"INTERFACE CIRCUIT OF THE SLAVE TYPE"

Description

The invention relates to a slave-type interface circuit intended for the switching of a plurality of channels and comprising logic inputs capable of receiving data signals and clock signals supplied by a bus, a register for receiving the signals received from the bus via the logic inputs, and a memory for the data received by the register.

A circuit of this kind is known from European Patent 51332 in the name of N.V. PHILIPS' GLOEILAMPENFABRIEKEN, published on 11th April 1984, in the form of a station which is addressable notably as a master or as a slave and which is employed in a two-wire, series-type bus system.

For a given number of applications and particularly in television, one or more interface circuits are generally used between receiver and a television-equipment connector. The circuits generally known for this application effect switching operations in a conventional fashion and cannot be controlled from a bus.

The invention aims to provide an interface circuit which can be controlled from a bus, particularly of the previously mentioned type, but which can nevertheless be used, at least for some of its functions, in an environment where the switching operations are effected in a conventional fashion.

To achieve this,the invention is characterised in that it comprises

- a first decoder which controls the switching of the channels on the basis of logic signals stored in the memory,

- a plurality of addressing inputs, each of which can be set to a logic level so as to generate a displayed address,

- a second decoder which signals whether said displayed address corresponds to a first operating state of the circuit, on-line with said bus, or to a second state, off line, on the basis of logic control signals applied directly to the logic inputs,

- a pointer block which directs the signals available at the logic inputs either to said register if said displayed address corresponds to operation on-line with said bus (first state), or to the first decoder so as to control directly the switching of at least some of said channels on the basis of said logic control signals if said displayed address corresponds to off-line operation (second state),

- a blocking circuit which comprises a first series of logic gates which inhibit the passage of the logic signals stored in the memory towards said first decoder when the circuit is in its second state of operation (off-line).

The circuit in accordance with the invention may be rendered inactive by means of a second series of logic gates which inhibit the passage of the logic signals available at the output of the first decoder towards the channels to be switched in the presence of a logic signal which is referred to as the interface-circuit "off" signal.

The circuit can be initialised by means of a pulse generator which, when the circuit is activated, produces a pulse causing the initialisation of the memory and of at least a part of the register.

In a preferred embodiment, the circuit operates with a series bus, in its first state, characterized in that the register is a shift register which receives, in series at an input located at an upstream end, series logic information supplied by the bus in the form of cycles, each of which presents at least one sequence synchronized by a clock signal supplied by the bus when the circuit is in its first state, and produces, at an output located at a downstream end, an end-of-sequence acknowledge signal which is applied to the input of a first control flip-flop

which is synchronised by said clock signal and whose output synchronises a second control or pointer flip-flop whose inverted output is looped to its input, in that the presence of a logic level 1 at the output of the first flip-flop and at the inverted output of the pointer enables initialisation of said register, for example by setting its first flip-flop to the "1" state and resetting its other flip-flops to zero, in that it comprises a logic comparator which identifies, in a given part of an address sequence, series logic information supplied by the bus if said address sequence corresponds to said displayed address, and which produces an identification logic signal, in that it comprises a circuit for storing said identification signal, and in that it comprises a gate enabling the loading into the memory of the data from the register when the output of the first control flip-flop and of the pointer are in the logic state "1", when a stored identification signal is present and also a said end-of-sequence acknowledge signal, which thus corresponds to the end of a data sequence which follows said address sequence.

In an embodiment the register is initialised between the address sequence and the data sequence by setting the first flip-flop of the register to the "1" state and resetting its other flip-flops to zero when the output of the first control flip-flop is in the "1" state and the output of the pointer is in the zero state.

In a preferred version the bus supplies, in accordance with the procedure described in the cited European patent, start and stop data for a cycle of data supplied by the bus with the aid of logic relations between the data and clock signals.

The circuit then comprises a third and a fourth control flip-flop which, when the interface circuit is in its first state, receive the clock signal at their input and are synchronized one by the information signal and the other by the inverse thereof, so that the output of the third flip-flop is set to the "1" state by a first type of coincidence

between the clock and information signals, corresponding to the start of a said cycle, and the fourth flip-flop is set to the "1" state by a second type of coincidence between the clock and information signals, corresponding to the end of a cycle, the third flip-flop being reset to zero by the inverse of the clock signals and the fourth flip-flop by the output of the third flip-flop, the output of the fourth flip-flop being designated so as to produce, when it is in the logic "1" state, initialisation of the register by setting its first flip-flop to the "1" state and resetting its other flip-flops to zero.

