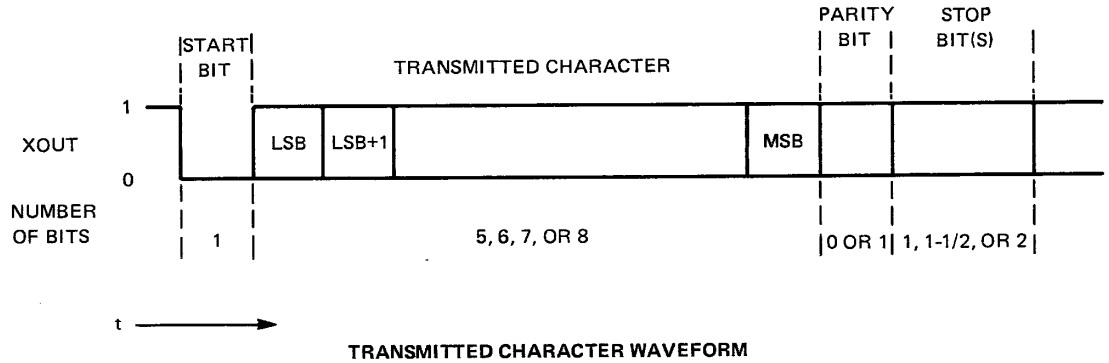
3.3 TRANSMITTER OPERATION

3.3.1 Transmitter Initialization

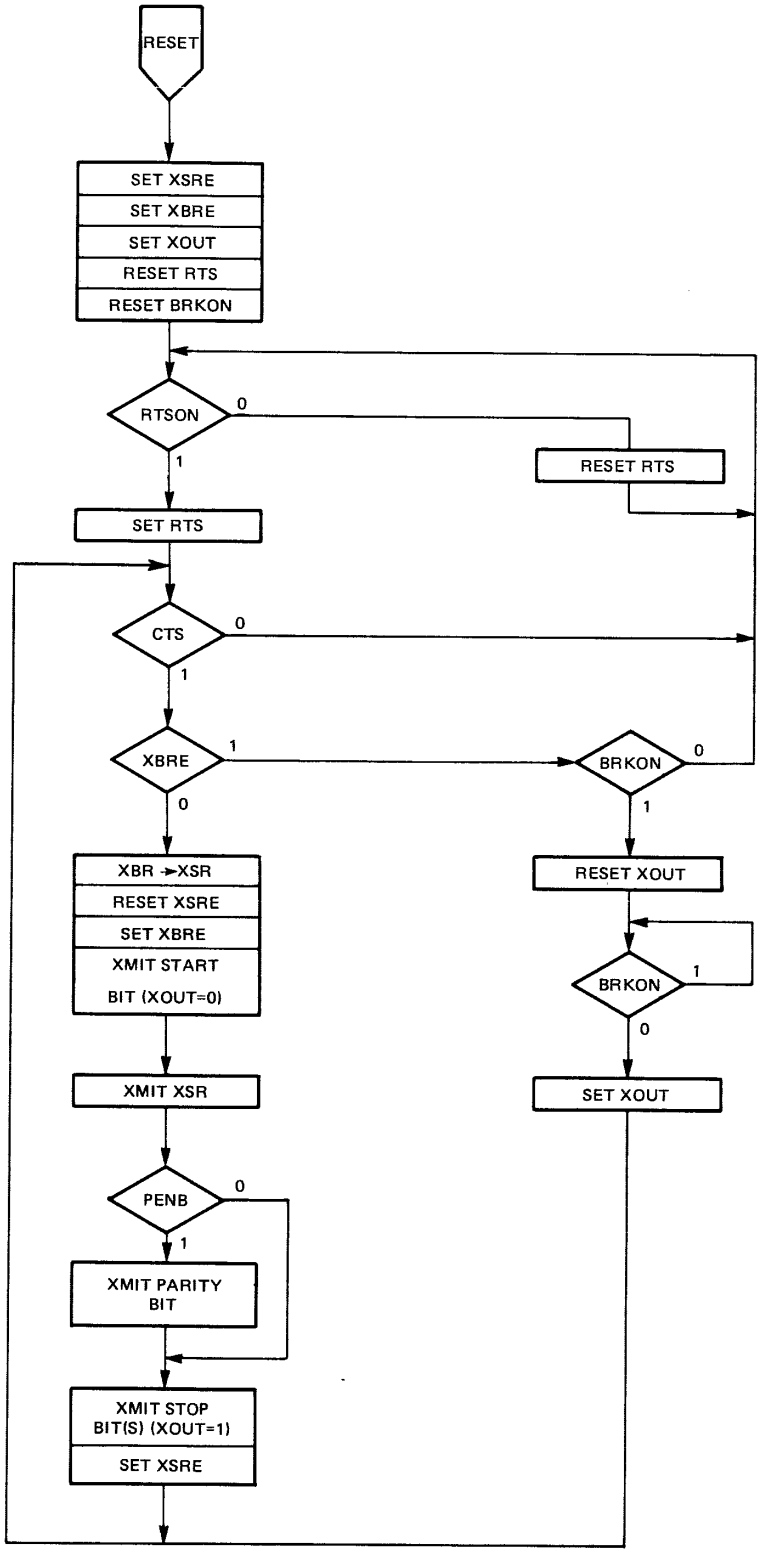
The operation of the transmitter is described in the following flow chart. The transmitter is initialized by issuing the RESET command (output to bit 31), which cause the internal signals XSRE and XBRE to be set, and BRKON to be reset. Device outputs $\overline{RTS}$ and XOUT are set, placing the transmitter in its idle state. When RTSON is set by the CPU, the $\overline{RTS}$ output becomes active and the transmitter becomes active when $\overline{CTS}$ goes low.

