INTERFACE BOARD FOR COMPUTERS

Description

In an installation or system with a medium/large computer it is desirable to be able to use intelligent terminals. When such intelligent terminals are used the degree of interaction between the medium/large computer and the terminals is governed to a large extent by the basic software of the computer, to the extent to which it is the latter which controls the channels of dialogue to the terminal.

In this context the applicant has developed a personal computer (PC) capable of acting as an intelligent terminal in such a way that it can use the resources of a medium/large computer through a communication channel, complying with a pre-established format.

The set of functions which the medium/large computer offers are structured like an intelligent terminal for which the file server is called FSERV, but it is not limited solely to providing access to files. It is also a logical entity in that dialogue exists between the PC and the medium/large computer.

At this point we decided to lay out a diagram according to which the PC could substitute for non-intelligent terminals, as well as intelligent terminals.

According to this diagram the medium/large computer

would consider the PC to be a set of three terminals, i.e:

- A logical terminal (Fserv)
- An interactive representational screen-type terminal (Display).
- A non-interactive printing-type terminal (Printer).

This set of three terminals combined in a PC are what we shall call an Office Work Station (O.W.S.).

Communication between a medium/large computer and terminals can be established through a series interface-type (SIF) communication channel, in such a way that all communications through the channel are available to all terminals but, using a logical identification system, only one of them can respond.

In the dialogue the initiative belongs solely to the controller, in this case the medium/large computer, and therefore a procedure of periodic interrogation is established between the latter and the terminals. The fact that the line is shared by various terminals means that the total response speed is determined by the response speed of each terminal.

An object of this invention is the provision of an interface useable as an interface between a communication channel and a PC bus, which handles the communication protocol with a work station (W.S.C.) as described above, such that more

response time is allowed for the PC, at the same time maintaining the dialogue with the medium/large computer.

An aspect of the invention provides an interface board for computers useable as an interface between a communications channel and the bus of a personal computer connected to a medium/large computer replacing intelligent and non-intelligent terminals, such that the medium large computer considers the personal computer to be a set of three terminals, one logical, another interactive and a third non-interactive, which together with the personal computer form a working station characterised in that the board includes functional blocks comprising an input storage area (1), a central storage area (2), a central memory (3) and a communication memory (4) and for sending and receiving information a block (5) capable of writing information directly to the input storage area (1), or, via a processor (6), of communicating with the other functional blocks; the arrangement being such that the board functions as an interface between the communications channel and the bus of a personal computer and handles the communication protocol and the separation between the three terminal devices of the work station.

The board may include:

- a microprocessor (6) which runs a fixed program for handling the protocol logic and maintaining a communication with the personal computer, circuitry for sending, receiving and pre-analysing information,

- static random access memory (RAM) and read only memory (EPROM), each of eight kilobytes, together forming the central memory functional block (3),
- random access memory (RAM) of one kilobyte, which forms the functional communication memory block (4) with two simultaneous access ports which enables communication between the interface and the personal computer bus.

Preferably the input storage area functional block (1) is the first receptor of information and is operable to determine the maximum length of a block of information to be handled by the interface board and is common to the three logical devices which form the work station.

Desirably, the central storage area functional block (2) is operable as an intermediate store for information in the input storage area (1) to free the input storage area.

It may be provided that the communication memory functional block (4) which has two ports with simultaneous access from the interface and from the personal computer is operable to direct signals from one port to the other such that the interface generates interrupts for the personal computer.

The functional block (5) for sending and receiving information may comprise two functional branches, one for sending and the other for receiving information, in each one of which there is first a conditioning block (7) for the signal in the branch, and the sending branch additionally include a block (9) for the serialisation of

the words and a block (8) for parity bit generation.

The receiving branch may additionally include a serial/parallel converter block (10), a block (11) for address and parity checking, an error checking block (12), and a pre-decoding block (13), of which an EPROM memory of four kilobytes used as a decoder forms part, such that if the information is control information, it is passed via an interrupt block (14) to the processor and if it is data, it is passed via a write block to the input storage area (1).

This board provides the separation between the three devices of the work station (W.S.C) in such a way that their data is separated in the connection or interface to the PC.

The microprocessor which runs the fixed program (Firmware) for handling the protocol logic and maintaining the communication with the PC preferably has a working frequency of about 3.75 MHz.

The circuitry for sending, receiving and pre-analysing information may be implemented by SSI and MSI circuits and an electrically programmable read only memory (EPROM) of four kilobytes used as a decoder.

A board embodying the invention offers the advantages of:-

A. Fast analysis of the information and quick response time to fulfil the physical specifications of the channel and to optimise efficiency as regards line utilisation (since we are dealing with a line which can be shared between various terminals).

B. Sufficient processing capacity to be able to handle the communication protocol and to distribute the information between the three constituent units which form the work station (O.W.S.). This enables the processor of the PC to be freed from an excessive overload and to avoid a slowing down of the dialogue with the medium/large computer.

In addition, it enables part of the information processing to be included in the board itself.