The fourth flip-flop can attractively be set to the logic "1" state when no identification has been stored after an address sequence. To this end, the circuit is arranged so that the fourth flip-flop has its output set to the logic "1" state when the stored identification signal is in the logic "0" state and when the output of the first flip-flop is in the logic "1" state, thus producing initialisation of the register.

The output of the fourth flip-flop can also be set to the logic "1" state in various configurations, particularly in the presence of a pulse produced by the pulse generator or when the output of the first and the second flip-flop and the clock signal all have the logic "1" level (end of cycle).

In an embodiment, the circuit for storing the identification signal comprises two inverting AND-gates, the output of one of which is looped to one of the inputs of the other and vice versa, the other input of the first of the two inverting AND-gates receiving the inverse of the identification signal, be it only if the output of the pointer has the logic "0" state and if the clock signal is in the logic "1" state, the other input of the second of the two inverting AND-gates receiving the logic product of the inverses of the outputs of the third and the fourth flip-flop so as to permit the storage of the identification signal in each cycle.

For the purpose of communication with the bus with a view to the acknowledgement of a data sequence following an address sequence ,the circuit preferably comprises an acknowledge flip-flop which is synchronised by the clock signal and whose input receives the logic product of the acknowledge signal and the stored identification signal, and produces an authorisation signal at its output to the bus. This acknowledge flip-flop may also be reset to zero by said pulse. Said pulse can also resetthe first control flip-flop to zero and set the fourth control flip-flop to the "1" state.

In an embodiment which is suitable for the case involving a two-wire series bus, the circuit is characterized in that it comprises two logic inputs which can be connected in the first state of the circuit, to a bus delivering an information signal and a clock signal, in that the memory comprises a plurality of memory flip-flops, and in that the first series of logic gates of the pointer circuit comprises a third and a fourth inverting AND-gate, one input of which is connected to the respective output of two said memory flip-flops and the other input of which receives from the second decoder a signal with the logic level "1" in the first state of the circuit and the logic level zero in the second state, the output of the third and of the fourth inverting AND-gate being connected to a respective input of a fifth and sixth inverting AND-gate whose output is connected to the first decoder, and in that the other input of the fifth and the sixth inverting AND-gate receives the respective output from an eighth and a ninth inverting AND-gate, respectively, validating the signals available at the logic inputs when the circuit is in its second state.

The invention will be more readily understood if the description which follows,and which is given by way of a non-exhaustive example, is read in conjunction with the figures, in which:

figure 1 shows an overall diagram of a device in accordance with the invention;

- figure 2 shows a diagram of the bus receiver of figure 1;

- figure 3 shows an embodiment of the pointer logic and the memory of figure 1;

- figure 4 shows an embodiment of the bus receiver of figure 1; and

- figures 5a and 5b show a cycle delivered by the bus and a time diagram of the principal signals of figure 4, respectively.

According to figure 1, the interface circuit comprises addressing inputs S_0, S_1 and S_2 each of which can be connected by wiring either to a given potential, for example, the power-supply voltage of the circuit, or to ground so as to generate an address A_0, A_1 and A_2, each bit of which is encoded either as 0 or 1. Such an address may also be generated on the basis of a multistate logic and a decoder.

The displayed address, A_0, A_1 or A_2, is applied to the input of a function decoder FDEC producing at its output a state-of-operation logic signal CS which depends on the state of operation of the circuit to which the displayed address corresponds. For example, it may be decided that all addresses except one (e.g. 111) are to correspond to operation of the interface circuit on-line with a bus (first state), the remaining address (e.g. 111) corresponding to operation of the circuit off-line (second state) on the basis of logic signals applied directly to some of its inputs.

The switching logic signal CS is applied to a pointer block SBL which also receives the signals SDA and SCL which correspond to the logic levels of inputs L_1 and L_2, respectively.

