Inhouse DLC Interface

The inhouse data link control (K201) is an interface enabling your NCR PERSONAL COMPUTER, as the primary station, to communicate over dedicated lines with up to 31 secondary stations, for example, cash terminals (NCR 2126) and dialog stations (NCR 7900).

The primary station continuously polls the secondary terminals, checking whether a data transfer is requested. The secondary terminal can then transmit the requested data, after which it returns to the receive mode. Transmission speed is 48000 bits per second.

The maximum advisable length of cable in the entire network is 3600 metres (12000 feet).

The interface can communicate data with the system microprocessor under DMA or under direct microprocessor I/O control.

INTERFACE CONNECTIONS

The interface uses only 4 wires, two for transmission (Ta, Tb), the other two for receiving (Ra, Rb). These wires are connected to a standard 9-pin D-connector as shown in Figure 5.1.

Wire	Color	Pin
Ta	white	1
Tb	green	2
Ra	yellow	3
Rb	brown	4

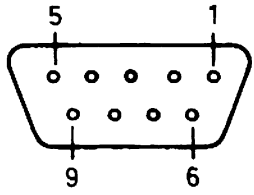


Figure 5.1 DLC Interface Connections

The interface hardware includes two jumpers which select the type of interrupt and the DMA channel used (Figure 5.2).

Jumper		Selection				
W1	W2	IRQ	Int. type		DMA channel	
A - B	A - B	3	11	*	1	*
B - C	B - C	2	10		2	
		* Factory setting				

Figure 5.2 DLC Interface Jumper Settings

DLC CONTROLLER

The DLC interface is controlled by an integrated circuit clocked by a 2.304 MHz oscillator. Figure 5.3 illustrates the integration of this controller in the NCR PERSONAL COMPUTER.

CONTROLLER PIN CONFIGURATION

Figure 5.4 illustrates the pin connections of the DLC integrated circuit. The significance of the individual connections is explained in this section.

CLOCK Input - Pin B20
System clock input frequency of 4.77 MHz.

Reset DRV Input - B2
Initialization (reset) signal synchronized to the falling edge of the clock.

Address lines Input - Pins A22 ... A31
The 10 least significant system address bus lines (A0 to A9). Pin A31 represents the least significant address line.

