516-36
JCS
6/10/70

# NODE MODEM INTERFACE FOR COMPUTER TERMINALS

The following is a brief description of an interface constructed to facilitate the connecting of any device to a node on the PDP-516 communication ring.

The communication ring via the node is capable of supplying a 16 bit word (text) to a device or receiving a 16 bit word (text) from a device. The direction of information flow or the availability of information at either the PDP-516 computer (CPU) or the device end of the ring is indicated by an operational code. There are four OP code flags that can be set by the node to indicate to the device the type of text that was sent by the CPU. After the OP code flag is recognized by the device, it must then be reset by the device. There are also four interrupt flags that can be set by the device to request service with the CPU by causing a program interrupt. When the CPU recognizes the interrupt flag it will cause the interrupt handling program to service the device.

I. The block diagram, Figure 1, shows the interface circuitry used when receiving information from the CPU. There are two types of 16 bit words that can be sent to the device. The two types are, a 16 bit data word or a 16 bit command word. The four least significant bits reflect the mask register of the node. As shown in Fig. 1, a 16 bit holding register receives the data in two 8 bit bytes from the node using control lines from the node to jam-transfer at the proper times. After the data is settled in the register, one of two write flags (WCF, WDF) in the four bit OP code flag register is set. The device must determine which flag is set to make proper use of the 16 bit word received. The device should then issue a reset pulse to clear the flag register in anticipation of the next data word. The remaining two OP code flags (RDF, RSF) are used when transmitting data or status to the CPU from the device.

The four OP code flags stored in the 4 bit register can be connected to the device by two different methods as follows:

1. The OP code flags can be connected individually by four lines to the device. The device must have a means of individually recognizing each line when it is set.

2. The OP code flags are or'ed together in the interface and can be connected to the device by one line. The four flags are then multiplexed in the interface and are read in on the four least significant data lines to be decoded by the device for appropriate action.

II. The following discussion will show the method for transmitting information to the CPU from a computer type terminal. Referring to Fig. 2, there is a 16 bit data holding register into which the device parallel transfers its data word. After the data word is set into the register, the device should set the Device Read Interrupt flag (DRDI) in the 4 bit interrupt flag register. This causes the CPU program to interrupt, scan which device on the ring is seeking attention and what action is needed. The CPU interrupt program sends a read command to the device, picks up the data word in the holding register, resets the Device Read Interrupt flag (DRDI), and sets the Read Data Flag (RDF in Fig. 1) to indicate to the device that the data word has been taken. The device should reset the Read Data Flag (RDF) and repeat the above procedure if another data word is to be sent to the CPU.

The data word of text does not necessarily need to be 16 bits long. A word of shorter length can be loaded into the least significant bit positions of the register with the more significant bit position wired to ground (via the input cable) to transmit zeroes to the CPU.

III. The status register as shown in Figure 2 is a 12 bit holding register than can be loaded by the device. The six least significant bits of this register must contain the device code for the particular device. A list of these device codes are shown in Figure 3, others will be added as new devices are connected to the transmission ring. It is recommended that the six lines used for the device codes being transferred into the status register.

The six most significant bits of the status register can be assigned any appropriate status functions but must be discussed with the system programmers[1] for the proper device terminal signal process programs.

The sending of status to the CPU is handled in exactly the same manner as sending the data word. The Device Status Interrupt flag (DSTI) is used to seek the attention of the CPU and the Read Status Flag (RSF of Fig. 1) is set when the status has been accepted.

---

[1] C. Christensen and A. D. Hause.

IV. Cable Connections Between Device and Node Interface. Refer to Figure 4.

1. OTD1 thru OTD16; 16 output data lines to the device. Signal range is from ground to plus 5 volts. The polarity of these lines can be reversed by a patch arrangement on the interface board. These lines are normally driven by an open collector transistor (integrated circuit series SN7405) thereby requiring a pull-up resistor at the device end of the cable. The series SN7405 can be replaced with a series SN7404 integrated circuit eliminating the necessity for pull-up resistors but care must be taken to avoid reflections or line drop if the cable is very long.

2. DD1 thru DD16; 16 input data lines from the device to the 16 bit holding register of the interface. Signal levels must be as follows: ground = 0, +5 V. = 1.

3. DJAM; a positive going pulse (ground to +5 V.) to load the data DDL thru DD16 into the 16 bit holding register.

4. DNA1 thru DNA6, DST0 thru DST5; 12 status lines from the device to the 12 bit status register in the interface. The six lines DNA1 thru DNA6 must contain the device name code. The six lines DST0 thru DST5 are any appropriate status for the particular device. Signal levels are: ground = 0, +5V. = 1.

