PHILIPS

Electronic components and materials

Technical note 072

Introducing the Signetics 2651 PCI

Terminology, formats and operation modes

Data exchange between microprocessors and peripherals is normally performed in the parallel mode, using the data bus. This means that the data is transferred along a number of parallel connections, all bits of the data word being transferred simultaneously. However, when a peripheral is remote from the microcomputer system, it is usually more economical and sometimes obligatory to communicate via a single data line. This means that the parallel data within the microcomputer must be converted into a serial form before transmission and vice versa.

The conversion between serial and parallel data can be done either by hardware or software, but the hardware solution must usually be used as the software conversion puts too great a load on the microprocessor. The Signetics 2651 Programmable Communications Interface (PCI) has been developed to perform this task of parallel/serial conversion: it is a single chip providing the complete hardware for virtually any mode of serial data communication. **Figure 1** shows two typical applications of serial data interfaces to microcomputers. Note that for communication over a telephone line, a modulator/demodulator (modem) is required.

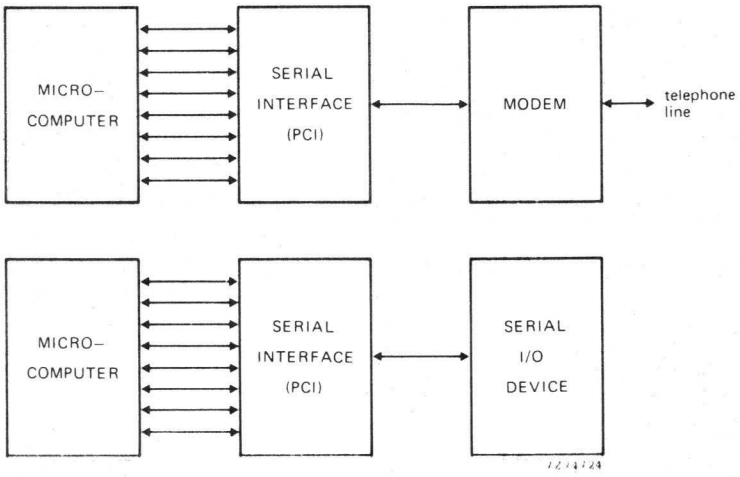


Fig. 1 Typical applications of serial data interfaces to microcomputers.

signetics

Serial data communication formats

Figure 2 shows the serial bit stream equivalent to three eight-bit data words. Whereas parallel data bits can be recognized at a receiver by their separate connections, serial data bits can only by distinguished by their separation in time. The receiver must therefore be supplied with *timing information. Framing information* is necessary to be able to recover the original data words from the serial bit stream

Timing information enables the receiver to distinguish between consecutive bits of the serial data stream, while framing information enables the receiver to recognize the start and finish of each data word.

Figure 3 shows the use of timing information to recover the original data. Each bit of the serial data stream must be transmitted for a fixed duration, called the *unit interval.*

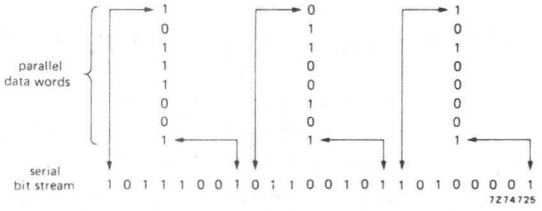


Fig. 2 Relation between the parallel data words and the serial bit stream transmitted or received.

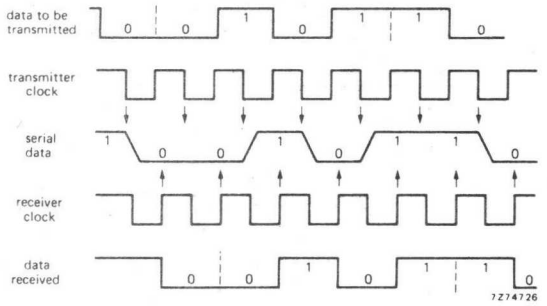


Fig. 3 Use of a receiver clock to recover data from the serial bit stream.

The receiver clock must be synchronized to the frequency of the transmitter clock, with a fixed phase delay, to allow sampling of the serial data waveform at the same time in each unit interval. The maximum rate at which information can be sent over the data line is known as the *baud rate*; it is equal to the number of unit intervals per second. Thus for a unit interval of 20 ms, the baud rate is 50 *baud*. Commonly used baud rates range from 50 baud up to 19,2 kbaud. A standard teletype uses 110 baud.

To recover the original data, the timing information must contain:

- the baud rate;
- bit synchronization information.

To reform the original data words, the framing information must contain:

- identification of the first bit of a data word;
- the number of bits per word;
- the sequence in which the bits are sent (msb or lsb first).

Of these, the baud rate, the number of bits per word and the sequence in which the bits are sent, are usually fixed and already known by the receiver before the transmission of data. Thus the data signal must contain the bit synchronization and first-bit identification information. Several serial data communication formats have been devised for this purpose. The two basic formats are *synchronous* and *asynchronous,* while a mixture of the two is called *isochronous.*

Asynchronous format

When using the asynchronous format, the transmitter transmits each word separately. Each word is preceded by one start bit and followed by a parity bit and 1, 1½ or 2 stop bits. This format is illustrated in Fig. 4.