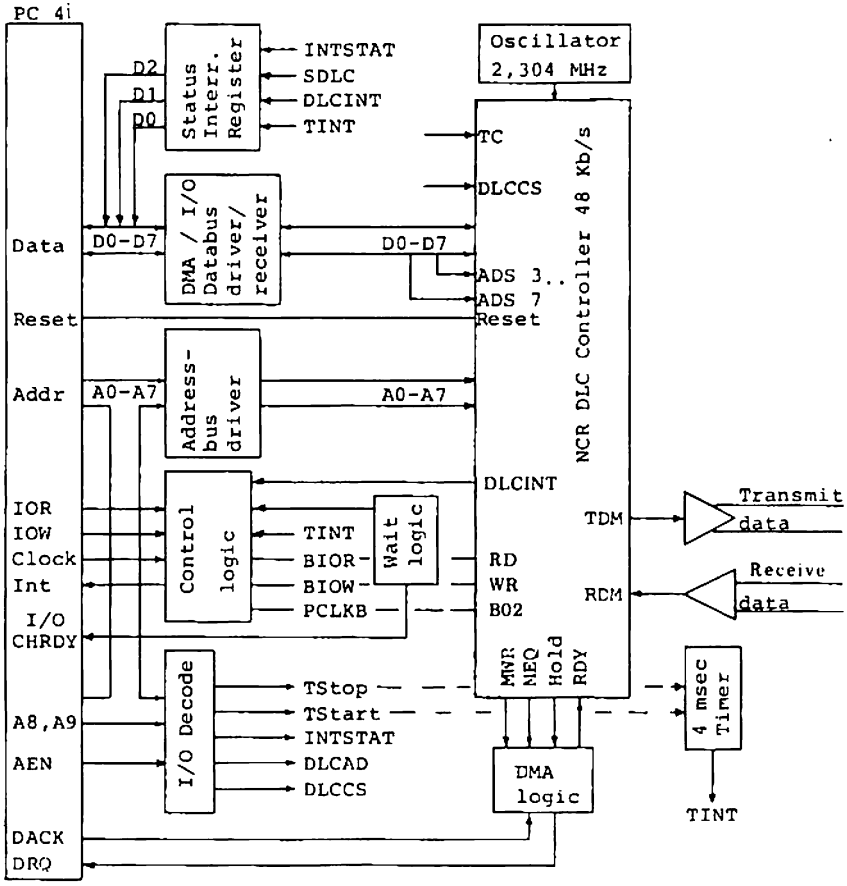


Figure 5.3 DLC Controller: Signal Flow

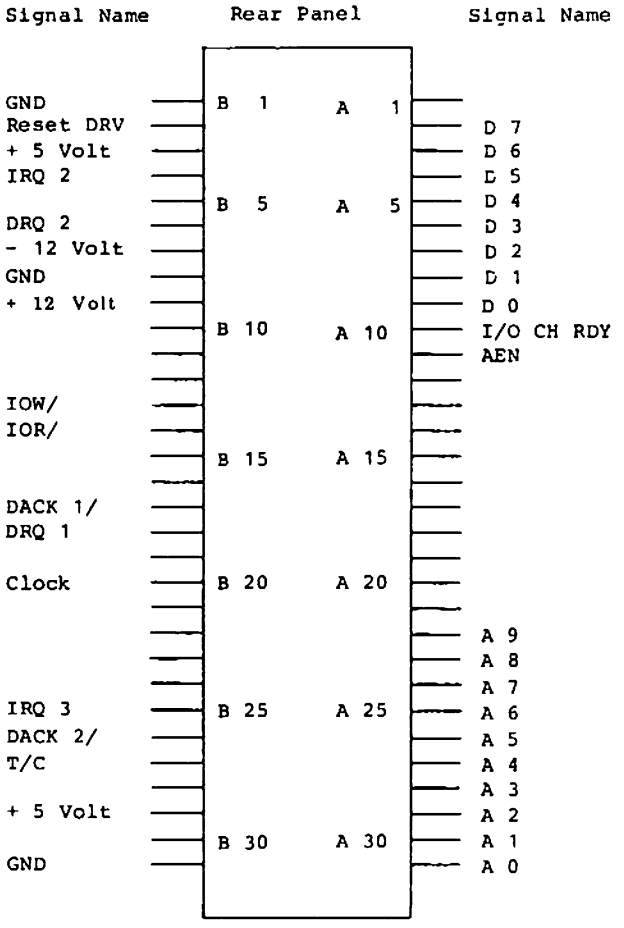


Figure 5.4 DLC Integrated Circuit

Data lines - Pins A2 ... A9
The system data bus (D0 to D7). The least significant data bit is represented at pin A9.

I/O CH RDY Output - Pin A10
I/O Channel Ready signal. This line is normally high (ready). It is pulled low by a memory or I/O device to lengthen I/O or memory cycles. Thus, slow devices can be attached to the interface with a minimum of difficulty. Any slow device should should drive this line low immediately upon detecting a valid address and a read/write command. This line is never held low for longer than 10 clock cycles.

IRQ2, IRQ3 Output - Pins B4, B25
These interrupt request lines are used to signal to the processor that the DLC controller requires attention. Jumpers are factory set to issue IRQ3 (Figure 5.2). The interrupt line is held high until it is acknowledged by the microprocessor.

IOR/ Input - Pin B14
I/O Read command. This signal instructs the DLC controller to place its data on the syatem data bus. It may be controlled by microprocessor or DMA.

IOW/ Input - Pin B13
I/O Write command. This signal instructs the DLC controller to read data from the system data bus. It may be controlled by microprocessor or DMA,

DRQ2, DRQ1 Output - Pins B6, B18
The DMA request lines. The selected DRQ line must remain high until acknowledged by the corresponding DACK/ line. The factory setting of the corresponding jumpers is that DRQ1 is selected (see Figure 5.2).

DACK1/, DACK2/ Input - Pins B17, B24
Acknowledge signal lines corresponding to the respective DMA request lines.

AEN Input - Pin A11
When this line is active, the microprocessor is degated so that a DMA transfer can take place.

T/C Input - Pin B27
Terminal Count. This line is pulsed active when the terminal count for the DMA channel has been attained.