3.3.2 Data Transmission

If the Transmit Buffer Register contains a character, transmission begins. The contents of the Transmit Buffer Register is transferred to the Transmit Shift Register, causing XSRE to be reset and XBRE to be set. The first bit transmitted (start bit) is always a logic 0. Subsequently, the character is shifted out, LSB first. Only the number of bits specified by RCL1 and RCL0 (character length select) of the Control Register are shifted. If parity is enabled, the correct parity bit is next transmitted. Finally the stop bit(s) selected by SBS1 and SBS0 of the Control Register are transmitted. Stop bits are always logic one. XSRE is set to indicate that no transmission is in progress, and the transmitter again tests XBRE to determine if the CPU has yet loaded the next character. The waveform for a transmitted character is shown below.



TRANSMITTED CHARACTER WAVEFORM



TMS 9902 TRANSMITTER OPERATION

3.3.3 BREAK Transmission

The BREAK message is transmitted only if XBRE = 1, $\overline{CTS}$ = 0, and BRKON = 1. After transmission of the BREAK message begins, loading of the Transmit Buffer Register is inhibited and XOUT is reset. When BRKON is reset by the CPU, XOUT is set and normal operation continues. It is important to note that characters loaded into the Transmit Buffer Register are transmitted prior to the BREAK message regardless of whether the character has been loaded into the Transmit Shift Register before BRKON is set. Any character to be transmitted subsequent to transmission of the BREAK message may not be loaded into the Transmit Buffer Register until after BRKON is reset.

3.3.4 Transmission Termination

Whenever XSRE = 1 and BRKON = 0, the transmitter is idle, with XOUT set to one. If RTSON is reset at this time, the $\overline{RTS}$ device output will go inactive, disabling further data transmission until RTSON is again set. $\overline{RTS}$ will not go inactive, however, until any characters loaded into the Transmit Buffer Register prior to resetting RTSON are transmitted and BRKON = 0.

3.4 RECEIVER OPERATION

3.4.1 Receiver Initialization

Operation of the TMS 9902 receiver is described in the following flowchart. The receiver is initialized any time the CPU issues the RESET command. The RBRL flag is reset to indicate that no character is currently in the Receive Buffer Register, and the RSBD and RFBD flags are reset. The receiver remains in the inactive state until a 1 to 0 transition is detected on the RIN device input.