When the data line is quiet, the signal is a one. The start bit is a zero, which tells the receiver that a data word is coming. Since the start bit can occur at any moment, synchronization of the receiver clock to the data signal must be repeated for each data word. Therefore, the receiver clock runs at a multiple (usually 16x or 64x) of the actual baud rate. Figure 5 shows the synchronization principle with a clock running at 16 times the baud rate. The receiver clock is derived from this 16x clock by means of a divide-by-16 circuit, which starts at the moment a start bit is detected (falling edge of the previously quiet line).

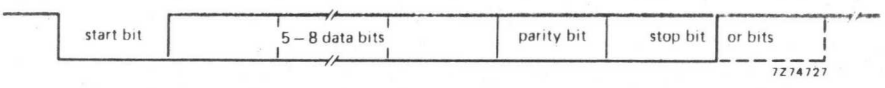


Fig. 4 The asynchronous serial data format.

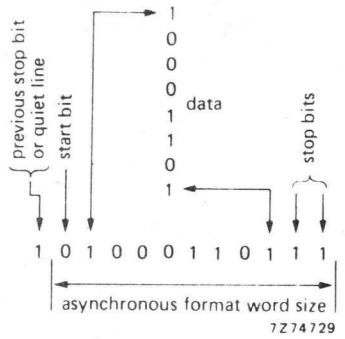


Fig. 6 Example of an asynchronous data format using eight data bits and two stop bits.

After eight pulses of the 16x clock, the data line is sampled again and if the line is still a zero, the start bit is accepted. After eight more pulses, the first data bit appears on the line and is sensed 16 pulses after the start bit was accepted. This means that the bit is sensed at the middle of the bit--time to avoid switching transients. Each following group of 16 clock pulses will represent a unit interval, after which a new bit is sensed.

The maximum inaccuracy resulting from this method of synchronization is initially one sixteenth (in the case of a 64x clock, 1/64th) of a bit-time. The accuracy of synchronization of the following data bits depends on the equality of the transmitter and receiver clock frequencies. However, since the next data word will provide a new start bit, this synchronization only has to last for the duration of one data word.

From the foregoing description, it is clear that all the bits transmitted, including the start bit, must be exactly one unit interval long, and that the time between two consecutive data words need not be an integral number of unit intervals. To ensure correct detection of the next start bit (data word) the line must be forced to the quiet state at the end of the data bits of every word. This is done by adding the stop bit or bits onto the end of the word, which, being logical one, provide the quiet state. Thus the receiver will always see a clear falling edge at the beginning of the start bit of the next data word, even if the receiver clock runs slightly slower than the transmitter clock. Figure 6 shows an example of an asynchronous data format with eight data bits and two stop bits.

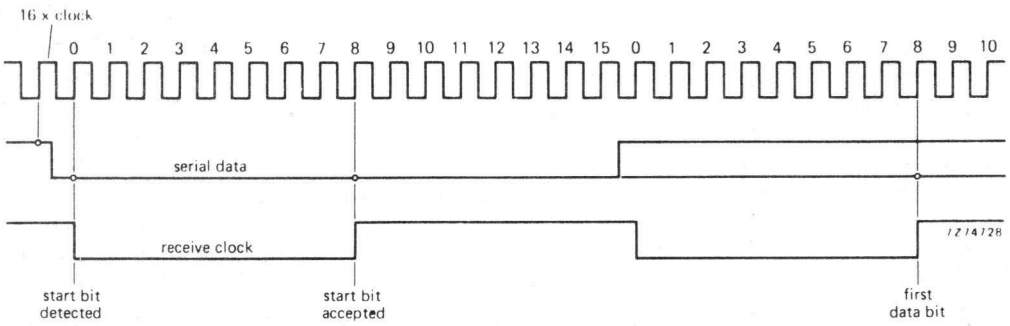


Fig. 5 Synchronization principle for the asynchronous format, here using 16x clock.

Synchronous format

In the synchronous format, data words are grouped into blocks before transmission. This provides a continuous stream of valid data bits, one per unit interval. Once the transmitter and receiver clocks are synchronized, the receiver looks for framing information. This is done in the *hunt mode* in which it continually checks the received bit sequence for synchronization characters.

Each block is preceded by one or more synchronization words, of a fixed character, see Fig. 7. The sync words are called SYN (in the case of one sync word) or SYN1 and SYN2 in the case of two sync words. The bit patterns corresponding to these characters must not occur in the data to be transmitted.

When a bit sequence conforming to the sync character(s) is recognized, the receiver switches to the data mode: the first bit following the sync character(s) is the first bit of the first data word. Figure 8 shows the bit sequence of a data block in synchronous format with two sync characters and five bits per word.