If the displayed address A_0, A_1 or A_2 corresponds to operation on-line with a bus, the signals applied to the inputs L_1 and L_2 are provided by the bus, SDA then being an information signal and SCL a clock signal. The pointer block SBL is controlled by the switching logic signal CS

so as to direct the signals SDA and SCL towards a bus receiver RBUS intended to perform certain logic operations and particularly to check whether the displayed address A_0, A_1 or A_2 corresponds or not to an encoded address present in the signal transmitted by the bus. The bus receiver RBUS receives the displayed address A_0, A_1 or A_2 in such a way as to perform this check.

If the check is positve, and subject to other logic operations, the bus receiver RBUS transmits an acknowledge signal ACK and authorises, via a loading signal LDA, the loading of data transmitted by the bus into a memory M. The memory M controls a series of switches COM via a switching decocer CDEC.

If the check is negative, another terminal has been addressed and the bus receiver is no longer active: no memory loading can now occur.

If the displayed address A_0, A_1 or A_2 corresponds to off-line operation, the signals applied to the inputs L_1 and L_2 are controlled logic signals which are applied directly to the switching decocer CDEC by the switching block SBL. Thus, in accordance with the invention the inputs L_1 and L_2 are used for two different functions. This makes it possible for a single interface circuit, notably an integrated circuit, to respond by simply displaying an address A_0, A_1 or A_2 to all possible uses of a range of products, among which those having the highest level are controlled by microprocessor and bus and the others in the conventional way.

The state-of-operation logic signal CS is also applied to a blocking circuit BC which inhibits the transport of data from the memory M to the switching decocer CDEC when the displayed address A_0, A_1 or A_2 corresponds to off-line operation.

As appears from figure 2, the bus receiver RBUS controls a gus logic BUSL, a register REG and an address sequence controller AIC.

The register REG receives the logic information signal SDA from the bus in the form of cycles, each of which

comprises an address sequence, a data sequence and a clock signal SCL.

The address sequence identification circuit AIC receives output signals from the register REG and also the signals A_0, A_1 and A_2 corresponding to the displayed address. It performs a logical comparison between all these signals in order, on the one hand, to establish whether the displayed address A_0, A_1 or A_2 corresponds to the address transmitted by the bus and, on the other hand, to recognise a flag, always identical, which is transmitted in the address sequence of a cycle, the purpose of this operation being the detection of any signal-transmission errors. The address sequence identification circuit AIC produces at its output and identification logic signal DVA which is received by the bus logic BUSL.

The bus logic BUSL also receives the logic information signal SDA and the clock signal SCL, notably for the purpose of recognising the beginning and the end of each cycle. It also receives an end-of-sequence acknowledge signal ACN which is produced by the register REG when all bits of a sequence have been received by the register REG.

The bus logic BUSL then controls the circuit on the basis of three logic signals:

a) a reset-to-zero signal RST1 from the register REG which is produced at the beginning of each sequence or in the event that the address is not recognised or in the event of its absence;

b) an authorisation signal ACK which is transmitted towards the bus (input L_1) in order to confirm the acceptance of a sequence by the logic and to authorise the transmission of the next sequence by the bus;

c) a signal LDA which orders the data in the register REG to be loaded into the memory M.

When the interface circuit is connected to the power supply source, the supply voltage V is preferably applied to a pulse generator PG which produces a pulse PON for initialisation of the bus logic BUSL, of the register REG, and also of the memory M.

As appears from figure 3, the memory M comprises eight D-type flip-flops M_0 to M_7, the blocking circuit BC comprising eight gates and inverting AND-gates 80 to 87, the pointer block SBL comprising four inverting AND-gates 72, 78, 88 and 89 and two AND-gates 28 and 58, the function decoder FDEC consisting of an AND gate 79, and the switching decoder CDEC being composed of a decoder CDEC 1, processing the information from the memories M_1 and M_2, and of a direct control effected from the other memories M_3 to M_7. The switching block COM consists of nine controlled switches I_1 to I_9. More specifically, each of the flip-flops M_0 to M_7 receives at its input D a respective output QR_0 to QR_7 from the register REG, the latter being formed, for example as will be seen in figure 4, by a shift register comprising flip-flops R_0 to R_7 receiving the logic information SDA in series. Each of the flip-flops M_0 to M_7 receives at its clock input CK the inverse $\overline{LDA}$ of the signal for loading data in the memory. When $\overline{LDA}$ changes to the state "1", authorising loading, $\overline{LDA}$ changes to zero, producing a negative-going edge which activates the inputs CK of the flip-flops M_0 to M_7. Additionally, the flip-flops M_0 to M_7 receive at their reset-to-zero input R the pulse PON produced upon connection to the power supply source.