C. Adequate storage capacity for the information in transit to be retained efficiently. The fact that intermediate data storage registers exist allows a better average flow.

As a consequence of the processing capacity introduced, flexibility to adapt to various communication protocols is obtained, so that in addition to the standard protocol required for the work station (O.W.S.) it is possible to handle other protocols under the same physical conditions.

The above and other aspects features and advantages of the invention will become apparent from the following description of one embodiment of the invention now made with reference to the attached drawings which represent, merely by way of illustrative and non-restrictive example one interface board embodying the invention. In the drawings:-

Fig. 1 represents the functional block diagram of an interface board assembly proposed.

Fig. 2 is a functional diagram of the send/receive block of the functional board assembly.

Figs. 3 to 8 are distinct representative parts of the electrical diagram of the board.

The Figures show a board intended to be used as an interface between a communications channel and the bus of a personal computer known as a PC, the said board being constituted according to a functional block diagram, as represented in Fig. 1, in which there are four functional blocks intended for the storage of information with various purposes, the said blocks consisting of: an input storage area or buffer (1), a central storage area or buffer (2), a central memory (3) and a double port communication memory (4).

For sending and receiving there is a special block (5), which is capable of writing directly into input storage area (1) or even of communicating with the rest

of the functional blocks through a processor (6).

In the installation mentioned, the input storage area (1) is the first receptor of information and fixes the maximum length of the block which can be handled by the interface, the said input storage area (1) being common for the three constituent units which the system includes, being: an intelligent terminal (Fserv), an interactive terminal (Display) and a non-interactive terminal (Printer). These three units combined in a PC, together form a work station (O.W.S.).

The central storage area (2) is an intermediate store to provide efficient retention in such a way that, bearing in mind that the input storage area (1) is common to the three constituent units forming the O.W.S., it is convenient to have an area available for each unit in which a copy of input storage area (1) can be made to free it quickly.

The communications memory (4) possesses two ports with simultaneous access from the interface and from the PC, so it constitutes the vehicle of interchange of information between the two.

This memory provides in addition a mechanism for activating signals in one port from the opposite port, these signals being able to be used to carry out the physical communication protocol, so that the interface can generate interrupts on the PC and this can in turn indicate the necessity for attention to the interface.

The central memory (3) contains the program and the necessary control variables, whilst the processor (6) pays attention to the line commands using interrupts generated by the send/receive block (5), whilst the dialogue with the PC is maintained using polling on the part of the interface and by interrupts on the part of the PC.

The send/receive block (5) is constituted according to a functional diagram as represented in Fig. 2, the said diagram showing two different branches, one corresponding to send (right hand branch) and the other to receive (left hand branch), in each one of which there is a first conditioning block (7) of the signal between the line and the rest of the circuits.

In the send branch there is also a parity generation block (8) and another block (9) for the serialisation of the words, this block having the characteristic of enabling a fixed number of bits to be sent per line carrying out one single instruction to the processor, whilst the total parity (and in this case the intermediate) are generated automatically.

In the receive branch after the appropriate serial/parallel conversion (10) the signal is verified as follows:

- Checking parity and testing the direction of the terminal, which are carried out in block (11).
- Checking errors in transmission, which is carried

out in block (12), and storing these so that the processor can take the necessary actions when interrupted.

- Pre-decoding, which takes place in block (13), carried out by an EPROM, such that if the information is control information it passes to block (14) to advise the processor so that it carries out the appropriate operations, whereas if it is data it passes to block (15) to be written directly into the input storage area or buffer (1).

In accordance with all this, the function referred to in the electrical diagram represented by parts, in Fig. 3 to 8, is translated into the operations which are specified as follows:

1. RECEPTION AND SERIAL/PARALLEL CONVERSION

1.1. RETRIEVAL OF THE LOGICAL VALUES OF "BIT"

The signal which arrives by the line, which can be very attenuated if the source is at a great distance, is retrieved by the operational input amplifier (16) (see Fig. 3), which delivers the said restored signal on pin (17).

Circuits (18 & 19) are connected together forming a shift register of 16 bits whose function consists of retrieving bit by bit the logical value one or zero of the signal delivered by the operational amplifier and which enters in the said register by pins (20 & 21) of

circuit (19).

Detection of the logical values of the bits is done by sampling the input signal with a clock of 15 MHz in such a way that the signal which arrives at pins (20 & 21) of (19) is copied every 66.6 ns in the following position of the shift register, and a positive edge in the output of the amplifier generates an edge on pin (22) of (19) 399.6 +/- 66.6 ns later. The said edge is used as a pulse or signal of bit validation or strobe and the value of pin (23) of (19) at that moment determines the logical value of the bit. To make the circuit connection, the line being active, the correct value is retrieved from the first complete bit received.

1.2 COUNTING ZEROS

According to Fig. 3, circuit (24) is charged with counting zeros in such a way that each bit validation pulse in pin (22) of (19) gives a clock pulse to the counter, which carries out one of the two following actions:

COUNTING: If pin (25) is at one, i.e. if pin (17) of operational amplifier (16) is at zero, this indicates that the bit received is a zero.