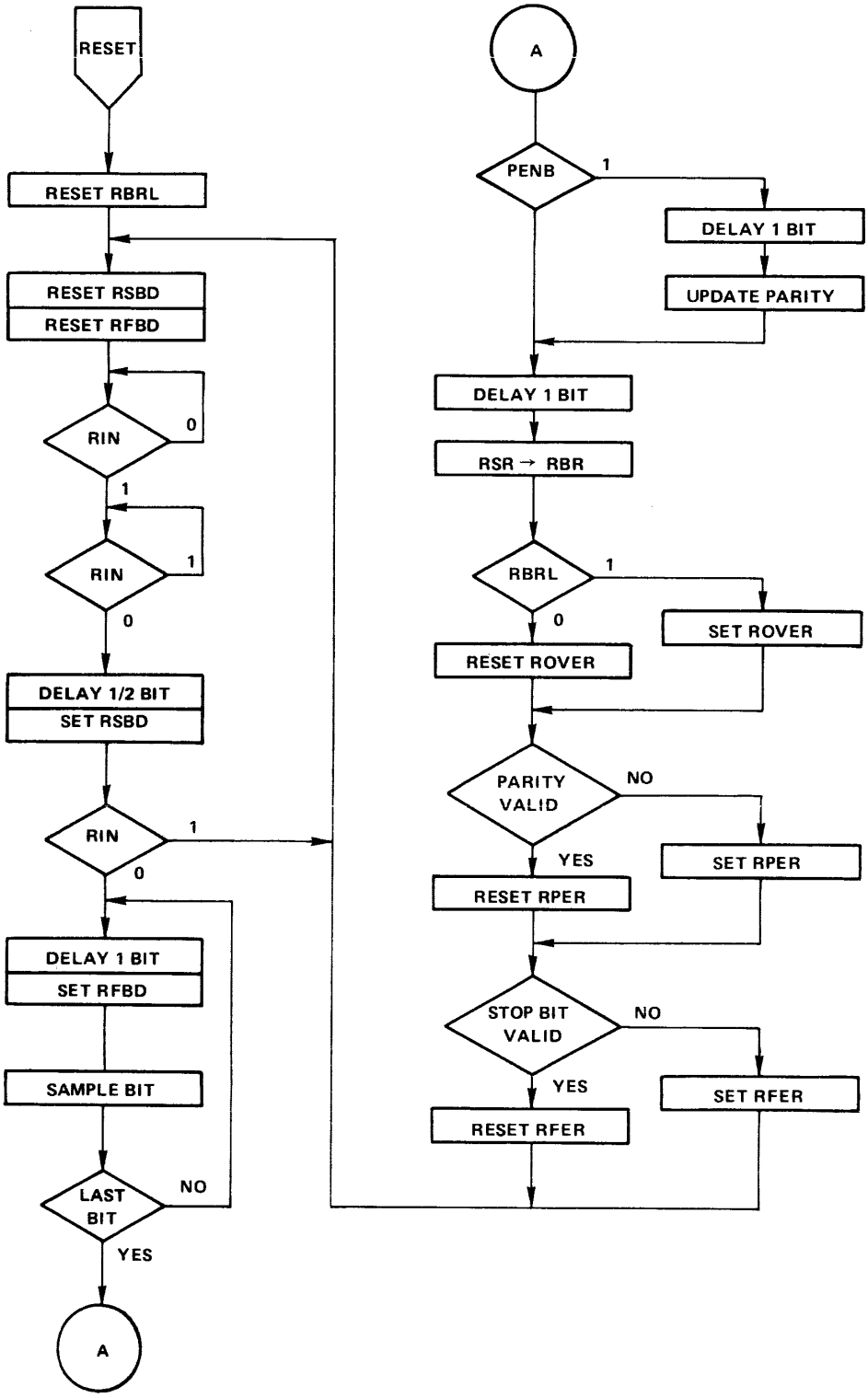
3.4.2 Start Bit Detection

The receiver delays one-half bit time and again samples RIN to ensure that a valid start bit has been detected. If RIN = 0 after the half-bit delay, RSBD is set and data reception begins. If RIN = 1 no data reception occurs.

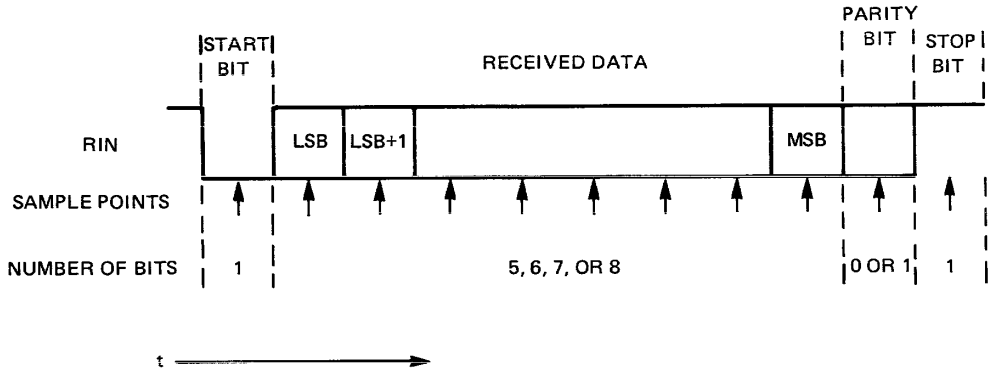
3.4.3 Data Reception

In addition to verifying the valid start bit, the half-bit delay after the 1-to-0 transition also establishes the sample point for all subsequent data bits in this character. Theoretically, the sample point is in the center of each bit cell, thus maximizing the limits of acceptable distortion of data cells. After the first full bit delay the least significant data bit is received and RFBD is set. The receiver continues to delay one-bit intervals and sample RIN until the selected number of bits are received. If parity is enabled one additional bit is read for parity. After an additional bit delay, the received character is transferred to the Receive Buffer Register, RBRL is set, ROVER and RPER are loaded with appropriate values, and RIN is tested for a valid stop bit. If RIN = 1, the stop bit is valid. RFER, RSBD, and RFBD are reset and the receiver waits for the next start bit to begin reception of the next character.

If RIN = 0 when the stop bit is sampled, RFER is set to indicate the occurrence of a framing error. RSBD and RFBD are reset but sampling for the start bit of the next character does not begin until RIN = 1.



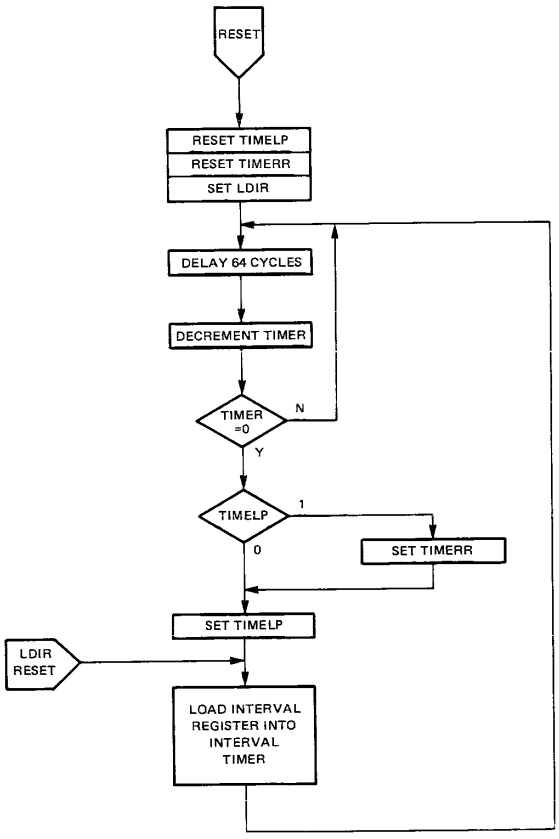
TMS 9902 RECEIVER OPERATION



CHARACTER RECEPTION TIMING

3.5 INTERVAL TIMER OPERATION

A flowchart of the operation of the Interval Timer is shown below. Execution of the RESET command by the CPU causes TIMELP and TIMERR to be reset and LDIR to be set. Resetting LDIR causes the contents of the Interval Register to be loaded into the Interval Timer, thus beginning the selected time interval. The timer is decremented every 64 internal clock cycles (every 2 internal clock cycles when in Test Mode) until it reaches zero, at which time the Interval Timer is reloaded by the Interval Register and TIMELP is set. If TIMELP was already set, TIMERR is set to indicate that TIMELP was not cleared by the CPU before the next time period elapsed. Each time LDIR is reset the contents of the Interval Register are loaded into the Interval Timer, thus restarting the timer.