The following voltages are present at the DLC controller:

+ 5 V 5%	- Pins B3, B29
+ 12 V 5%	- Pin B9
- 12 V 10%	- Pin B7
Ground	- Pins B1, B10, B31

SOFTWARE CONTROL
The interface address lines are decoded as follows:

IN 380H	Data Receive Hold Register
OUT 380H	Data Transmit Hold Register
IN 381H	Status Register
OUT 381H	Command Register
OUT 382H	Group Address Register
OUT 383H	Transmit DMA Address (low) Register
OUT 384H	DMA Counter Transmit Register
OUT 385H	Receive DMA Address (low) Register
OUT 386H	DMA Counter Receive Register
OUT 388H	Unique Address Register
IN 389H	Interrupt Status
OUT 38AH	Start Timer (4 ms)
OUT 38BH	Stop and Reset Timer

Command Register

This register controls the functions of the DLC integrated circuit. Each of the eight bits is dedicated to a different task. The structure of this Command Register byte is shown in Figure 5.5.

Bits:							
7	6	5	4	3	2	1	0
TXFR	TXFLG	RXA	RXB	IO/DMA	DIAG	SWRST	0

Figure 5.5 Command Register

TXFR

Transmit Frame. Setting this bit instructs the DLC controller to begin transmission of a frame. Subsequently data for transmission is expected from the system bus. In DMA mode the controller internally resets this bit when the transmission is complete. Under direct microprocessor control of data transfer this bit is reset when a Transmitter Not Serviced condition arises.

TXFLG

Setting this bit causes the controller to transmit contiguous flags. If TXFR is set simultaneously the transmitted frame is completed before the contiguous flags are transmitted.

RXA, RXB

The binary value contained in these two bits (RXA is MSB) determines which receive mode is active:

0 Receive mode not active
1 Accept all addresses (Primary)
2 Accept group, global, or unique address (Secondary)
3 Accept unique addresses only

These two bits are reset internally by the controller at the conclusion of frame reception. Global address is F9.
The transmit bits (TXFR, TXFLG) take priority over the receive bits. In the event of a conflict, all transmission is completed before receive mode is entered.

IO/DMA
As long as this bit is zero, data transfer between DLC controller and system (both transmit and receive) takes place under DMA control. Otherwise data is transferred under microprocessor I/O control.

DIAG
Setting this bit causes the integrated circuit to carry out internal diagnostics. In transmit diagnostics the output lines Ta and Tb are disabled. Transmitter output is then routed back into the Receive Hold Register. Receive diagnostics can also be performed.

SWRST
Setting this bit resets the DLC controller. This reset must be performed after every introduction of power to the integrated circuit.

Status Register
Figure 5.6 summarizes the bits of the Status Register.

Bit:							
7	6	5	4	3	2	1	0
EOF	RNS	TNS	FCS ERROR	RX ABORT	not used	DMATRM	SVCRQ

Figure 5.6 Status Register

EOF
End of Frame. This bit is set after a frame has been received whereupon the DLC controller generates an interrupt.

RNS
Receiver Not Serviced. This bit is received after a frame has been received if the Receive Hold Register was not read in time. This condition can occur in both DMA and processor I/O modes of operation.

TNS
Transmitter Not Serviced. The Transmit Hold Register was not written in the required time. This error causes the issue of an interrupt. The TNS condition can occur in both DMA and processor I/O modes of operation, but not under the following circumstances:

* The transmit command was aborted by controller command.
* The condition was recognized during the last byte transfer in a microprocessor I/O sequence.

FCS ERROR
Frame Check Sequence Error. After a complete frame has been received, the received frame's FCS bytes are compared with those generated by the DLC controller during data byte reception. Any discrepancy results in this status bit being set.

RX ABORT
If seven contiguous "ones" are detected on a received frame, this bit and the EOF bit are set and an interrupt is issued.

DMATRM
This bit is used for DMA transfers between DLC controller and the system. In the receive mode of operation DMATRM is set when the count written to the DMA Counter Receive Register is exhausted. In transmit mode it is set when the count in the DMA Counter Transmit Register is exhausted, or when a T/C (Terminal Count) signal is received from the DMA controller.

SVCRQ
Service Request. This bit is used for transfer under direct microprocessor control, that is, when the Transmit Hold Register is waiting for a byte of data (transmit mode) or when a byte of data is waiting to be read from the Receive Hold Register (receive mode). An interrupt is generated when this bit is set.