It was decided arbitrarily that the displayed address 111 ($A_0 = A_1 = A_2 = 1$) would correspond to off-line operation. This condition is detected by the AND-gate 79 whose output produces the state-of-operation logic signal CS whose value is 0 for the 7 displayed addresses corresponding to the operation of the interface circuit in accordance with the first state, on-line with the bus, and "1" for the address 111 corresponding to the operation in accordance with the second, off-line state.

Branching is then effected in the following manner:

The signal Cs, inverted by an inverter 70, is applied to one of the inputs of the two AND-gates 58 and 28, whose other input receives the signal SCL and the signal SDA

respectively, deliver by the inputs L_2 and L_1, respectively. For the 7 displayed addresses corresponding to the first state, the gates 58 and 28 allow the signals SCL and SDA to pass in the direction of the bus receiver RBUS. The signal CS, inverted by the invertor 70, is also applied to one of the inputs of each of a series of AND-gates 83 to 87 and inverting AND-gates 80 to 82. The other input of the gates 80 to 87 receives the output Q from the flip-flops M_0 to M_7, respectively. For the 7 displayed addresses corresponding to the first state, the gates 80 to 87 authorise the transmission of memory information in the flip-flops M_0 to M_7 to the output of the gates 80 to 87. The outputs of the inverting AND-gates 81 and 82 is connected to an input of two inverting AND-gates 88 and 89, respectively, the other input of which is connected to the output of two inverting AND-gates 78 and 72, respectively. The inverting AND-gate 78 receives at one input the signal SCL delivered by L_2 and at its other input the operating-state signal CS, which is also applied to one of the inputs of the inverting AND-gate 72. The other input of the latter receives the signal SDA from L_1.

When CS = 0 (first state), the outputs of the gates 72 and 78 are in the "1" state and authorise the passage of information from the output of the gates 81 and 82 to the decoder CDEC1.

When CS = 1 (second state), the gates 28 and 58 are non-conducting and the gates 72 and 78 permit the passage of the logic signals SCL and SDA, corresponding this time to logic levels applied at inputs L_1 and L_2. A logic "1" is present at the output of the gates 81 and 82 and authorises the passage of SCL and SDA through the gates 88 and 89, so as to be applied directly to the decocer CDEC1.

Nine controlled switches I_1 to I_9 receive the output of nine AND-gates 101 to 109, one input of the letter receiving the output from an inverting AND-gate 90. The other input of the AND-gates 101 to 104 receives an output from the decoder CDEC1, and that of the AND-gates 105 to 109 receives the outputs from the gates 83 to 87.

When the output of the inverting AND-gate 90 is in the "1" state (normal case, OFF = 0), the gates 101 to 109 authorise the control of the switches I_1 to I_9. In the first state, M_1 to M_7 control the switches I_1 to I_9 and, in the second state, the logic levels SCL and SDA applied directly to the two inputs of the decoder CDEC1 control the switches I_1 to I_4 in accordance with four possible configurations, e.g. the closure of one of them at a time.

As will be explained later in this description, the flip-flop M_0 is reserved for the storage of an authorisation state D_0 for a stop command OFF. The output of the inverting AND-gate 80 is applied to an input of the inverting AND-gate 90 whose other input receives said command OFF. If $Q(M_0) = 0$, the output of the gate 80 is at "1" level and authorises the transmission of the command OFF via the gate 90. The output of the gates 101 to 109 is then "0" and deactivates the switches I_1 to I_9.

As the other input of the inverting AND-gate 80 receives the signal CS inverted by 70, if CS = 0 (first state), the output of the gate 80 is determined by $Q(M_0)$ and, if CS = 1, (second state), the transmission of the command OFF is still authorised.