INTERVAL TIMER OPERATION

4. DEVICE APPLICATION

This section describes the software interface between the CPU and the TMS 9902 ACC and discusses some of the design considerations in the use of this device in asynchronous communications applications.

4.1 DEVICE INITIALIZATION

The ACC is initialized by the CPU issuing the RESET command, followed by loading the Control, Interval, Receive Data Rate, and Transmit Data Rate registers. Assume that the value to be loaded into the CRU Base Register (register 12) in order to point to bit 0 is 0040_{16}. In this application, characters will have 7 bits of data plus even parity and one stop bit. The $\overline{\phi}$ input to the ACC is a 3 MHz signal. The ACC will divide this signal frequency by 3 to generate an internal clock frequency of 1 MHz. An interrupt will be generated by the Interval Timer every 1.6 milliseconds when timer interrupts are enabled. The transmitter will operate at a data rate of 300 bits per second, and the receiver will operate at 1200 bits per second.

Had it been desired that both the transmitter and receiver operate at 300 bits per second, the "LDCR @RDR,11" instruction would have been deleted, and the "LDCR @XDR,12" instruction would have caused both data rate registers to be loaded and LRDR and LXDR to have been reset.

4.1.1 Initialization Program

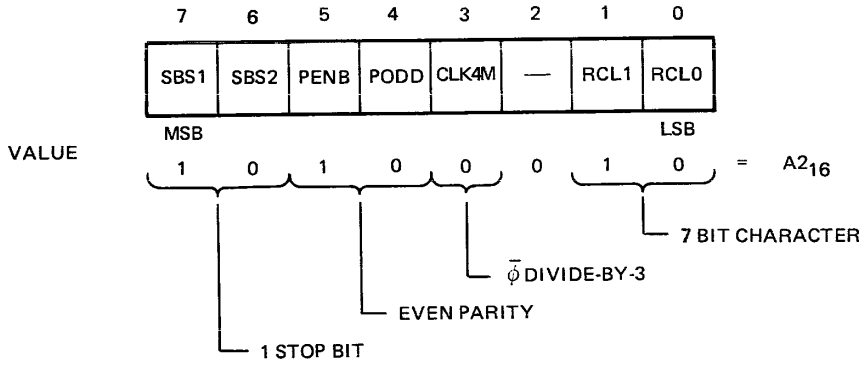
The initialization program for the configuration previously described is as shown below. The RESET command disables all interrupts, initializes all controllers, sets the four register load control flags (LDCTRL, LDIR, LRDR, and LXDR). Loading the last bit of each of the registers causes the load control flag to be automatically reset.

```
        LI      R12, >40        INITIALIZE CRU BASE
        SBO     31              RESET COMMAND
        LDCR    @CNTRL, 8       LOAD CONTROL AND RESET LDCTRL
        LDCR    @INTVL, 8       LOAD INTERVAL AND RESET LDIR
        LDCR    @RDR, 11        LOAD RDR AND RESET LRDR
        LDCR    @XDR, 12        LOAD XDR AND RESET LXDR
          .
          .
          .
CNTRL   BYTE    >A2
INTVL   BYTE    1600/64
RDR     DATA    >1A1
XDR     DATA    >4DO
```

The RESET command initializes all subcontrollers, disables interrupts, and sets LDCTRL, LDIR, LRDR, and LXDR, enabling loading of the control register.

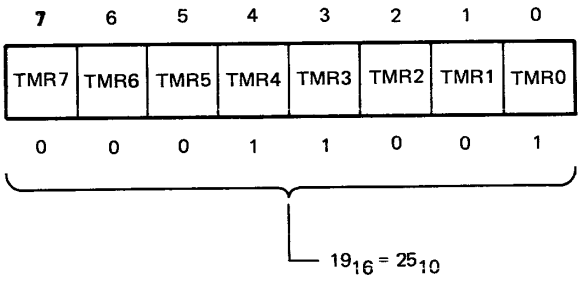
4.1.2 Control Register

The options described previously are selected by loading the value shown below.



4.1.3 Interval Register

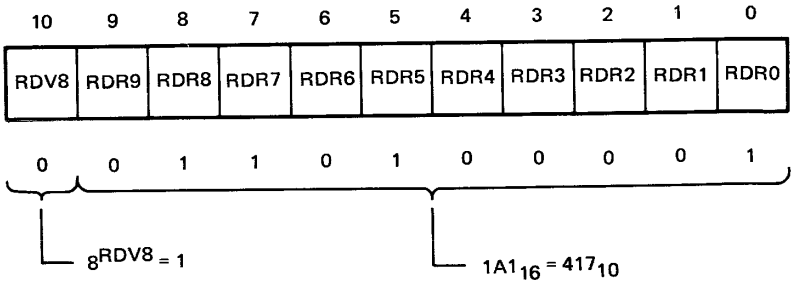
The interval register is to be set up to generate an interrupt every 1.6 milliseconds. The value loaded into the interval register specifies the number of 64 microsecond increments in the total interval.