If, during transmission of a data block, the microprocessor fails to supply a new data word for transmission, the transmitter automatically inserts sync characters (SYN or SYN1-SYN2 pairs as appropriate) to prevent a gap occurring. Sync characters are inserted until a new data word is available. These sync characters can be automatically discarded by the receiver, while providing both framing and timing synchronization.

Since synchronization of the receiver and transmitter clocks must be maintained over a long stream of data, the timing information is usually continuously extracted from the data signal. Sometimes, however, a synchronization signal is sent to the receiver separately.

Isochronous format

In order to recover the data words from the serial data stream, the isochronous method employs the framing of the asynchronous format (start and stop bits) and the timing of the synchronous method (clock frequency equal to the baud rate). The asynchronous framing permits gaps in the data, although these must now be an integral number of unit intervals so that the timing can remain synchronized.

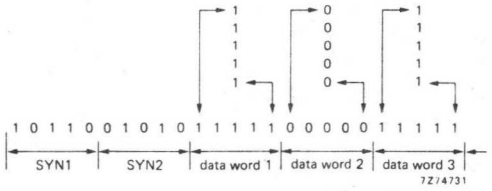


Fig. 8 Synchronization at the start of a data block in the synchronous format. Double sync operation with five bits per data word.

Modem control

Telephone lines are designed for speech transmission and will only transmit analogue signals in the range 300 Hz to 3400 Hz. Digital signals cannot be transmitted. Thus when transmitting digital data over telephone lines, a modem (modulator-demodulator) is required to provide the conversion to and from analogue signals. In order to regenerate the modulated data, a synchronous modem must be able to generate a receiver clock for the conversion of the serial data into parallel data. Figure 9 shows the use of a modem to interface digital signals to a telephone line. As can be seen from the diagram, several control lines are required between the serial interface and the modem. These are:

- TxC: *Transmitter Clock*, timing information for synchronous modem;
- TxD: *Transmit Data* in serial form to the modem;
- RTS: *Request to Send*, request to the modem to prepare for the transmission;
- DTR: *Data Terminal Ready*, request to the modem to establish the connection to the telephone line and enter the data mode (as opposed to the dial and talk modes).
- RxC: *Receiver Clock*, for synchronous modem;
- RxD: *Receive Data*, from the modem to the serial interface;
- DSR: *Data Set Ready*, signals that the modem is connected to the telephone line and in the data mode;
- CTS: *Clear to Send*, signals that the modem is ready to accept serial data for transmission;
- DCD: *Data Carrier Detect*, signals that the modem is receiving a signal suitable for demodulation.

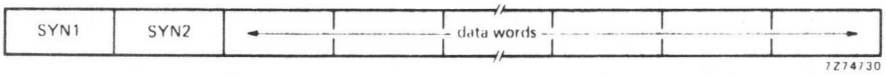


Fig. 7 The synchronous serial data format.

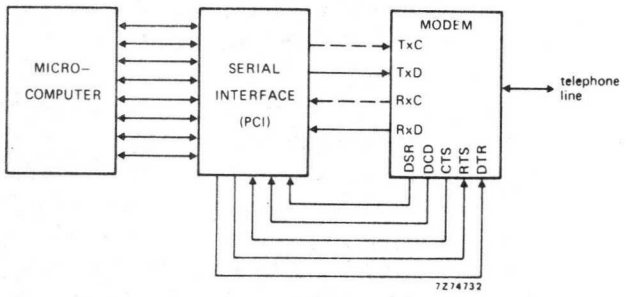


Fig. 9 Transmitting digital data over a telephone line by using a modem, TxC and RxC are required if a synchronous modem is used.

Transparent operation – synchronous mode

Data communication systems commonly employ a seven-bit, 128-character code, known as seven-bit ISO code, see Appendix A. This includes alphanumerics, general purpose control characters and ten transmission control characters (TC_n). *Communication control procedures* have been established to provide rules for the use of these transmission control characters.

In some cases, when it is required to obtain the maximum throughput of the communication network, data may be transmitted without using ISO code, in a network that normally uses ISO code. In this case, data words with the same bit pattern as the transmission control characters will be misinterpreted by the system as transmission control characters.

To overcome this problem, the communication control procedures have been extended to specify *transparent operation,* to allow *code-independent* data transfer. This operation is governed by the transmission control character DLE (Data Link Escape).

The start of code-independent data is preceded by two transmission control characters DLE and STX (Start of Text). The end of the sequence is indicated by the characters DLE and ETB (End of Transmission Block) or DLE and ETX (End of Text). Between these start and finish character sequences, all bit patterns except DLE are treated as data. Control characters can still be used for control purposes if preceded by the DLE character. Sync characters used to fill gaps in the synchronous format should also be preceded by DLE to prevent interpretation of these characters as data. The exception is the case where the data word has the same bit pattern as DLE plus parity bit. This is transmitted as DLE DLE and the second DLE is interpreted as data, not control. Table 1 shows the interpretation of various character sequences.