As figure 4 shows, the register REG is a shift register consisting of nine D-type flip-flops $R_0 \ldots R_8$ which are loaded upon a negative-going edge applied to their clock input CK. The output Q of the registers R_0 to R_7 is connected to the D-input of the next flip-flop, i.e. R_1 to R_8, respectively. Each of the flip-flops R_0 to R_8 is clocked at its input CK via SCL of the clock signal SCL so as to operate upon the positive-going edges of the clock signal SCL. The flip-flop R_0 is set to the "1" state (input 5) and the flip-flops R_1 to R_8 are reset to zero by an initialisation signal RST1 available at the outpput of an inverting AND-gate 63. Before each cycle the signal RST1 performs an initialisation. This results in $R_0(Q) = 1$. This corresponds in time to the starting bit STA of the type of cycle shown in figure 5a. The shift register,

clocked by SCL, receives all bits of the address sequence A. This sequence is correctly positioned in the shift register when the logic "1" introduced into R_0 at the time of the initialisation passes on to R_8. The bit A_6 is then stored in R_7 and so on until A_0 which is stored in R_1. The bit W is still equal to zero. The bits A_2, A_1 and A_0 in the sequence will be compared one by one with the bits with the same name in the displayed address with the aid of three exclusive-NOR gates 10, 11 and 12. At its inputs the gate 10 receives A_0 and $Q(R_1)$, the gate 11 receiving A_1 and $Q(R_2)$, and the gate 12 receiving A_2 and $Q(R_3)$. If the address A_0, A_1, A_2 transmitted in the address sequence A is identical to the displayed address, each of the gates 10, 11 and 12 delivers at its output a logic "1". For the embodiment shown in figure 4 it has also been assumed that the bits A_6, A_5, A_4 and A_3 formed an identification flag which is always identical according to the sequence 1001. Consequently, the complete identification of an address sequence is performed by a multiple AND-gate 15 producing at its output an identification signal DVA and receiving at its inputs the outputs $Q(R_7)$ and $Q(R_4)$ and the inverted outputs $P(R_5)$ and $P(R_6)$ and, preferably, $P(R_0)$ as well as the outputs from the gates 10, 11 and 12. The gates 10, 11, 12 and 15 constitute the address sequence identification circuit AIC. On the other hand, the output $Q(R_8)$ produces an end-of-sequence acknowledge logic signal ACN. The signal W which follows A_0 in the address sequence is, by convention, a zero.

The lower part of figure 4, enclosed by a dot-dash line, forms the bus logic BUSL.

It comprises a first control flip-flop DC1 receiving at its D-input the signal ACN available at the output Q of R_8, and at its clock input CK the clock signal SCL, and a second control or pointer flip-flop PNT, whose inverting output P is looped to the input D, its clock input CK being connected to the output Q of the first flip-flop DC1. The pointer PNT has the task of indicating whether an address sequence or a data sequence of a cycle is present.

A third control flip-flop STR and a fourth control flip-flop STP have the task of controlling the start and end of a cycle. The bus indicates the start of a cycle (see figure 5b) by a transition from the level "1" to the level "0" of the signal SDA when the clock signal SCL is at the level "1", and the end of a cycle by a transition from the level "0" to the level "1" of the signal SDA when the clock signal SCL is at the level "1". On the other hand, the information-carrying transitions of the signal SDA take place when the clock signal SCL is at the logic "0" level. The fourth flip-flop STP serves to indicate whether the circuit is in an active or inactive state in relation to the bus. In the latter case, its output initialises the register R.

The third flip-flop STR receives the clock signal SCL at its input D and is clocked at its clock input CK by the signal SDA so as to detect the start of a cycle, whereas the fourth flip-flop STP receives the clock signal SCL at its input D and at its clock input CK the signal SDA, inverted on the basis of SDA, by an inverter 28 so as to detect the end of a cycle. The output Q from the third flip-flop STR is applied to the zero-resetting input R of the pointer PNT and of the fourth flip-flop STP. The third flip-flop STR is reset to zero by the inverse SCL of the clock signal SCL, obtained by the inverter 60 whose output is connected to the input R of STR. The authorisation signal DVA produced at the output of the gate 15 is stored by a flip-flop RS which comprises two inverting AND-gates 66 and 67 looped to each other, which means that the output of the gates 66 and 67 is connected to an input of the gate 67 and the gate 66, respectively. This storage is authorised by an inverting AND-gate 65 and takes place when DVA = 1, SCL = 1 and PNT Q (output of the pointer) = 0. This last condition indicates that the procedure is in the address part of a cycle and avoids any accidental identification in a data sequence. The inverting AND-gate 65 thus receives at its inputs the signals DVA and SCL and the inverted output P from the pointer PNT, and its output is connected