LOADING: If pin (25) is at zero, this indicates that the bit received is a one.

As long as ones are received before the counter has reached the end, the counter will be reloaded which will put it in the position of counting equal to two; and

insofar as fourteen consecutive zeros are received, the counter will arrive at the fifteen state, when it will pull up pin (26) and generate a -CYO signal which will block the counter until the arrival of a one which will reload it.

The -CYO signal remains active after the arrival of fourteen zeros until a one arrives, i.e. a minimum of fourteen zeros are counted, this being necessary to anticipate cases in which the fourteen zeros have been preceded by a word with zeros at the end.

1.3 COUNTING THE BITS OF WORDS

As can be appreciated in Fig. 3, circuit (27) is used for counting the bits of each word following fourteen zeros, in such a way that the clock pulses of the counter give us the bit validation pulse on pin (22) of (19) and it causes either counting or loading of the counter according to the state of pin (28).

When pin (26) of counter (24) goes high NOR gate (29) generates a zero at the load input of counter (27), which on receiving the corresponding clock edge reloads with the state two, and from the moment when the output of (24) goes low, counter (27) will count fourteen bits and will authorise the possibility of producing the word strobe validation pulse in NAND gate (30). At the same time this same signal causes reloading of the counter to initiate counting the bits of the following word, when the signal CYW is activated, which will place a zero in

the load input through gate (29).

Supposing that the word strobe validation pulse is authorised at pin (31) of NOR gate (30), the edge which affects outputs (32 & 33) of (18) using inverter (34) causes a pulse on pin (35) of bistable (36) which toggles it. Immediately on the following shift in (18), pin (37) resets the bistable via inverter gate (34). In this way a word validation pulse 132 ns wide is obtained.

1.4 CHECKING PARITY

Checking the parity of the word which is arriving is carried out by bistable (38), which appears in Fig. 3, in such a way that the clock of this bistable is obtained from the bit strobe validation signal on pin (22) of (19) via inverter gate (39). Inputs (40 & 41) of the said bistable (38) are connected at pin (23) of (19) which holds the logical value of the bit.

When the bit which enters is a one the bistable is obliged to toggle whereas when it is a zero there is no change, and since the number of ones must be odd and we are setting off from zero, the negative output (42) of bistable (38) must be a one if there has been no error. The signal -PRIT indicates an error if it is a one.

Resetting of the parity check bistable is carried out when the output of NAND gate (43) is zero, and this occurs when it is ready to initiate counting of the bits of a new word, the output of (29) being zero. In this case outputs (44 and 45) of (19) and gate (46) provide

the clearing pulse of the bistable via NAND gate (43), see Fig. 3.

1.5 INDICATION OF ZEROS

Once more referring to Fig. 3, on storing the information that fourteen zeros have arrived, the output of gate (43) being the input of this bistable, and the pulse edge which resets the parity check bistable, serve to store in bistable (47) the output state of counter (24) which enters by pin (48).

The indication of fourteen zeros remains active from when it starts to receive the first bit of the word after them, until the beginning of the following word.

1.6 SERIAL/PARALLEL CONVERSION

Also in Fig. 3, circuits (49) and (50) constitute a shift register of 16 bits, whose clock provides a bit strobe validation pulse from pin (22) of (19), its input being by pins (51 & 52) of (50), from pin (23) of (19). In this way the logical contents of the fourteen bits of the word are stored.

The bits corresponding to the address enter in comparator (53) to compare them with the signal selection in output (54) at NAND gate (55) whose output corresponds to the SELESP signal which is activated when the address is one-one-one.

The bits corresponding to outputs (56) of (50) and (57) of (49) are combined in NOR gate (58), since for all the control words (CW) these two bits are zero, and for

their later decoding it is irrelevant which of the two is one, to avoid the corresponding error.

1.7 SIGNAL OF COINCIDENCE (MATCH)

This signal indicates that the address contained in a control word CW coincides with that selected by the miniature switches and is obtained at the output of NOR gate (59), being able to be actuated either starting from output (54) of comparator (53) or when there is a special selection of the output of (55).

Comparator (53) has on the one hand the bits corresponding to the address of the CW coming from (49), and on the other hand the values one or zero selected by the miniature switches.

The -PRIT signal also enters by pin (60) of the comparator to ensure that it is zero. Comparison is authorised by pin (61) where the signal of fourteen zeros arrives, and in this way the MATCH signal can only be activated when words which have been preceded by fourteen zeros and which have no parity error are received.

Another means of activation of the MATCH signal is by special address, in which case all the stations are affected and carrying out the comparison is not necessary.

1.8 SPECIAL SELECTION

The SELESP signal is activated when a control word CW has as its address one-one-one, the signal of four inputs being obtained from NAND gate (55), three of which

correspond to the three bits of address of the CW coming from (49), and the other corresponding to the authorisation given by NAND gate (62). Special selection is only possible when the signal of fourteen zeros is activated and there is no parity error.