Table 1 Transparent operation using the DLE character.

sequence transmitted and received	interpretation	
	data	control
DLE ETB		ETB
DLE DLE	DLE	
X ETB	X ETB	
X ETX	X ETX	
X SYN	X SYN	
DLE SYN		gap filler
DLE STX		STX
DLE DLE DLE ETB	DLE	ETB
DLE DLE DLE DLE	DLE DLE	
DLE DLE ETB	DLE ETB	

Note: X is any character other than DLE.

Error detection

Parity check

Regardless of the transmission medium, the data signal entering the receiver will contain noise. At times, this noise content may be sufficient to cause incorrect detection of data in the receiver.

The parity check provides a simple method of detecting a single error in a data word; a parity bit is added to each data word so that the total number of ones in each word is always even, or always odd. This is called *even parity* or *odd parity.* The receiver can then detect an odd number of incorrect bits in the data word. An even number of errors will not be detected.

Framing error

In the asynchronous mode, the receiver may detect the wrong number of stop bits, called a framing error.

Overrun error

The microprocessor should read any received data words before the following data word is received. If this is not done, the contents of the receiver register are overwritten and the earlier data word lost. This is called an overrun error.

The Signetics 2651 PCI

The Signetics 2651 Programmable Communications Interface is a universal synchronous/asynchronous data communications controller chip. Although designed specifically for use with the Signetics 2650 microprocessor, the PCI can easily be used with other CPUs in either polled or interrupt-driven environments.

The 2651 accepts programmed instructions from the microprocessor and supports many serial data communication disciplines in half or full duplex. The PCI supports isochronous, asynchronous and synchronous operations, including transparent synchronous operation.

The PCI converts the parallel data received from the microprocessor into a serial data stream, ready for transmission. At the same time, because of the independent transmit and receive circuitry, it can receive a serial data stream and convert this into parallel data for the microprocessor.

The block diagram of the PCI is shown in Fig. 10. All data and control transfers between the PCI and the microprocessor are accomplished via the data bus buffer, connecting the internal data bus to that of the microprocessor.

Overall control of the PCI is by the Chip Enable ($\overline{CE}$) signal. Address lines A0 and A1 allow selection of the required registers, while the Read/Write ($\overline{R}/W$) signal controls the direction of data flow between the PCI and the microprocessor.

The operation of the PCI is determined by the contents of the two mode registers and the command register. The registers are loaded via the data bus during system initialization.

The data in mode register 1 controls:

- the format:
  - synchronous
  - isochronous (asynchronous with 1x clock)
  - asynchronous (with 16x clock)
  - asynchronous (with 64x clock);
- the data word length:
  - 5, 6, 7 or 8 bits per word;
- the error checking:
  - no checking
  - odd parity
  - even parity;
- the number of stop bits for isochronous or asynchronous operation:
  - 1, 1½ or 2 bits;
- the use of SYN and DLE characters:
  - single sync operation (SYN1)
  - double sync operation (SYN1 + SYN2)
  - transparent operation (DLE + SYN).

The data in mode register 2 controls clock selection and the baud rate when internal clocks are selected:

- receiver clock:
  - internal or external;
- transmitter clock:
  - internal or external;
- baud rate:
  - 16 commonly used baud rates in the range 50 baud to 19 200 baud.

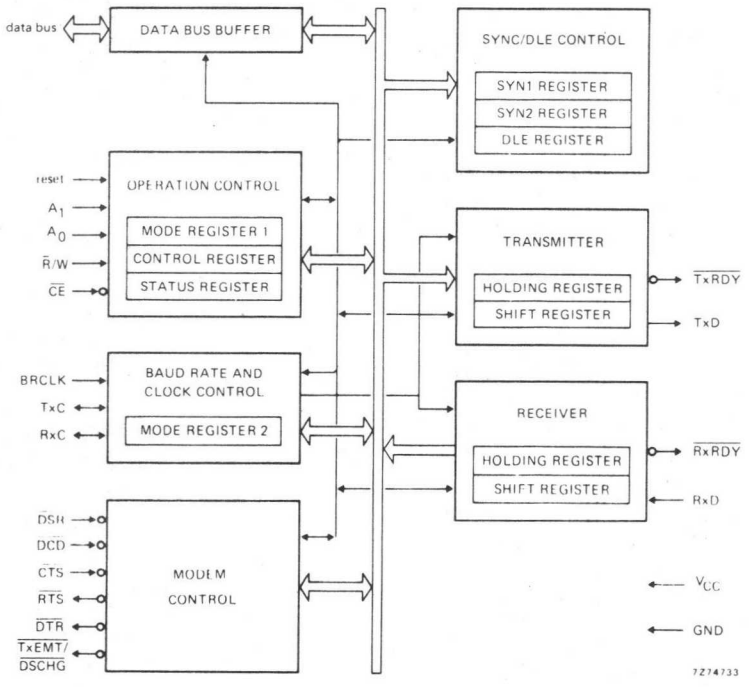


Fig. 10 Block diagram of the 2651 PCI.