With the exception of SVCRQ reading the Status Register resets all its bits. SVCRQ is reset only after the data transfer via the bus has occurred.

Interrupt Status Register
Three bits of the Interrupt Status Register read via port 389H are significant:

Bit 0 A DLC controller interrupt sets this bit.
Bit 1 4 ms timer interrupt.
Bit 2 Reflects DRQ/IRQ jumper selection:
0 - DRQ2 and IRQ2
1 - DRQ1 and IRQ3

Unique Address Register
This register identifies the NCR PERSONAL COMPUTER in the DLC configuration. Normally, the computer will be functioning as the primary unit. The Unique Address Register is automatically set to 01 (primary) at DLC reset (which should always be performed after power introduction and before a new command), so that this register need not normally be written. If you have cause to use your NCR PERSONAL COMPUTER as a secondary device in a DLC configuration, you can write this register with a binary value as shown in Figure 5.7.

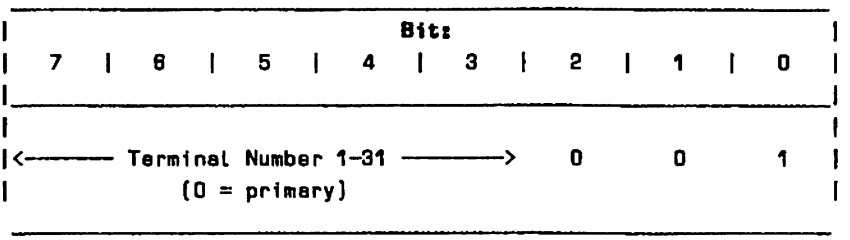


Figure 5.7 Unique Address Register

In addition the transmit and receive lines from the adapter exchange significance.

PROGRAMMING CONSIDERATIONS

With the aid of flow charts the remainder of this Chapter discusses some fundamental points to observe when designing software for the DLC interface.

Transmitting via CPU

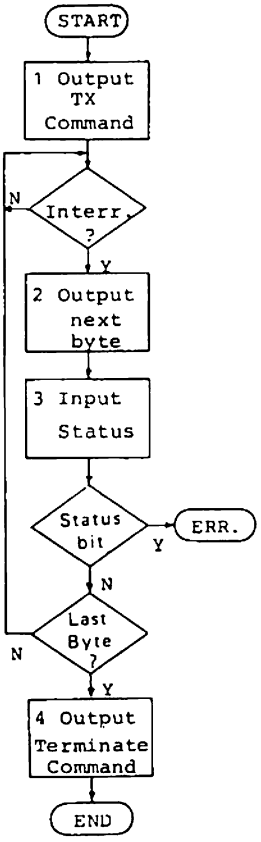


Figure 5.8 (1 of 2) CPU Transmit

1. The Command Register is written with bit 3 set, indicating that transfer will take place under direct CPU control. The values for bits 6 and 7 conform to the type of transmission required. When the Command Register has been written, the transmission sequence is started and an interrupt is generated.

2. Transmission circuits are enabled following servicing of the first interrupt. Whenever an interrupt occurs, it must be serviced (that is, data must be output) within six bit times, otherwise a Transmitter Not Serviced condition arises and the frame is terminated.

3. After the interrupt has been serviced, check the Status Register for possible errors. Normally the register should have all bits zero.

4. To terminate the frame in progress issue a transmission terminate command (Command Register bits: 0xxx1000), again within six bit times, otherwise a TNS error will

Figure 5.8 (1 of 2) CPU Transmit occur.