2. DECODING AND CHECKING ERRORS

2.1 EPROM DECODER

Passing on to Fig. 4, decoding of the control words is carried out using an EPROM (63), (see Fig. 4), at whose address inputs the following signals arrive:

Outputs corresponding to the shift register formed by circuits (49) and (50), (Fig. 2), excluding those of address and taking the combination of (64 & 65) by NOR gate (58).

Signal on pin (66) of (63), which indicates that the words to be decoded are data ESCR.

Signal of special selection SELESP.

The output of data from the said EPROM can be divided into two groups:

a) (67 & 68) are signals with special functions.

b) (69 & 70) form a command or error code which is memorised in (71).

2.2. CHECKING ERRORS

Continuing with Fig. 4, we believe that in the process of decoding it is necessary to carry out a check for any type of error, there being two types of error detection:

a) The first type are those which when detected are stored in (72) and which can be read by the processor when it is interrupted.

b) The second type are those which at the time of occurring generate an interrupt and are coded in data outputs (69 & 70) of EPROM (63).

The following signals enter (72): giving a defined value of zero or one, setting and resetting of errors:

-TCK1, which enters by pin (73) and comes from the output of NAND gate (74). This signal is generated in output (68) of EPROM (63) and is filtered by the STRB signal (word validation pulse) and the MATCH signal (coincidence of addresses), and is used to guard against the reception of an incorrect control word CW.

-TCK2, which enters by pin (75) and comes from the output of NAND gate (76). This signal is generated in output (77) of EPROM (63) and is filtered by the STRB signal and is used to guard against the reception of an incorrect data word DW.

-TCKP, which enters by pin (78) and comes from the output of NAND gate (79). This signal is generated by the -PRIT parity signal, filtered by the strobe STRB validation signals or pulses and whether either fourteen zeros have been received (in the case of a CW) or it is data (in the case of a DW) as given by NOR gate (80). This signal is used to guard against the reception of a word with a parity error, which naturally could not be

attributed to any particular station.

-AUCK, which enters by pin (81) and comes from the circuitry which constitutes the direct memory access (D.M.A.).

-BUCK, which enters by pin (82) and is used to erase the error written by AUCK. Address decoder (83) in Fig. 7 generates it.

-BTCK, which enters at NAND gate (84) whose output enters in pins (85 & 86) of (72). It is used to erase errors written by -TCK1, -TCK2 and -TCKP and is generated by address decoder (83) in Fig. 7. The errors in transmission are also erased by signal BSELES generated by the input of a written command.

2.3 WRITTEN COMMANDS

Continuing with Fig. 4, when a written command is recognised the necessary actions must be carried out on one hand to enable the successive words to request DMA so they can be written in the memory and on the other hand to give an indication to the EPROM that the successive words must be analysed as data.

Output (67) of EPROM (63) will be activated with written commands, and, conveniently filtered by signals STRB and MATCH, generates the signal -SELES at pin (87) of (88).

This signal is held in (72) when it enters by pins (89 & 90). To erase this indication the BSELES signal is used, which is generated in decoder (83) of Fig. 7.

On the other hand, signal SELE acts as a clock on pins (93 & 94) of two bistables (91 & 92), which activate signals WR and ESCR when the inputs are held to ground.

The WR signal is used to enable the DMA request via NAND gate (95) of Fig. 6, and the ESCR signal enters in EPROM (63) by pin (66) of the latter and conditions it for analysis of the word as a data word DW.

This signal ESCR is deactivated when the arrival of fourteen zeros or resetting occurs, using the CYORES signal which comes from NAND gate (96) in Fig. 6. The WR signal is also deactivated by this signal, but it can be deactivated in turn by any error in transmission via NAND gate (97).

2.4 INTERRUPT REQUEST

To advise the processor of the input of a new command, or even to inform it of a specific error code, EPROM (63) activates output (98), see Fig. 4, and for the interrupt request to proceed there are two routes:

a) The first route is via gate (99) and is filtered by signals STRB and MATCH, this route being the normal one to inform the processor of the arrival of a new command or of a specific error in the reception of a control word CW.

b) The second route is via gate (100) and is filtered by signals ESCR and STRB, this route being used to inform the processor of any specific error code in the reception of a data word DW.

Signal -INTCW on pin (101) of (102) corresponds to the interrupt line, and this signal is activated by the output of NAND gate (103) which combines the inputs of gates (99 & 100).

This signal is deactivated by the -LEST signal generated by address decoder (104) in Fig. 7.

The output of NAND gate (103), in addition to activating the -INTCW signal causes the loading of (71) with a code generated by EPROM (63) in its outputs (69 & 70) and with the information held in (72). At this, when the processor recognises the interrupt and activates the -LEST signal, the output of information from (71) is authorised so that it can be read at the computer connection.