25 X 64 MICROSECONDS = 1.6 MILLISECONDS

4.1.4 Receive Data Rate Register

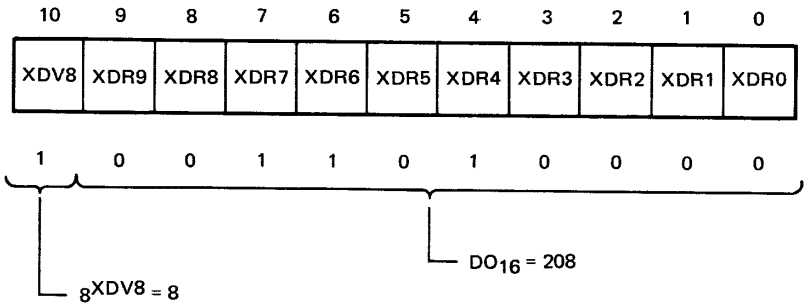
The data rate for the receiver is to be 1200 bits per second. The value to be loaded into the receive data rate register is as shown:



$10^6 \div 1 \div 417 \div 2 = 1199.04$ BITS PER SECOND

4.1.5 Transmit Data Rate Register

The data rate for the transmitter is to be 300 bits per second. The value to be loaded into the transmit data rate register is:



$1 \times 10^6 \div 8 \div 208 \div 2 = 300.48$ BITS PER SECOND

4.2 DATA TRANSMISSION

The subroutine shown below demonstrates a simple loop for the transmitting of a block of data.

```
        LI      RO, LISTAD      INITIALIZE LIST POINTER
        LI      R1, COUNT       INITIALIZE BLOCK COUNT
        LI      R12, CRUBAS     INITIALIZE CRU BASE
        SBO     16              TURN ON TRANSMITTER
XMTLP   TB      22              WAIT FOR XBRE = 1
        JNE     XMTLP
        LDCR    *RO+,8          LOAD CHARACTER INCREMENT POINTER RESET XBRE
        DEC     R1              DECREMENT COUNT
        JNE     XMTLP           LOOP IF NOT COMPLETE
        SBZ     16              TURN OFF TRANSMITTER
```

After initializing the list pointer, block count, and CRU base address. RTSON is set to cause the transmitter and the $\overline{\text{RTS}}$ output to become active. Data transmission does not begin, however, until the $\overline{\text{CTS}}$ input becomes active. After the final character is loaded into the transmit buffer register, RTSON is reset. The transmitter and the $\overline{\text{RTS}}$ output do not become inactive until the final character has been completely transmitted.

4.3 DATA RECEPTION

The software shown below will cause a block of data to be received and stored in memory.

```
CARRET   BYTE   >OD
RCVBLK   LI     R2, RCVLST      INITIALIZE LIST COUNT
         LI     R3, MXRCNT      INITIALIZE MAX COUNT
         LI     R4, CARRET      SET UP END OF BLOCK CHARACTER
RCVLP    TB     21              WAIT FOR RBRL=1
         JNE    RCVLP
         STCR   *R2,8           STORE CHARACTER
         SBZ    18              RESET RBRL
         DEC    R3              DECREMENT COUNT
         JEQ    RCVEND          END IF COUNT=0
         CB     *R2+, R4        COMPARE TO EOB CHARACTER, INCREMENT POINTER
         JNE    RCVLP           LOOP IF NOT COMPLETE
RCVEND   RT                     END OF SUBROUTINE
```

4.4 REGISTER LOADING AFTER INITIALIZATION

The control, interval, and data rate registers may be reloaded after initialization. For example, it may be desirable to change the interval of the timer. Assume, for sample, that the interval is to be changed to 10.24 milliseconds. The instruction sequence is as follows:

```
         SBO    13              SET LOAD CONTROL FLAG
         LDCR   @INTVL2,8       LOAD REGISTER, RESET FLAG
          .
          .
          .
INTVL2   BYTE   10240/64
```

Caution should be exercised when transmitter interrupts are enabled to ensure that the transmitter interrupt does not occur while the load control flag is set. For example, if the transmitter interrupts between execution of the "SBO 13" and the next instruction, the transmit buffer is not enabled for loading when the transmitter interrupt service routine is entered because the LDIR flag is set. This situation may be avoided by the following sequence:

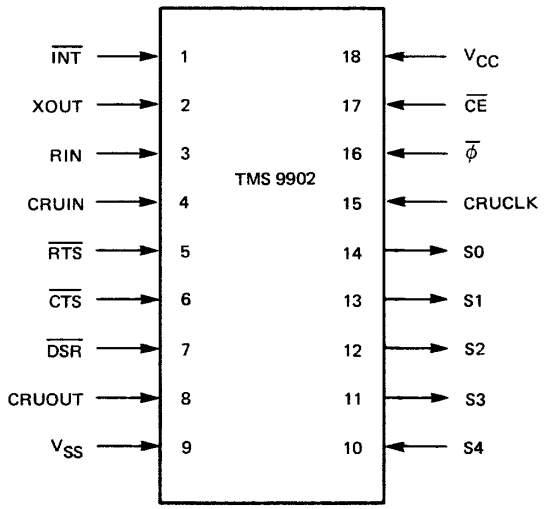
```
         BLWP   @ITVCHG         CALL SUBROUTINE
          .
          .
          .
ITV CPC  LI MI  0               MASK ALL INTERRUPTS
         MOV    @24(R13), RIZ   LOAD CRU BASE ADDRESS
         SBO    13              SET FLAG
         LDCR   @INTVL2, 8      LOAD REGISTER AND RESET FLAG
         RTWP                   RESTORE MASK AND RETURN
          .
          .
          .
ITVCHG   DATA   ACCWP, ITVCPC
INTVL2   BYTE   10240/64
```

In this case all interrupts are masked, ensuring that all interrupts are disabled while the load control flag is set.