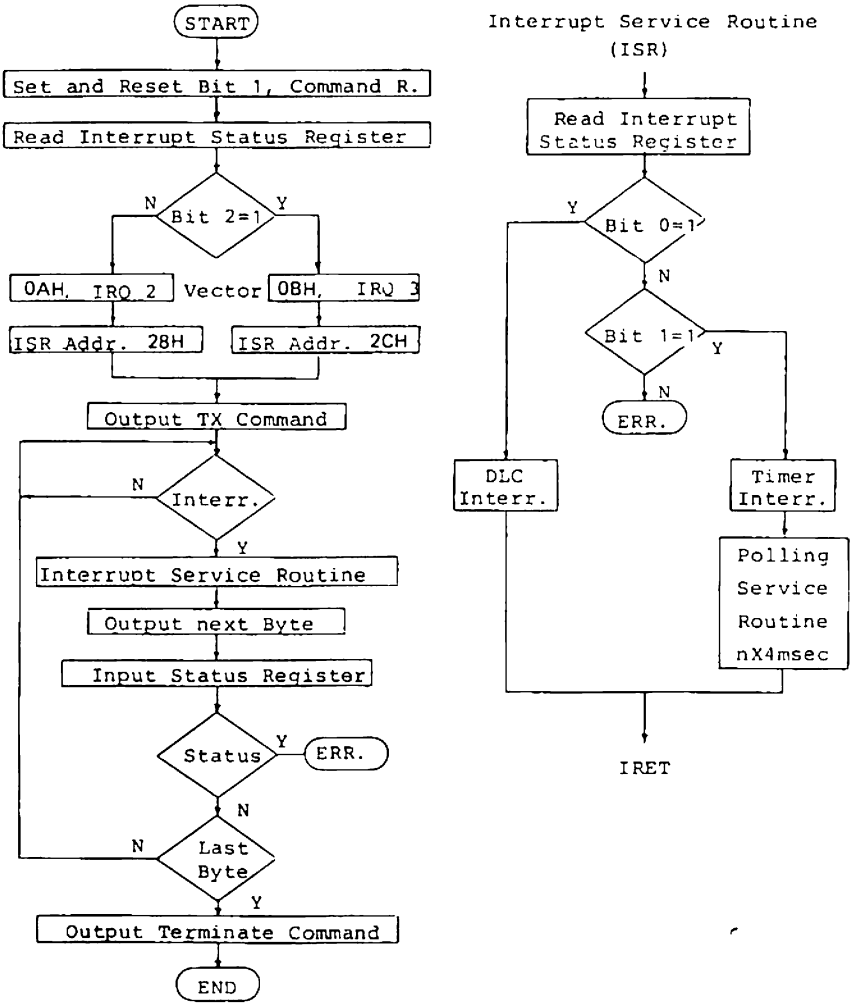


Figure 5.8 CPU Transmit (2 of 2)