5. SJAM; a positive going pulse (gnd to +5V.) to load the 12 bit status holding register.

6. DWRI, DRDI, DAWI, DSTI; four lines that connect to the 4 bit interrupt flag register. Signal levels are: ground = 0, +5V = 1. These signals are parallel loaded into the interrupt register by the pulse IJAM as described below in #7.

   A. DWRI is the mnemonic for Device Write Interrupt. This is used when the Device Input Register is empty.

   B. DRDI is the mnemonic for Device Read Interrupt. This is used to tell the interrupt program that the device output register has data in it.

C. DAW is the mnemonic for Device Awake Interrupt. This interrupt causes the device program to Awake.

D. DSI is the mnemonic for Device Status Interrupt. This interrupt indicates a change of status.

All of the above flags are reset by the node upon receiving the answering command from the CPU except for the alarm call.

7. IJAM; a positive going pulse (gnd to +5V.) to load the interrupt flags into the 4 bit interrupt register.

8. WCF, WDF, RDF, BSF; four lines that connect from the interface to the device that can be used to determine the I/O instruction issued by the CPU for the device.

A. WCF is the mnemonic for Write Command Flag. When this flag is set it indicates to the device that a 16 bit word has been transmitted from the CPU to the data holding register in the interface, and the lines OTD1 thru OTD16 (see Fig. 1) are ready to be read by the device. It also tells the device the 16 bit word is a command word, whose individual bits can be decoded and used by the device.

B. WDF is the mnemonic for Write Data Flag. When this flag is set it indicates to the device a 16 bit data word has been sent to the data holding register (see Fig. 1) in the interface and the lines OTD1 thru OTD16 are ready to be read by the device.

C. RDF is the mnemonic for Read Data Flag. When this flag is set it indicates to the device the 16 bit data word setting in the data holding register (see Fig. 2) has been picked up and is on its way to the CPU.

D. RSF is the mnemonic for Read Status Flag. When this flag is set it indicates to the device the 12 bit status word setting in the status register (see Fig. 2) has been picked up and is on its way to the CPU.

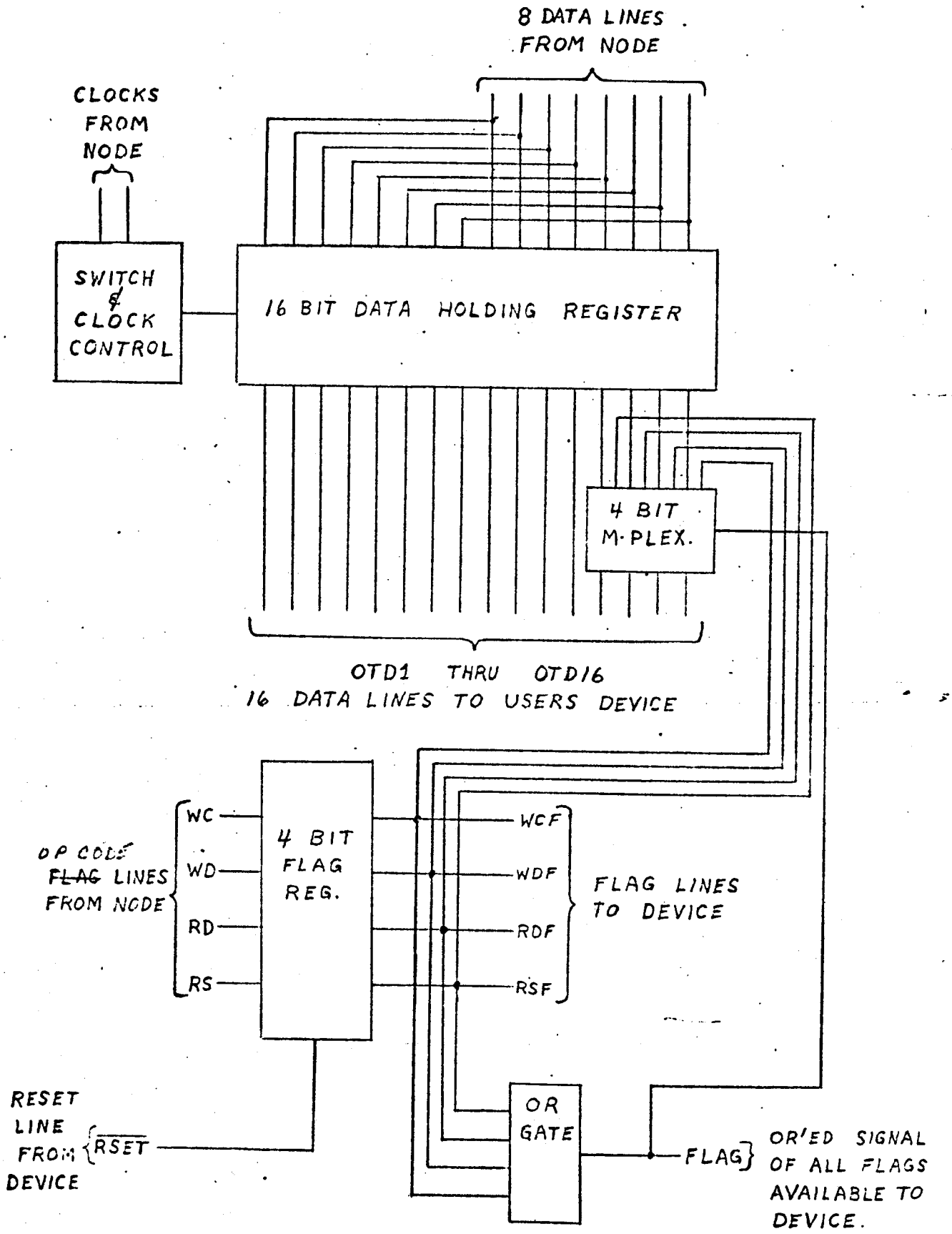
The signal levels for all of the above flags is ground and +5Volts. A patch arrangement is provided on the interface board letting the user choose which level shall indicate the set or reset condition. These lines are normally driven by open collector transistors (integrated circuit series SN7405) requiring pull-up resistor at the device end of the cable. See cable connection #1 OTD1 thru OTD16 for option regarding use of series SN7404 integrated circuit.