3. LINE OUTPUT

3.1. OBTAINING THE FOURTEEN BITS OF A WORD

Given that the protocol demands that the communication is carried out with words of fourteen bits and the processor has a bus of eight bits, some address lines in the decoding of the terms must be used to generate all the bits with one single instruction of input-output. In addition, the equipment itself carries out the calculation of parity.

For the serialisation of the fourteen bits of the word, they are loaded in a shift register with parallel input and series output formed by circuits (105 & 106), see Fig. 5.

The first bit is the start one; it will always be one and has been assigned address line (134), whereas the second bit indicates if we are dealing with a device status word (DSW) or a data word DW and has been assigned address line (133).

Bits three to ten correspond in the case of a DSW to the address of the station and to the state, and in the case of a DW to the eight bits of data. These bits have also been assigned to the data bus from the processor.

The eleventh bit corresponds in the case of a DSW to the device type indication and in this case is fixed at one, whereas in the case of a DW it corresponds to the parity bit of the eight data bits. To generate the appropriate value for this bit a parity generator (107) and a multiplexer (108) are provided in such a way that using address lines (135 & 136) either the parity of the eight bits of data which enter at (107) or a one are selected.

Bits twelve and thirteen correspond in the case of a DSW to identification of the device and are always one, whereas in the case of a DW they are always zero. Both are combined and assigned to address line (133), see Fig. 5.

The fourteenth bit always corresponds to the total parity of the word and in the case of a DW, its value is always fixed although the data has its own parity bit and the rest of the values are constant; whereas in the case

of a DSW its value depends on the parity of bits three to ten. Using lines (135 & 136), which correspond to the selection inputs of multiplexer (108), it is possible to obtain an appropriate value in both cases.

For the transmission of a DSW, the fourteenth parity bit has to be zero if bits three to ten have even parity, and one if their parity is odd, corresponding to output (108) of (107). This affirmation is based on the fact that the remainder of the bits which do not enter into the parity generating circuit are all ones.

3.2 AMPLITUDE CODING OF THE BITS

For transmission each bit is modulated as follows. The timing of the start of the bit is fixed. The bit starts at high level and goes low after a time depending on whether it is a one or a zero.

To obtain the above type of modulation a bistable (110) is used, see Fig. 5, whose direct output by pin (111) goes to the line exciter, and which is in three distinct successive positions:

a) Firstly it will be at high level, being the part common to the transmission both of one and of zero.

b) Secondly the output will depend on the value to be transmitted.

c) Thirdly it will be at low level, corresponding to the final part common to all bits.

To generate times appropriate for each state of bistable (110) represented in Fig. 5, a shift register of

sixteen bits with series input and parallel output is used, formed by circuits (112 & 113), in which a zero pulse is made to circulate. The edges which are produced in the various outputs of the shift register cause the corresponding changes in the state of the bistable.

Initially, before starting transmission, the shift registers (112 & 113) are cleared. In this way, via gates (114 & 115) a one is present at inputs (116 & 117) of shift register (113), and when this one reaches pin (118) of (112) the output of NAND gate (115) goes to zero. In the following clock the zero is introduced into the shift register and the output of gate (115) goes to one, see Fig. 5.

In this way a zero pulse remains circulating in the circuit formed by the shift registers and gates (114 & 115).

When the zero pulse passes by outputs (119 & 120) of (113), a negative pulse is caused by gates (124 & 125) which operates on the input of the bistable, giving it a defined value of zero or one.

When the zero pulse reaches output (121) of (113), it gives a clock pulse to bistable (110) which copies to the output the value which the output of shift register (106) shows at that moment, which will correspond to the bit which is going to be transmitted at that moment.

When the zero pulse reaches outputs (122 & 123) of (112), a negative pulse is produced via gates (126, 127 &

128) which operates the clear input of the bistable.

When the zero pulse which is circulating in (112 & 113) reaches outputs (116 & 117) of (113), a negative pulse is caused via gates (114 & 115), in such a way that this causes the zero pulse to start to introduce itself into the shift register on one hand, and a clock impulse to be given at (106 & 105) on the other hand, which causes the value of the following bit to be transmitted presenting itself in output (129) of (106).

3.3. COMMENCEMENT OF TRANSMISSION

The first step in initiating transmission consists of preparing the line, which is brought about by the -INTRA signal which comes from the address decoder of terminals (104) in Fig. 6. This signal initialises bistable (130), the output of which operates the input of line driver (131).

The next operation consists of executing a write instruction to a terminal, with the address bits being redundant for terminal selection, with an appropriate value according to the type of word which is required to be transmitted. This instruction will activate the -CSRS signal by gate (132), which on one hand causes the loading of shift registers (106 & 105) acting on their input (137) and on the other hand puts into operation the synchronisation circuit of the transmission clock.

The train of impulses from the transmission clock is that which enters by inputs (138) of (112 & 113) which

form the shift register in which the zero pulse circulates. NAND gate (139) is that which permits the passage of this train of impulses. This passage must be allowed at an appropriate moment, so that a jittering cannot occur in the clock input of (112 & 113), i.e. it should be guaranteed that the input of the train of impulses of the clock should be allowed when the latter is at low level.