4.5 TMS 9902 PIN ASSIGNMENTS AND FUNCTIONS

SIGNATURE	PIN	I/O	DESCRIPTION
$\overline{INT}$	1	O	Interrupt – when active (low), the $\overline{INT}$ output indicates that at least one of the interrupt conditions has occured.
XOUT	2	O	Transmitter serial data output line – XOUT remains inactive (high) when TMS 9902 is not transmitting.
RIN	3	I	Receiver serial data input line – RCV – must be held in the inactive (high) state when not receiving data. A transition from high to low will activate the receiver circuitry.
CRUIN	4	O	Serial data output pin from TMS 9902 to CRUIN input pin of the CPU.
$\overline{RTS}$	5	O	Request-to-send output from TMS 9902 to modem. This output is enabled by the CPU and remains active (low) during transmission from the TMS 9902.
$\overline{CTS}$	6	I	Clear-to-send input from modem to TMS 9902. When active (low), it enables the transmitter section of TMS 9902.
$\overline{DSR}$	7	I	Data set ready input from modem to TMS 9902. This input generates an interrupt when going On or Off.
CRUOUT	8	I	Serial data input line to TMS 9902 from CRUOUT line of the CPU.
V_{SS}	9	I	Ground reference voltage.
S4 (LSB)	10	I	
S3	11	I	
S2	12	I	
S1	13	I	
S0	14	I	Address bus S0-S4 are the lines that are addressed by the CPU to select a particular TMS 9902 function.
CRUCLK	15	I	CRU Clock. When active (high), TMS 9902 from CRUOUT line of the CPU.
$\overline{\phi}$	16	I	TTL Clock.
$\overline{CE}$	17	I	Chip enable – when CE is inactive (high), the TMS 9902 address decoding is inhibited which prevents execution of any TMS 9902 command function. CRUIN remains at high-impedance when $\overline{CE}$ is inactive (high). .
V_{CC}	18	I	Supply voltage (+5 V nominal).

TMS 9902
18-PIN PACKAGE



5. TMS 9902 ELECTRICAL SPECIFICATIONS

5.1 ABSOLUTE MAXIMUM RATINGS OVER OPERATING FREE AIR TEMPERATURE RANGE (UNLESS OTHERWISE NOTED)*

Supply voltage, V_{CC} . . . . . −0.3 V to 10 V
All Inputs and Output Voltages . . . . . −0.3 V to 10 V
Continuous Power Dissipation . . . . . 0.7 W
Operating Free-Air Temperature Range . . . . . 0°C to 70°C
Storage Temperature Range . . . . . −65°C to 150°C

*Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the "Recommended Operating Conditions" section of this specification is not implied. Exposure to Absolute Maximum Rated conditions for extended periods may affect device reliability.

5.2 RECOMMENDED OPERATING CONDITIONS

	MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}	4.75	5	5.25	V
Supply voltage, V_{SS}		0		V
High-level input voltage, V_{IH}		2.2		V
Low-level input voltage, V_{IL}		0.6		V
Operating free-air temperature, T_A	0		70	°C