Receiving via CPU

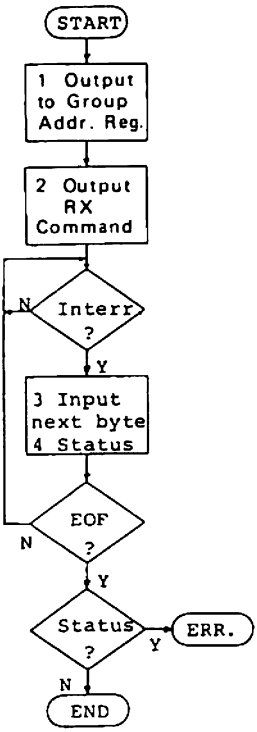


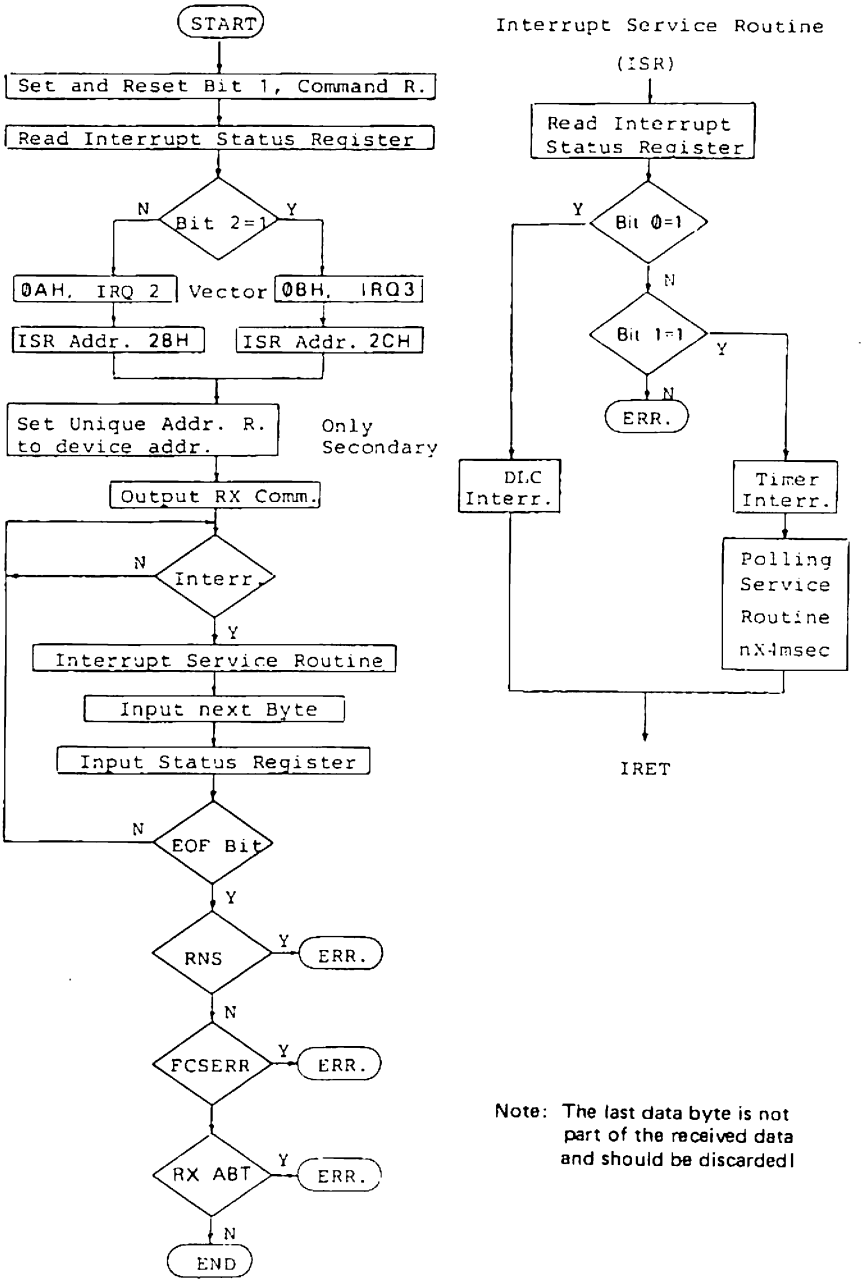
Figure 5.9 (1 of 2)
CPU Receive

1. Assuming that the DLC interface is the primary device, the Group Address Register need not be written. If a unique, group, or global command is to be issued (Command Register bits 4 and 5), this register must be written accordingly.

2. Issue a receive command detailed in bits 4 and 5 (RXA, RXB), with the IO/DMA bit set. A receive command may be given at the same time as a transmit command. In this case the latter is performed first.

3. The receive command starts by activating the receiver circuits. First a valid addressed frame is searched for. Upon detection an interrupt is issued, indicating that a data transfer to the system is required. This data byte must now be read by the CPU within 6 bit times after the interrupt, otherwise a Receiver Not Serviced condition will arise.

4. After the transfer of each data byte examine the Status Register. If the End of Frame (EOF) bit is set the data transfer is complete. If any other bit is set an error condition exists. If no bits are set there is still data to be received from the terminal.
   Note: the last byte of data received (the byte which caused EOF to be set) is extraneous and should be discarded.



Note: The last data byte is not part of the received data and should be discarded!

Figure 5.9 CPU Receive (2 of 2)

Transmitting via DMA

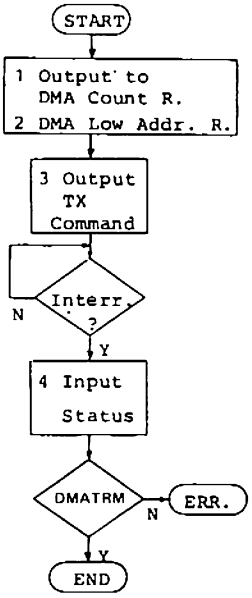


Figure 5.10 (1 of 2) DMA Transmit

1. Initialize the DMA Counter Transmit register to OFFH.

2. Write the 8 LSBs of the first memory location to the Transmit DMA Address Register. (The higher bits are to be supplied by the DMA controller).