9. FLAG; One line to the device which is the or'ed result of all of the flags discussed in #8 above. It is set when at +5 Volts and reset when ground. This flag may be useful if the device has only one detection circuit for a flag change. If this flag is used, a patch arrangement on the interface board allows the four flags to be multiplexed onto the four least significant data lines (OTD13 thru OTD16; see Fig. 1) to the device.

10. RSET; A line from the device to the interface to reset the flags discussed in #8 and #9 above. This line must be held positive (+5V.) and pulsed to ground to reset the flags.

11. VCC; +5 Volts must be supplied to the node and the node interface from the device when the device is active. The device must be capable of supplying approximately 2 amps.

12. GND.; Ground return for +5 Volts power supply.

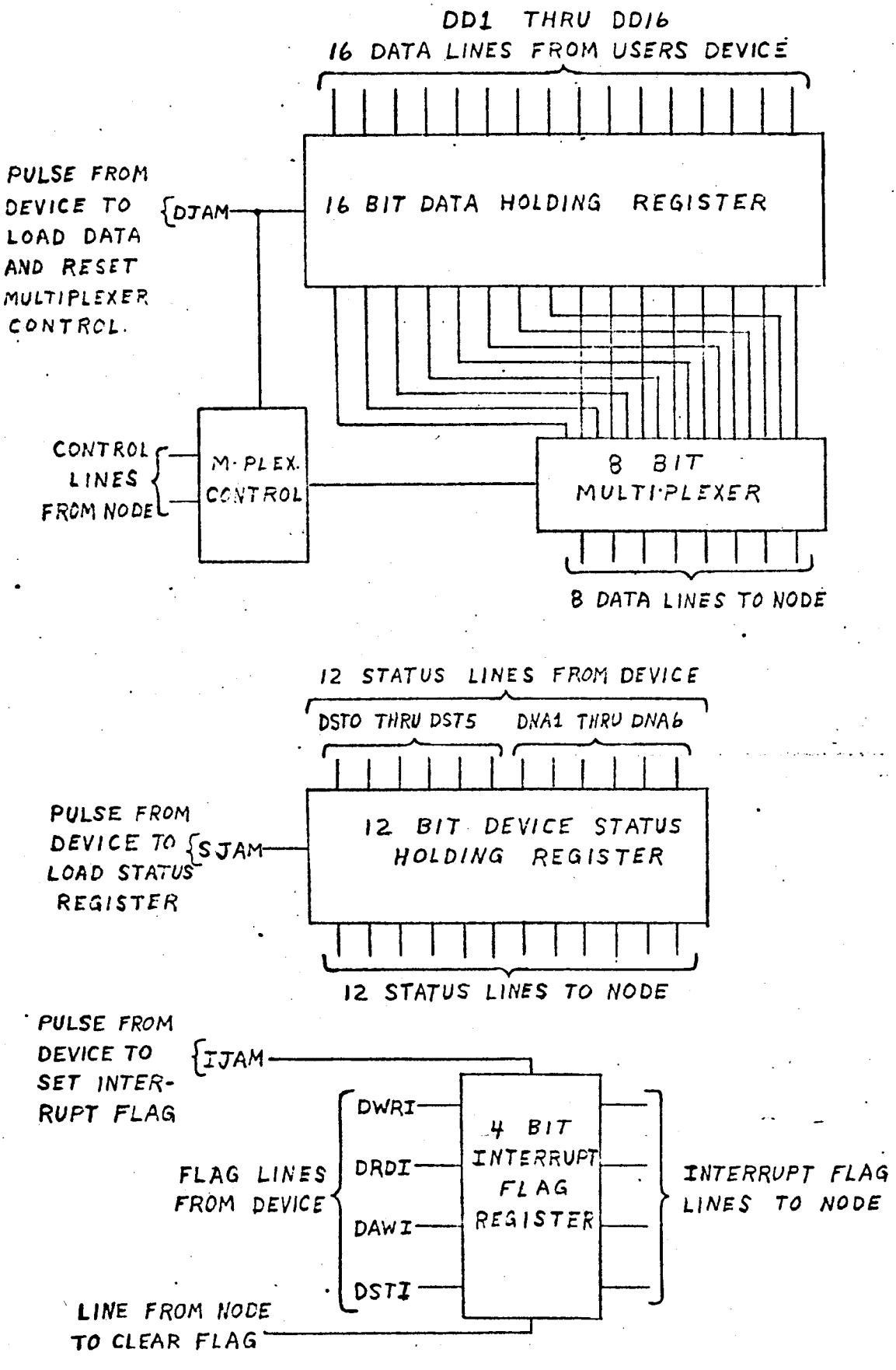



NODE MODEM TO USERS DEVICE INTERFACE

BLOCK DIAGRAM

FIG. 1

5/11/70




NODE MODEM TO USERS DEVICE INTERFACE

BLOCK DIAGRAM

FIG. 2

# I/O RING DEVICE CODES

| Device Code | Device |
|---|---|
| 00 | Mode is not connected to any device |
| 01 | 103A (15 char/sec) Dataphone |
| 02 | 103A (10 char/sec) Dataphone |
| 03 | PDP-8 D.E.C. computer |
| 04 | SOROBAN (SCCR) Card Reader |
| 05 | Graphical Terminal Glance 2A |
| 06 | Graphical Terminal Glance 2B |
| 07 | Graphical Terminal Glance X |