The internal baud rate is derived from an externally generated 5,0688 MHz clock. When both receiver and transmitter clocks are programmed as external, the 5,0688 MHz clock is not required. If an internal clock is programmed, the programmed baud rate clock appears at the corresponding pin (TxC: transmitter clock; RxC: receiver clock). If an external clock is programmed, these pins become inputs for the external clock(s). With an external clock (max. 0,8 MHz) the baud rate is no longer governed by mode register 2. Thus the baud rates can be:

- synchronous and isochronous operation:
  0 to 800 kbaud;
  asynchronous operation with 16x clock:
  0 to 50 kbaud;
- asynchronous operation with 64x clock:
  0 to 12,5 kbaud.

The data in the command register controls:

- the transmitter:
  enable or disable;
- the receiver:
  enable or disable;
- the modem control signals DTR and RTS;
- resetting error flags in the status register;
- forced breaks in asynchronous format:
  The quiet line signal is a one. Break is a continuous zero sent by the transmitter which can be detected by the receiver.
- sending DLE in the synchronous format;
- the operation mode:
  - normal operation (Fig. 11a)
  - automatic echo mode, asynchronous (Fig. 11b). In this mode, the receiver controls the transmitter, causing it to transmit an echo of the received data to the external device.
  - SYN/DLE stripping mode, synchronous.
    SYN and DLE characters received by the PCI are not passed to the microprocessor.
  - local loop-back mode (Fig. 11c). The transmitter output is connected to the receiver input internally to provide a test facility or the CPU-PCI system.
  - remote loop-back mode (Fig. 11d). The receiver output is internally connected to the transmitter input to divert the received data to another computer.

The status register provides the microprocessor with information about the transmitter, the receiver, the modem control signals DSR and DCD, parity, overrun and framing errors and the detection of SYN/DLE characters.

The transmitter and receiver are double buffered, allowing the microprocessor to read/write one data word while another is being received/transmitted. This allows the microprocessor one complete serial word transmit/receive time to read/write the data word

When using synchronous formats, the SYN and DLE characters are supplied by the microprocessor during the initialization phase of the program and stored in PCI registers.

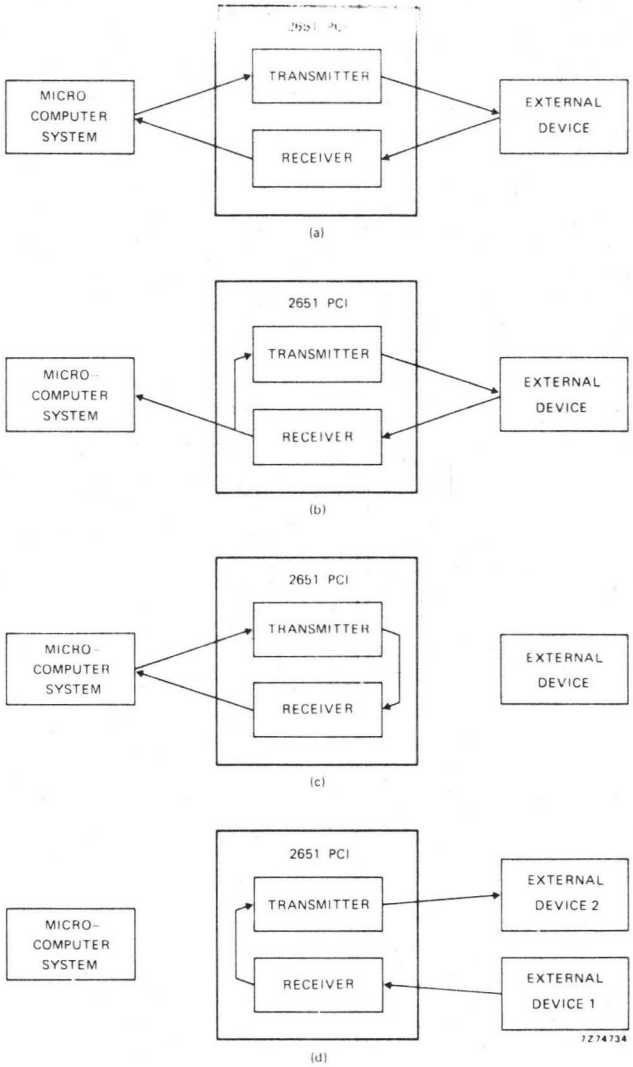


Fig. 11 Operation modes of the 2651 PCI.
(a) Normal operation mode.
(b) Automatic echo mode.
(c) Local loop-back mode.
(d) Remote loop-back mode.

Applications of the 2651 PCI

The 2651 PCI is suitable for almost any application where parallel source data must be transmitted over a single data line. Figure 12 shows two typical examples.