5.3 ELECTRICAL CHARACTERISTICS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS (UNLESS OTHERWISE NOTED)

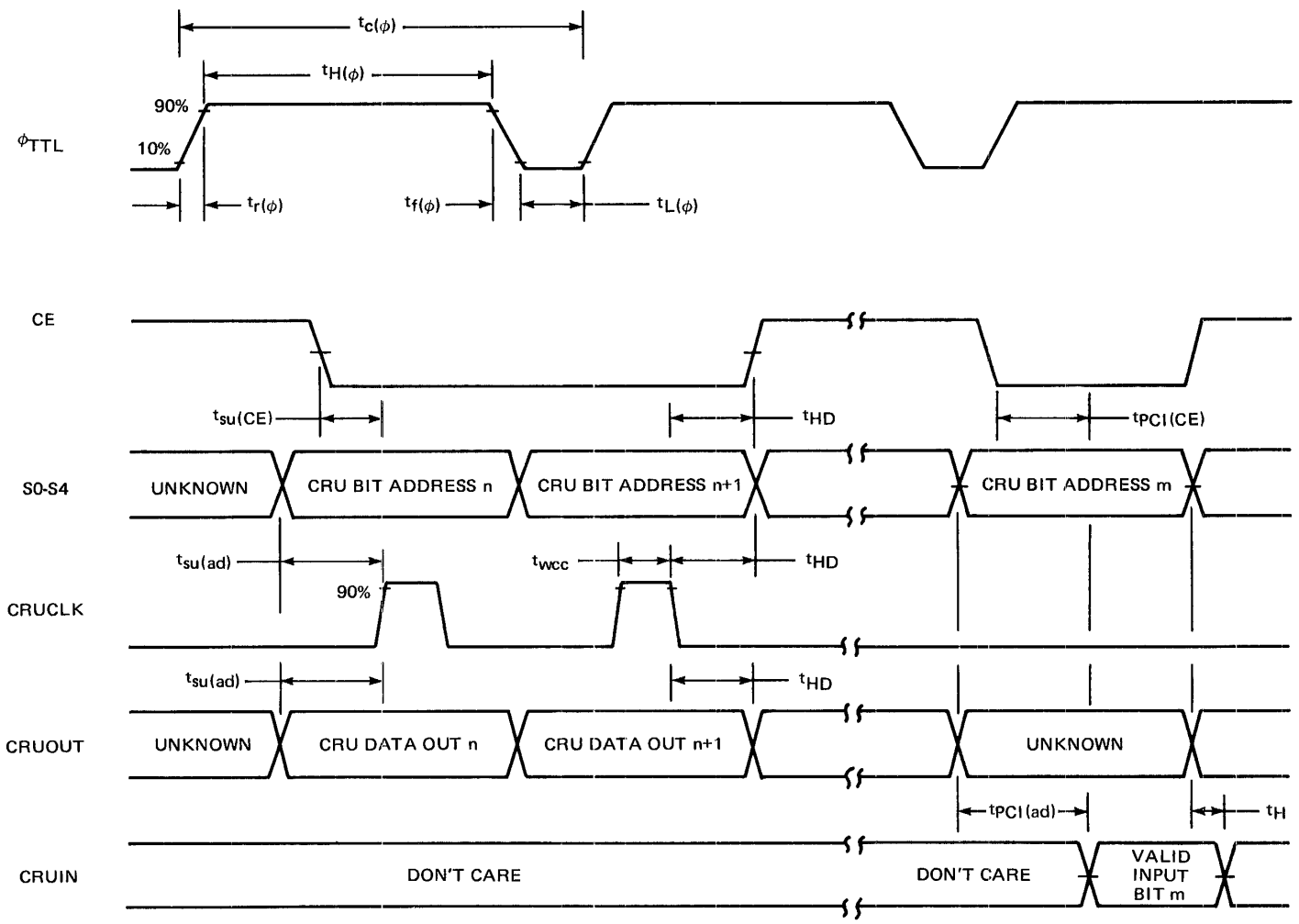
PARAMETERS		TEST CONDITIONS	MIN	TYP	MAX	UNIT
I_I	Input current (any input)	V_I = 0 V to V_{CC}		±10		μA
V_{OH}	High-level output voltage	I_{OH} = −100 μA		2.4		V
		I_{OH} = −400 μA		2.0		
V_{OL}	Low-level output voltage	I_{OL} = 3.2 mA		0.4		V
$I_{CC(AV)}$	Average supply current from V_{CC}	$t_{c(\phi)}$ = 250 ns, T_A = 25°C		100		mA
C_i	Capacitance, any input	f = 1 MHz, All other pins at 0 V		10		pF
C_o	Capacitance, any output			20		

5.4 TIMING REQUIREMENTS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS

PARAMETER		MIN	TYP	MAX	UNIT
$t_{c(\phi)}$	Clock cycle time		333	2000	ns
$t_{r(\phi)}$	Clock rise time		12		ns
$t_{f(\phi)}$	Clock fall time		12		ns
$t_{H(\phi)}$	Clock pulse width (high level)		240		ns
$t_{L(\phi)}$	Clock pulse width (low level)		55		ns
$t_{su(ad)}$	Setup time for address and CRUOUT before CRUCLK		220		ns
$t_{su(CE)}$	Setup time for CE before CRUCLK		180		ns
t_{HD}	Hold time for address, CE and CRUOUT after CRUCLK		80		ns
t_{wcc}	CRUCLK pulse width		100		ns