The activation of the signal CSRS causes inputs (140 & 141) of bistable (142) to take the value zero and one respectively. At the following negative edge of CK15M this bistable will reset itself and at the subsequent one bistable (143) will reset itself. NAND gate (144) will not have changed its output since at no time has it held its two inputs at one. Enabling gate (139) is closed both by pin (145) and by pin (146).

When the deactivation of CSRS occurs, inputs (140 & 141) of bistable (142) take the value zero and one respectively. At the following negative edge of CK15M this bistable initialises and the reset input of bistable (147) is activated via gate (144). At the following negative edge of CK15M, bistable (143) will set, and the transmission clock will enable itself.

As soon as the clock enables itself at (112 & 113), the one which was on the input of pins (116 & 117) of (113), given by gate (115), starts to advance until on arrival at outputs (118 & 148) of (112) it causes a

negative pulse which introduces itself in (113) to continue circulating in the shift register.

3.4 PROCESS OF TRANSMISSION

Once the instruction of input/output terminal has been executed and due to the signal CSRS, the mechanism explained in the previous paragraph starts. To transmit the following word the processor gives a new instruction input/output terminal, without knowing whether the previous word has been totally transmitted.

Whilst it is transmitting a word, it may activate the WAIT signal via output (149) of NAND gate (150). Moreover, the ability to load shift registers (106 & 105) via NAND gate (151) is inhibited.

When signal CSRS is activated with the line condition prepared by the signal ENLIN of pin (152) of bistable (130), and a word is being transmitted, with the clock enabled by (139), the WAIT signal is activated until the transmission process has finished.

The input circuit checks the end of word utilising the same bits counter as is used to give the strobe validation pulse of a word received. This counter will always start from state two, given that for each transmission a reception has had to precede, the last bit of which will have loaded it to cause it to pull up.

When the fourteenth bit of the word is transmitting, before the zero which is circulating by (112 & 113) arrives at output (148) of (112), the input circuit will

have activated the signal CYW and when this is produced it will put a one in input (140) of bistable (142). At the following negative edge of CK 15 gate (139) will close and it will enable itself to initiate the process for the following transmission inhibiting the WAIT signal and enabling the reloading of (106 & 105).

Once all the words of a message have been transmitted and it has been ensured that the last word has been transmitted completely, the signal FITRAS will activate which comes from the circuit of the address decoder of terminals (83) in Fig. 7, which will clear the line and will initialise the bistables of the clock synchronisation circuit and the shift registers.

4. DIRECT MEMORY ACCESS (DMA) OF LINE INPUT

4.1. REQUEST FOR DIRECT MEMORY ACCESS (DMA)

Requests for DMA will be produced when write commands are received over the line. In this case the WR signal will activate as explained in the text on decoding. DMA is carried out by stealing of cycles, i.e. DMA is requested for each word which enters and the request is cancelled each time it is written.

The -BUSRQ signal requesting DMA leaves pin (153) of bistable (154) (see Fig. 6). On receiving it, a resetting of the bistable by pin (155) via NAND gate (95) is activated.

To have available sufficient time for the processor to recognise it, the request for DMA is carried out when

the word bits counter which is entering is in the state eleven. The signal BITX is generated from the outputs of counter (27) in Fig. 2 and from gate (156) of the same Fig. 2 and is used to carry out the request for DMA on the condition that the WR signal is activated.

When the processor recognises the request for DMA, activating signal -BUSAK the following actions occur:

Via the two successive inverters (157) (Fig. 5), used to strengthen this signal, the processing of the bits corresponding to data via (158) (Fig. 5) is enabled on the common data connection of the outputs of shift register (156) (Fig. 2).

Likewise the outputs of the multiplexer-decoder (159) circuit (Fig. 5) are enabled which generate the control signals of the rest of the circuits which form the DMA.

The signal BUSACK in pin (160) of (157) is used also to separate from the common connection of addresses the lines which are used to form the words of fourteen bits, using (161) of the detail in Fig. 5, with the sole aim of reducing noise.

4.2 DIFFERENT TYPES OF DATA

The data words sent by the line can be classified into two distinct types according to the place where the information which they contain must be stored, those which go to the register and those which go to the input storage area (Buffer). The registers are a memory area

in which the host controller writes control information related to the data which follows.

The data and the registers must go to the memory area assigned to them.

The writing process always starts by writing the registers and then the data. The registers can be written partially, but always consecutively. To indicate which register is required a special data word is sent, whose contents is not for storage but to indicate which register number has to be used for writing the following data words DW.

In addition there are a pair of registers in which the starting address is written, at which the data which then arrives must be stored. The process of writing from the host controller can be summed up thus in the following operations:

A write command is sent from the host controller. The interface recognises it and activates the WR signal.

The following word is a DW which indicates which register is going to be written from.

The following DW are registers and are stored successively starting with that indicated above. When data address registers are written the corresponding initialisations are carried out for the reception of data.