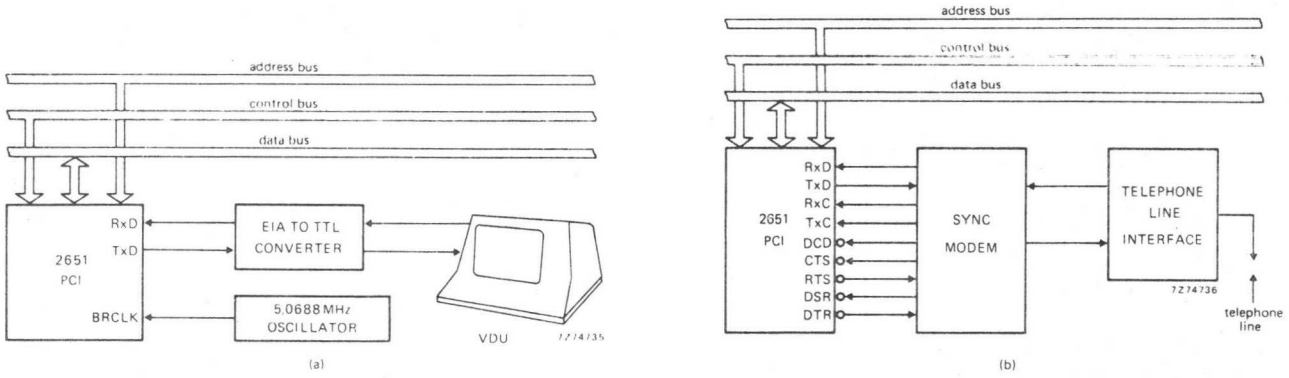


Fig. 12 Typical application for the 2651 PCI.
(a) Asynchronous interface to visual display unit.
(b) Synchronous interface to telephone line.

Appendix A

The international standard (ISO) 7-bit code as defined in Appendix B ref. 4:

7	6	5	4	3	2	1

7-bit character format.

column (0-7) (MS character)

row (1-15) (LS character)

CHARACTER SET								
M.S. CHAR / L.S. CHAR	0	1	2	3	4	5	6	7
0	NUL	(TC_7) DLE	SP	0	@	P	`	p
1	(TC_1) SOH	DC_1	!	1	A	Q	a	q
2	(TC_2) STX	DC2	"	2	B	R	b	r
3	(TC_3) ETX	DC_3	#	3	C	S	c	s
4	(TC_4) EOT	DC_4	$	4	D	T	d	t
5	(TC_5) ENQ	(TC_8) NAK	%	5	E	U	e	u
6	(TC_6) ACK	(TC_9) SYN	&	6	F	V	f	v
7	BEL	(TC_{10}) ETB	'	7	G	W	g	w
8	FE_0 (BS)	CAN	(	8	H	X	h	x
9	FE_1 (HT)	EM	)	9	I	Y	i	y
10	FE_2 (LF)	SUB	*	:	J	Z	j	z
11	FE_3 (VT)	ESC	+	;	K	[	k	
12	FE_4 (FF)	IS_4 (FS)	,	<	L	\	l	
13	FE_5 (CR)	IS_3 (GS)	-	=	M	]	m	
14	SO	IS_2 (RS)	.	>	N	↑	n	~
15	SI	IS_1 (US)	/	?	O	←	o	DEL

Appendix B

International standards governing serial data communication can be found in

1. CCITT V22 and V22 bis; data signalling rates (standard baud rates).
2. CCITT V21, V23, V26, V26 bis, V27, V30 and V35; modem standards.
3. EIA RS232-C, CCITT V24; interfaces employing serial binary data interchange.
4. ISO 646; 7-bit coded character set.
5. ISO 1177; character structure for start/stop and synchronous transmission.
6. ISO 1745; basic mode-control procedures for data communication systems.*
7. ISO 2111; code-independent transmission procedures.

CCITT: Comité Consultatif International de Telegraphie et du Téléphonie.

EIA: Electronic Industries Association (USA).

ISO: International Standardization Organization.

* Commonly called Communications Control Procedures.