3. Issue a transmit command in accordance with the type of interframe transmission required. If both TXFR and TXFLG are set (Command Register: 11xx0000) interframe time will be filled with contiguous flags. If TXFR is set and TXFLG is zero (Command Register: 10xx0000) the frame is followed by an idling sequence of 25 ones.

4. The DMA controller issues an interrupt after the last transfer. The Status Register should now be read. Two conditions are significant. The DMATRM bit set denotes successful completion of the DMA transfer. Any error results in a premature termination of the transfer and the setting of the Transmitter Not Serviced bit.

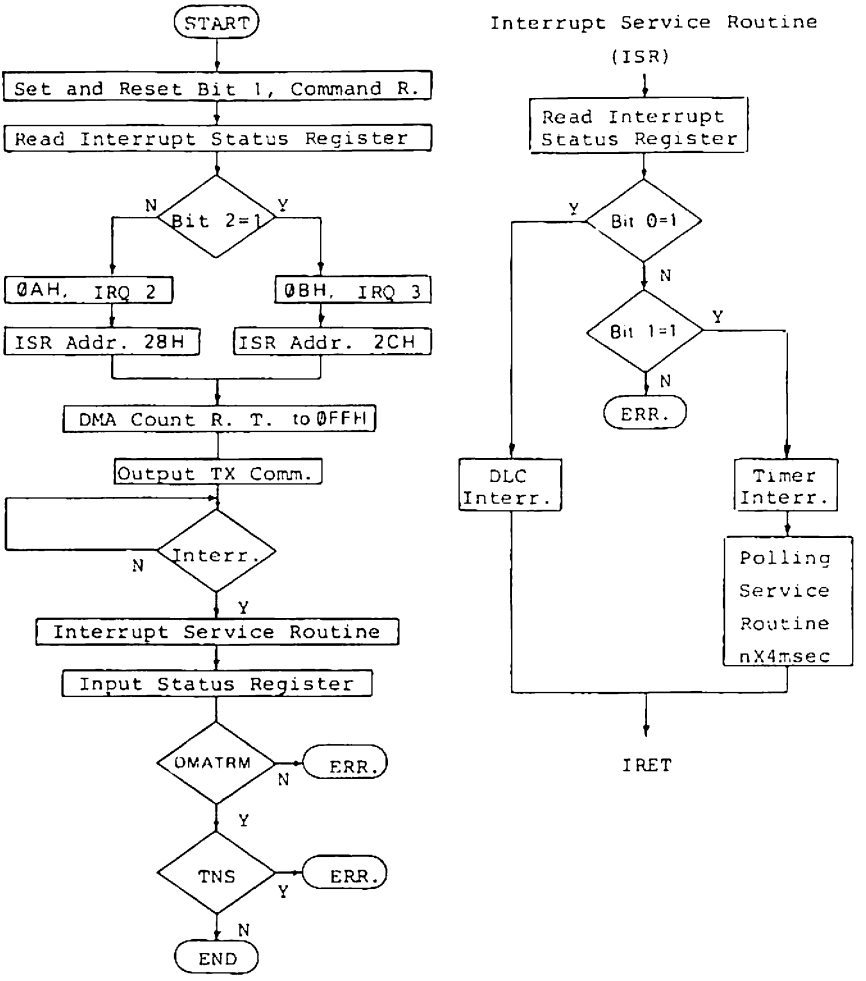


Figure 5.10 DMA Transmit (2 of 2)

Receiving via DMA

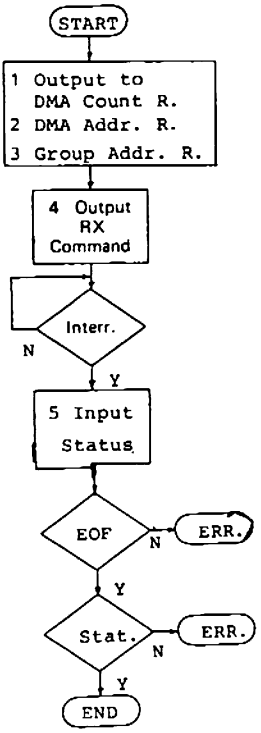


Figure 5.11 (1 of 2)
DMA Receive

1. Initialize the Receive DMA Count Register to OFFH.

2. Write the 8 LSBs of the first memory location to the Receive DMA Address Register (the higher bits are to be supplied by the DMA controller).