5.5 SWITCHING CHARACTERISTICS OVER FULL RANGE OF RECOMMENDED OPERATING CONDITIONS

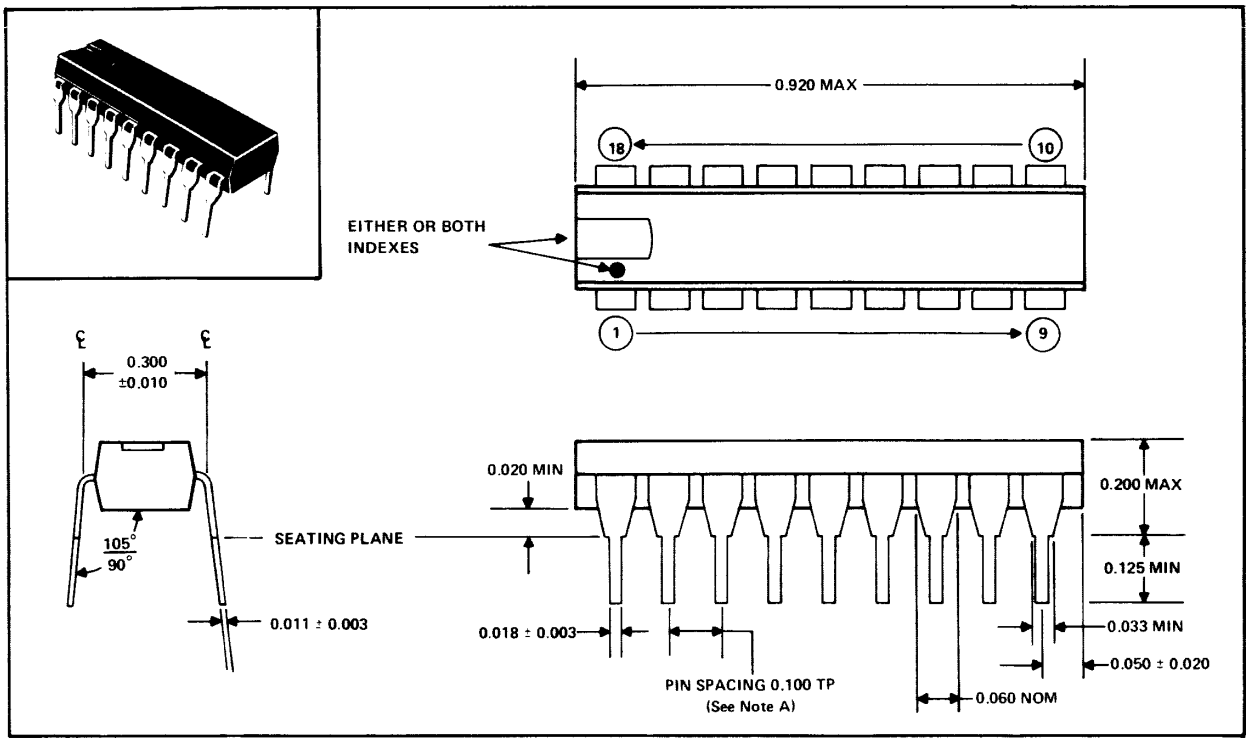
PARAMETERS		TEST CONDITIONS	MIN	TYP	MAX	UNIT
$t_{PCI(ad)}$	Propagation delay, address-to-valid CRUIN	C_L = 100 pF		400		ns
$t_{PCI(CE)}$	Propagation delay, $\overline{CE}$-to-valid CRUIN	C_L = 100 pF		400		ns
t_H	CRUIN hold time after address			20		ns



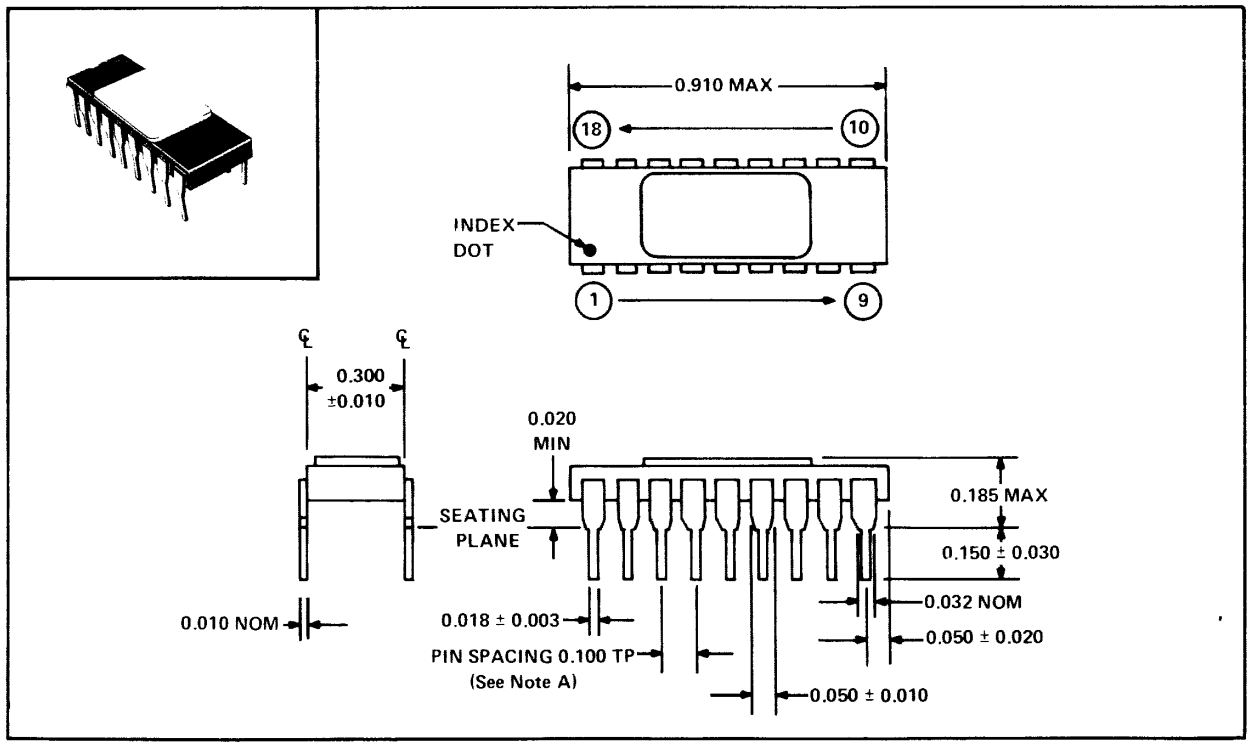
SWITCHING CHARACTERISTICS

6. MECHANICAL DATA

18-PIN PLASTIC DUAL-IN-LINE PACKAGE



18-PIN CERAMIC DUAL-IN-LINE PACKAGE



NOTES: A. Each pin centerline is located within 0.010 of its true longitudinal position.
B. All linear dimensions are in inches.