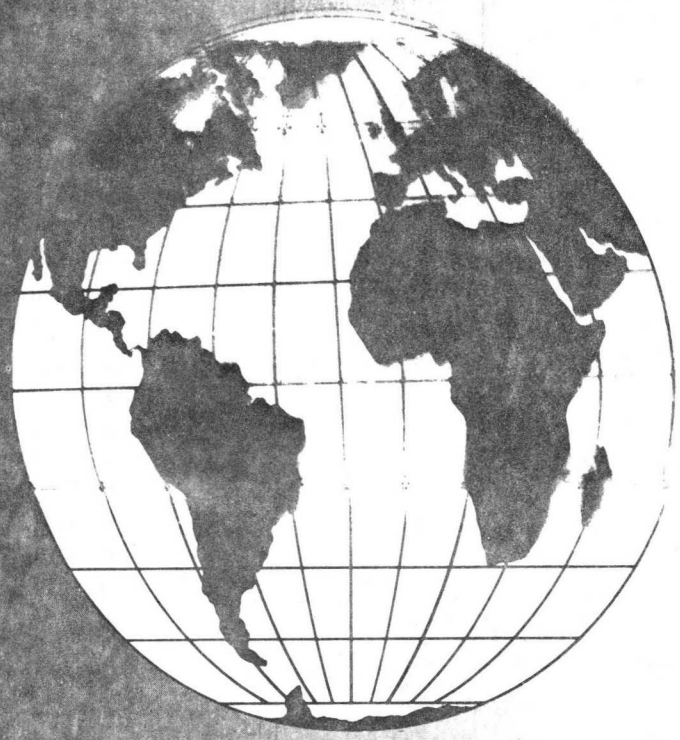
Related 2650 publications

no.	title	summary
AS50	Serial Input/Output	Using the Sense/Flag capability of the 2650 for serial I/O interfaces.
AS51	Bit & Byte Testing Procedures	Several methods of testing the contents of the internal registers in the 2650.
AS52	General Delay Routines	Several time delay routines for the 2650, including formulas for calculating the delay time.
AS53	Binary Arithmetic Routines	Examples for processing binary arithmetic addition, subtraction, multiplication, and division with the 2650.
AS54	Conversion Routines	• Eight-bit unsigned binary to BCD • Sixteen-bit signed binary to BCD • Signed BCD to binary • Signed BCD to ASCII • ASCII to BCD • Hexadecimal to ASCII • ASCII to Hexadecimal
AS55	Fixed Point Decimal Arithmetic Routines	Methods of performing addition, subtraction, multiplication and division of BCD numbers with the 2650.
SP50	2650 Evaluation Printed Circuit Board (PC1001)	Detailed description of the PC1001, an evaluation and design tool for the 2650.
SP51	2650 Demo System	Detailed description of the Demo System, a hardware base for use with the 2650 CPU prototyping board (PC1001 or PC1500).
SP52	Support Software for use with the NCSS Timesharing System	Step-by-step procedures for generating, editing, assembling, punching, and simulating Signetics 2650 programs using the NCSS timesharing service.
SP53	Simulator, Version 1.2	Features and characteristics of version 1.2 of the 2650 simulator.
SP54	Support Software for use with the General Electric Mark III Timesharing System	Step-by-step procedures for generating, editing, assembling, simulating, and punching Signetics 2650 programs using General Electric's Mark III timesharing system.
SP55	The ABC 1500 Adaptable Board Computer	Describes the components and applications of the ABC 1500 system development card.
SS50	PIPBUG	Detailed description of PIPBUG, a monitor program designed for use with the 2650.
SS51	Absolute Object Format	Describes the absolute object code format for the 2650.
MP51	Initialization	Procedures for initializing the 2650 microprocessor, memory, and I/O devices to their required initial states.
MP52	Low-Cost Clock Generator Circuits	Several clock generator circuits, based on 7400 series TTL, that may be used with the 2650. They include RC, LC and crystal oscillator types.
MP53	Address and Data Bus Interfacing Techniques	Examples of interfacing the 2650 address and data busses with ROMs and RAMs, such as the 2608, 2606 and 2602.
MP54	2650 Input/Output Structures and Interfaces	Examines the use of the 2650's versatile set of I/O instructions and the interface between the 2650 and I/O ports. A number of application examples for both serial and parallel I/O are given.
TN 064	Digital cassette interface for a 2650 microprocessor system	Interface hardware and software for the Philips DCR digital cassette drive.
TN 069	2650 Microprocessor keyboard interfaces	Simple interfaces for low-cost keyboard systems.