3. If the receive command is to be unique, group, or global, write the Group Address Register (4 MSBs only; LSBs must be all set).

4. Write the receive command to the Command Register, specifying RXA and RXB in accordance with the type of receive required and with the IO/DMA bit zero.

5. After a valid addressed frame has been detected data is transferred until all data has been received. Following the last byte an interrupt is generated indicating that a frame has been received. Reading the Status register should show that only the EOF bit and possibly the DMATRM bit are set. (DMATRM is set if the DMA count was exhausted before the end of frame was Figure 5.11 (1 of 2) detected). Any other bits set denote an DMA Receive error condition.

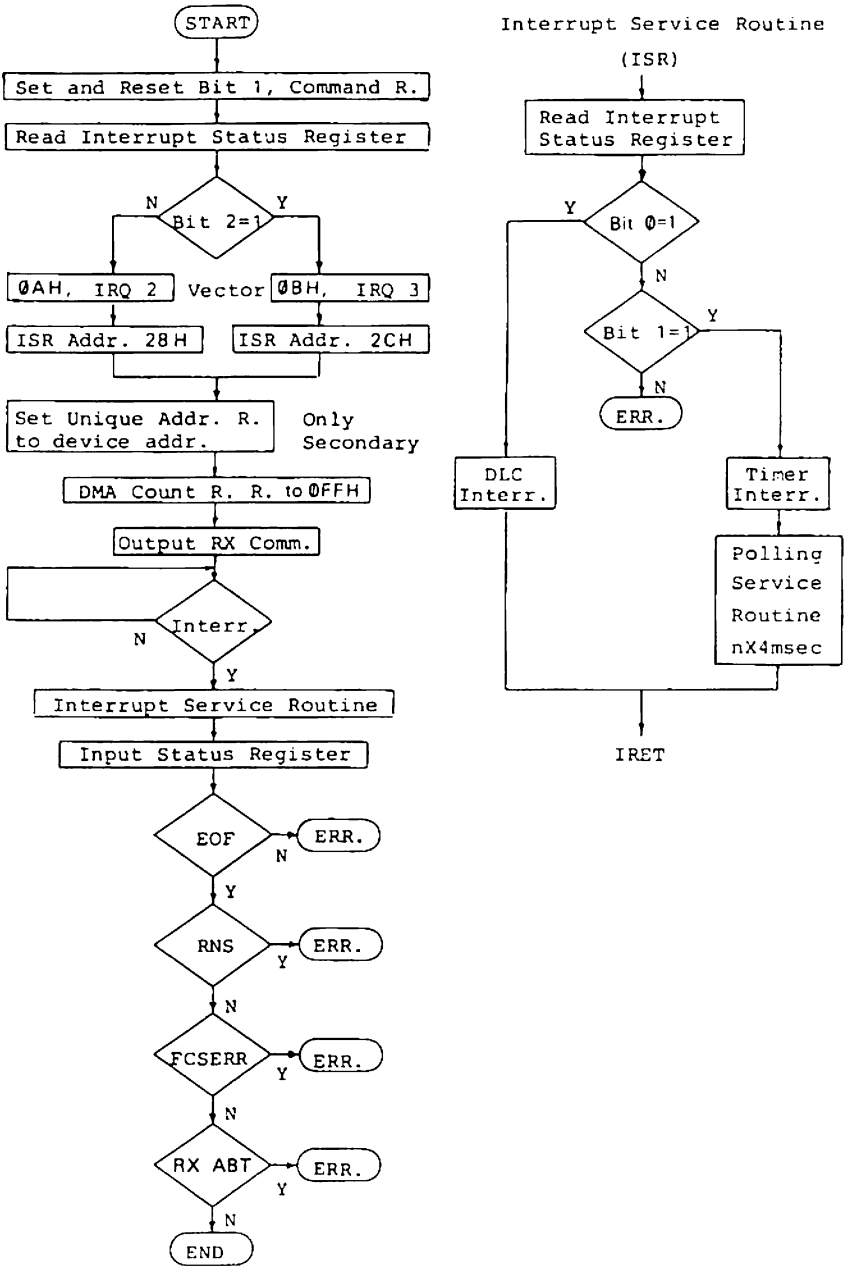


Figure 5.11 DMA Receive (2 of 2)