| 10 | Output Generator & Tester for ROM's and RAM's |
| 11 | |
| 12 | |
| 13 | |
| 14 | |
| 15 | |
| 16 | |
| 17 | |
| 20 | |
| 21 | |
| 22 | |
| 23 | |
| 24 | |
| 25 | |
| 26 | |
| 27 | |
| 30 | |

FIG. 3

CABLE CONNECTIONS BETWEEN
NODE INTERFACE AND DEVICE

| Row A | | Row B | | WD | WC |
|---|---|---|---|---|---|
| 1 | VCC | 1 | VCC | | |
| 2 | DST5 | 2 | OTD1 | Data MS | Control MS |
| 3 | DST4 | 3 | OTD2 | | |
| 4 | DNA6 | 4 | OTD3 | | |
| 5 | DNA5 | 5 | OTD4 | | |
| 6 | DNA4 | 6 | OTD5 | | |
| 7 | DNA3 | 7 | OTD6 | | |
| 8 | DNA2 | 8 | OTD7 | | |
| 9 | DNA1 | 9 | OTD8 | | |
| 10 | DST3 | 10 | OTD9 | | |
| 11 | DST2 | 11 | OTD10 | | |
| 12 | DST1 | 12 | OTD11 | | |
| 13 | DSTO | 13 | OTD12 | | Control LS |
| 14 | FLAG | 14 | OTD13 | | SIM (node mask register bits) |
| 15 | WCF | 15 | OTD14 | | RIM (node mask register bits) |
| 16 | RDF | 16 | OTD15 | | WIM (node mask register bits) |
| 17 | WDF | 17 | OTD16 | Data LS | CRdy (node mask register bits) |
| 18 | RSF | 18 | $\overline{\text{RSET}}$ | | |
| 19 | SJAM | 19 | IJAM | | |
| 20 | DJAM | 20 | | | |
| 21 | DD1 | 21 | DD2 | | |
| 22 | DD3 | 22 | DD4 | | |
| 23 | DD5 | 23 | DD6 | | |
| 24 | DD7 | 24 | DD8 | | |
| 25 | DD9 | 25 | DD10 | | |
| 26 | DD11 | 26 | DD12 | | |
| 27 | DD13 | 27 | DD14 | | |
| 28 | DD15 | 28 | DD16 | | |
| 29 | DWRI | 29 | DAWI | | |
| 30 | DRDI | 30 | DSTI | | |
| 31 | GND | 31 | GND | | |

FIG. 4