Electronic components and materials

for professional, industrial and consumer uses

from the world-wide Philips Group of Companies



Argentina: FAPESA I.y.C., Av. Crovara 2550, Tablada, Prov. de BUENOS AIRES, Tel. 652-7438/7478.
Australia: PHILIPS INDUSTRIES HOLDINGS LTD., Elcoma Division, 67 Mars Road, LANE COVE, 2066, N.S.W., Tel. 427 08 88.
Austria: ÖSTERREICHISCHE PHILIPS BAUELEMENTE Industrie G.m.b.H., Triester Str. 64, A-1101 WIEN, Tel. 62 91 11.
Belgium: M.B.L.E., 80, rue des Deux Gares, B-1070 BRUXELLES, Tel 523 00 00.
Brazil: IBRAPE, Caixa Postal 7383, Av. Paulista 2073-S/Loja, SAO PAULO, SP, Tel. 284-4511.
Canada: PHILIPS ELECTRONICS LTD., Electron Devices Div., 601 Milner Ave., SCARBOROUGH, Ontario, M1B 1M8, Tel. 292-5161.
Chile: PHILIPS CHILENA S.A., Av. Santa Maria 0760, SANTIAGO, Tel. 39-40 01.
Colombia: SADAPE S.A., P.O. Box 9805, Calle 13, No. 51 + 39, BOGOTA D.E. 1., Tel. 600 600.
Denmark: MINIWATT A/S, Emdrupvej 115A, DK-2400 KØBENHAVN NV., Tel. (01) 69 16 22.
Finland: OY PHILIPS AB, Elcoma Division, Kaivokatu 8, SF-00100 HELSINKI 10, Tel. 1 72 71.
France: R.T.C. LA RADIOTECHNIQUE-COMPELEC, 130 Avenue Ledru Rollin, F-75540 PARIS 11, Tel. 355-44-99.
Germany: VALVO, UB Bauelemente der Philips G.m.b.H., Valvo Haus, Burchardstrasse 19, D-2 HAMBURG 1, Tel. (040) 3296-1.
Greece: PHILIPS S.A. HELLENIQUE, Elcoma Division, 52, Av. Syngrou, ATHENS, Tel. 915 311.
Hong Kong: PHILIPS HONG KONG LTD., Comp. Dept., Philips Ind. Bldg., Kung Yip St., K.C.T.L. 289, KWAI CHUNG, N.T. Tel. 12-24 51 21.
India: PHILIPS INDIA LTD., Elcoma Div., Band Box House, 254-D, Dr. Annie Besant Rd., Prabhadevi, BOMBAY-25-DD, Tel. 457 311-5.
Indonesia: P.T. PHILIPS-RALIN ELECTRONICS, Elcoma Division, 'Timah' Building, Jl. Jen. Gatot Subroto, JAKARTA, Tel. 44 163.
Ireland: PHILIPS ELECTRICAL (IRELAND) LTD., Newstead, Clonskeagh, DUBLIN 14, Tel. 69 33 55.
Italy: PHILIPS S.p.A., Sezione Elcoma, Piazza IV Novembre 3, I-20124 MILANO, Tel. 2-6994.
Japan: NIHON PHILIPS CORP., Shuwa Shinagawa Bldg., 26-33 Takanawa 3-chome, Minato-ku, TOKYO (108), Tel. 448-5611.
(IC Products) SIGNETICS JAPAN, LTD., TOKYO, Tel. (03) 230-1521.
Korea: PHILIPS ELECTRONICS (KOREA) LTD., Philips House, 260-199 Itaewon-dong, Yongsan-ku, C.P.O. Box 3680, SEOUL, Tel. 44-4202.
Mexico: ELECTRONICA S.A. de C.V., Varsovia No. 36, MEXICO 6, D.F., Tel. 5-33-11-80.
Netherlands: PHILIPS NEDERLAND B.V., Afd. Elonco, Boschdijk 525, NL-4510 EINDHOVEN, Tel. (040) 79 33 33.
New Zealand: Philips Electrical Ind. Ltd., Elcoma Division, 2 Wagener Place, St. Lukes, AUCKLAND, Tel. 867 119.
Norway: ELECTRONICA A/S., Vitaminveien 11, P.O. Box 29, Grefsen, OSLO 4, Tel. (02) 15 05 90.
Peru: CADESA, Jr. Ilo, No. 216, Apartado 10132, LIMA, Tel. 27 73 17.
Philippines: ELDAC, Philips Industrial Dev. Inc., 2246 Pasong Tamo, MAKATI-RIZAL, Tel. 86-89-51 to 59.
Portugal PHILIPS PORTUGESA S.A.R.L., Av. Eng. Duharte Pacheco 6, LISBOA 1, Tel. 68 31 21.
Singapore: PHILIPS SINGAPORE PTE LTD., Elcoma Div., POB 340, Toa Payoh CPO, Lorong 1, Toa Payoh, SINGAPORE 12, Tel. 53 88 11.
South Africa: EDAC (Pty.) Ltd., South Park Lane, New Doornfontein, JOHANNESBURG 2001, Tel. 24/6701.
Spain: COPRESA S.A., Balmes 22, BARCELONA 7, Tel. 301 63 12.
Sweden: A.B. ELCOMA, Lidingövägen 50, S-10 250 STOCKHOLM 27, Tel. 08/67 97 80.
Switzerland: PHILIPS A.G., Elcoma Dept., Edenstrasse 20, CH-8027 ZÜRICH, Tel. 01/44 22 11.
Taiwan: PHILIPS TAIWAN LTD., 3rd Fl., San Min Building, 57-1, Chung Shan N. Rd, Section 2, P.O. Box 22978, TAIPEI, Tel. 5513101-5.
Turkey: TURK PHILIPS TICARET A.S., EMET Department, Inonu Cad. No. 78-80, ISTANBUL, Tel. 43 59 10.
United Kingdom: MULLARD LTD., Mullard House, Torrington Place, LONDON WC1E 7HD, Tel. 01-580 6633.
United States: (Active devices & Materials) AMPEREX SALES CORP., Providence Pike, SLATERSVILLE, R.I. 02876, Tel. (401) 762-9000.
(Passive devices) MEPCO/ELECTRA INC., Columbia Rd., MORRISTOWN, N.J. 07960, Tel. (201) 539-2000.
(IC Products) SIGNETICS CORPORATION, 811 East Arques Avenue, SUNNYVALE, California 94086, Tel. (408) 739-7700.
Uruguay: LUZILECTRON S.A., Rondeau 1567, piso 5, MONTEVIDEO, Tel. 9 43 21.
Venezuela: IND. VENEZOLANAS PHILIPS S.A., Elcoma Dept., A. Ppal de los Ruices, Edif. Centro Colgate, Apdo 1167, CARACAS, Tel. 36 05 11.

Printed in The Netherlands Date of release: 9 February 1978 T.D. 9399 500 07201