The following DW are data which are written successively up to the arrival of the fourteen zeros

which announce another command.

The three possible types of DW (selection of register, register and data) are distinguished by the twelfth and thirteenth bits.

4.3 WRITING OF REGISTERS

Once the request for DMA has been carried out, activating the -BUSRQ signal in (154) (Fig. 6), the processor will recognise it, activating the signal BUSACK, which via two inverters (157) will allow the turning over to the data bus of the eight bits of data of the DW via (158). Thus the outputs of multiplexer-decoder (159) will set themselves. The twelfth and thirteenth bits of the DW, which identify its type, enter into this circuit by (162 & 163) and the STPR of the word also enters.

The arrival of a DW corresponding to the selection of register causes the selection of output (164) of (159), which on production of a strobe validation pulse will cause a pulse in input (165) of (166) which will load the said counter with the value of the four least significant bits of the DW (only sixteen registers are permitted).

The arrival of the following DW corresponding to information for the register, will cause the selection of output (167) of (159). This activation enables the outputs of circuits (168 & 169) to pass to the address bus. (169) sets itself via gate (170).

Circuit (168) places between outputs (171 & 172) the contents of counter (166) which is the number of the register to which one wishes to write, whilst outputs (173) to (174) are put at zero.

Circuit (169) puts outputs (175 & 176) to zero, given that NAND gates (177) have a zero at their input. The signal at (178) is a one since it is an inactive output from (159). Outputs (179 & 180) are fixed at zero.

The strobe validation pulse causes a pulse at output (181) of (159) which has two objectives:

a) To activate the signal -SELD via gate (182) which on the one hand provides selection to the memory (183) of Fig. 7, via gate (184) of the same Fig. 7, and on the other hand gives a clock pulse to bistable (154) (see Fig. 6) which deactivates the -BUSRQ signal.

b) Via inverters (185 & 186) (Fig. 6), to cause a counting pulse by pin (187) of (166), which will cause the DMA writing pointer to pass to the following register.

Although the DMA can write sixteen registers, in reality the protocol specifies that it is possible to write only up to register number seven.

When one attempts to load counter (166) with a number larger than seven, the signal -AUCK is activated via gates (188, 189 & 190) which it will keep in error store (72) in Fig. 4.

Likewise when due to a clock impulse the counter reaches the value eight, the -AUCK signal will activate via gates (188 & 190).

4.4. WRITING DATA

As soon as the register counter (166) passes through the state of three the set of gates (191, 192 & 193) enable the DARL signal which, when a strobe validation pulse is given and output (181) of (159) is activated, will cause a pulse in inputs (194) of counters (195 & 196) which will load with the eight bits of data proceeding from the DW which indicate the low part of the address to which the data has to be written.

Similarly, when the register counter passes through the state of four, the activation of the DARH signal is enabled and, with the strobe validation pulse counter (197), will load with the four least significant bits of the DW.

When a DW is received, output (198) of (159) will activate which will cause the outputs of circuits (169 & 199) to pass to the nominated address bus. The signals on outputs (200 to 201) will be the contents of counters (195 & 196), the signals on outputs (175 & 176) will be the two last bits of counter (197) since gates (177) hold a one in their other input when an output of (159) is inactive. The signal in output (178) will be zero for output (198) of (159) to be active.

When a word strobe validation pulse occurs, output

(202) of (159) is activated, which will produce the activation of the selection signal -SELD and the increase of counter (196) and in this case of (195 & 197) by the propagation of the pulling up.

4.5 READING THE POINTERS OF THE DIRECT MEMORY ACCESS "DMA"

For the processor to be able to consider the number of data which has been written by the DMA, it is necessary for it to carry out readings of the final state of the pointers.

For this, circuits (203 & 204) can pass onto the nominated data bus the state of the counters of the DMA activating signals -PNT0 and -PNT1.

Circuit (204) is then used for the processor to be able to carry out readings of the signals which arrive by (205, 206 & 207), which indicate the address selected in miniature switches mSW and of the -INHC signal which comes from communication memory (4) with the PC bus.

5. THE CENTRAL PROCESSING UNIT "CPU"

A processor (208) constitutes the central processing unit CPU of the installation (see Fig. 7). The memory consists of an EPROM chip (209) and a RAM chip (183).

Memory selection is carried out by decoder (210), and to enable selection, signals -RD -WR and -MRQ are used, as well as the three most significant address lines.

The signals which the processor generates executing

instructions of input-output to write or read terminals, are the outputs of circuits (104) for the "in" and (83) for the "out". To enable selection signals -RD or -WR are used according to how they correspond, and signal -IORQ, as well as the lines of (211 to 212).

The address which is used to activate the corresponding signal is irrelevant, excepting in the case of signal -CSRS, as it causes a data word or a status word to be transmitted from the device, depending on the address lines not used in decoding.

6. COMMUNICATION WITH THE PC BUS

The communication of the interface with the PC is carried out via memory (4), which counts with two terminals or data ports and two address ports, and which can be accessed simultaneously from both.

Likewise (213) (Fig. 8) has the characteristic that when it is written in one position, output INTL activates and when it is written in another position output INTR activates.

Signal INTR is inverted by (214) and combines with one of the PC bus interrupt lines. In this way, when the interface wishes to communicate, an interrupt may be generated.

Signal INTL goes to input (215) of (204) in Fig. 6, with which it can be passed to the data bus if the processor activates the PNT1 signal.

Although access to the common memory can be

simultaneous, a conflict arises if access is at the same bit. In this case signal -BUSYL or -BUSYR in (213) is activated, depending on who has requested access in the first place. The signal -BUSYL activates the signal -WAIT via gate (216).

The selection of memory (4) is carried out from the side of the interface by means of signal -SELMC and input R/WR is activated by signal -WRC.

From the side of the PC bus the memory may be relocated depending on the position of the links. The four most significant bits of the PC address bus enter in one side of the comparator (217) and their value must coincide with that programmed via the two links for memory selection to be enabled.

7. RESET AND CLOCK

The output of comparator (218) provides reset. One of the inputs of the said comparator maintains a reference voltage given by the divider (219 & 220) which maintains a medium voltage on the positive input.

When an increase in potential occurs transistor (221) is saturated by the load current of capacitor (222) via resistor (223). The collector of (221) remains at a voltage close to five volts and (224) is saturated, putting a voltage of approximately 1,3 volts on the negative input of the comparator, thus generating a reset.

When capacitor (22) has charged to a voltage which

will not allow (221) to conduct, (224) also cuts off and the negative input of the comparator is pulled to ground via resistors (225 & 226).

When a lowering of potential occurs, (227) conducts with the discharge of capacitor (228) via resistor (229), which saturates (224) a reset is generated.

When connection occurs followed by rapid disconnection and reconnection, capacitor (230) allows (227) to conduct so generating a reset.

For an external reset, it is sufficient to discharge capacitor (222), either via a switch, or via gate (231).

The transmitting and receiving clock is obtained from oscillator (232) via gate (233) which is held by input (234) connected to five volts via resistor (235) so that the clock can be disabled by grounding if required.

The processor clock is obtained by dividing the previous signal by bistables (236) of the detail in Fig. 6, and via exciter circuit (237) of this same detail.

CLAIMS

1. An interface board for computers useable as an interface between a communications channel and the bus of a personal computer connected to a medium/large computer replacing intelligent and non-intelligent terminals, such that the medium/large computer considers the personal computer to be a set of three terminals, one logical, another interactive and a third non-interactive, which together with the personal computer form a working station characterised in that the board includes functional blocks comprising an input storage area (1), a central storage area (2), a central memory (3) and a communication memory (4), and for sending and receiving information, a block (5) capable of writing information directly to the input storage area (1), or, via a processor (6), of communicating with the other functional blocks; the arrangement being such that the board functions as an interface between the communications channel and the bus of a personal computer, and handles the communication protocol and the separation between the three terminal devices of the work station.

2. An interface board according to claim 1, characterised in that it includes:-

* C L M S *

- a microprocessor (6) which runs a fixed program for handling the protocol logic and maintaining a communication with the personal computer,
- circuitry for sending, receiving and pre-analysing information,
- static random access memory (RAM) and read only memory (EPROM), each of eight kilobytes, together forming the central memory functional block (3),
- random access memory (RAM) of one kilobyte, which forms the functional communication memory block (4) with two simultaneous access ports which enables communication between the interface and the personal computer bus.

3. An interface board according to claim 1 or claim 2, characterised in that the input storage area functional block (1) is the first receptor of information and is operable to determine the maximum

* C L M S *

length of a block of information to be handled by the interface board, and is common to the three logical devices which form the work station.

4. An interface board according to claim 1 or claim 2, characterised in that the central storage area functional block (2) is operable as an intermediate store for information in the input storage area (1) to free the input storage area.

5. An interface board according to claim 1 or claim 2, characterised in that the communication memory functional block (4) which has two ports with simultaneous access from the interface and from the personal computer is operable to direct signals from one port to the other such that the interface generates interrupts for the personal computer.

6. An interface board according to claim 1 or claim 2, characterised in that the functional block (5) for sensing and receiving information comprises two functional branches, one for sending and the other for receiving information, in each one of which there is first a conditioning block (7) for the signal in the branch and in that the sending branch additionally includes a block (9) for the serialisation of the words and a block (8) for parity bit

* C L M S *

generation.

7. An interface board according to claim 6, characterised in that the receiving branch additionally includes a serial/parallel converter block (10), a block (11) for address and parity checking, an error checking block (12), and a pre-decoding block (13), of which an EPROM memory of four kilobytes used as a decoder forms part, such that if the information is control information, it is passed via an interrupt block (14) to the processor and if it is data, it is passed via a write block to the input storage area (1).

8. A computer interface board substantially as hereinbefore described with reference to, and as illustrated in, the accompanying drawings.

9. A computer comprising an interface board as claimed in any preceding claim.