to the other input of the inverting AND-gate 66. The other input of the inverting AND-gate 67 receives a signal STN which is available at the output of an AND-gate 70 whose inputs are connected to the inverting outputs P of the control flip-flops STR and STP, respectively. As long as STN = 1, the signal DVA, which is temporary, will remain stored in the form of the signal ADC at the output of gate 66. The return of STN to the state "0" is equivalent to the deletion of this storage. The initialisation of the flip-flops R_0 to R_8 is effected by the signal RST1 available at the output of an inverting AND-gate 63 which receives at one input the output signal STN from the AND-gate 70 (initialisation when STRQ or STPQ = 1), and at its input a signal RST2 which is available at the output of an inverting AND-gate 44. The latter receives at its three inputs the clock signal SCL, the output Q from the first control flip-flop DC1, and the inverted output P from the pointer PNT.

The flip-flops R_0 to R_8 are initialised when the flip-flop STP stores an inactive state of the circuit (STPP), or between an address sequence and a data sequence (RST2), and also at the start of a cycle (AND-gate 70).

A flip-flop R_9 generates at its output P an inverted authorisation logic signal $\overline{ACK}$, which means that the acknowledgement corresponds to a logic "0". The flip-flop R_9 is clocked at its loading-authorisation input CK by the clock signal SCL and receives at its input D the output from an AND-gate 16, one input of which receives the end-of-sequence acknowledge signal ACN, while its other input receives the stored identification signal ADC.

The loading of the memory flip-flops M_0 to M_7 is obtained by an inverted loading signal $\overline{LDA}$ applied to their loading-authorisation inputs CK. Said loading is obtained by the change-over of $\overline{LDA}$ from the state "1" to the state "0". The signal $\overline{LDA}$ is obtained at the output of an inverting AND-gate 33 whose inputs receive the end-of-sequence acknowledge signal ACN, the stored identification signal ADC,

and the outputs Q of the first flip-flop DC1 and the pointer PNT. Loading is thus authorised when an end of a data sequence has been detected while previously an address sequence corresponding to that displayed by the circuit has been identified.

The fourth flip-flop STP is set to the state "1" by the output from an inverting AND-gate 71 which comprises four inputs. Its first input receives the inverted output P from STP (self-maintenance of the state "1" by STP) and its second input receives the inverse PON of the pulse $\overline{PON}$ produced when a voltage is first applied (STP being placed in the state "1" by the first application of voltage).

Its third input receives the output from an inverting AND-gate 69 whose inputs are SCL, the output Q from DC1, and the output Q from the pointer PNT (setting to the state "1" of STP at the end of the cycle, even in the event of failure of the bus). Its fourth input receives the output from an inverting AND-gate 68 whose inputs are the output Q from DC1 and the inverse $\overline{ADC}$ of ADC which is available at the output of the inverting AND-gate 67 (setting to the state "1" of STP if, at the end of an address sequence, no identification signal has been produced and stored).

It will be noted, finally, that the pulse PON, produced when the circuit is connected to voltage, also resets to zero (input R) the first control flip-flop DC1 and the flip-flop R_9.

The operation of the circuit in the event of an information cycle provided by the bus and corresponding to the displayed address A_0, A_1, A_2 at the inputs S_0, S_1 and S_2 of the circuit will now be described with reference to figures 5a and 5b.

The starting condition STA corresponds to a negative-going edge of SDA with SCL = 1. This is detected by STR whose output STRQ changes to the state "1" upon said negative-going edge of SDA; this change to the state "1" also has the effect of resetting the output STPQ of STP

and of PNT to zero and of making ADC = 0, because STN = 0 and DVA = 0. At that instant the output P of STP changes to the state "1" (STPQ = 0) and RST1 also changes to the state "0". Thanks to the previous initialisation, the flip-flop R_0 is at that instant in the state "1", and the flip-flops R_1 to R_8 are in the state "0". The output Q from STR is reset to zero by the negative-going edge of the starting clock pulse SCL. The bus transmits the address sequence A until R_8 receives, via a shift process, the logic "1" stored in R_0 at the instant of initialisation,resulting in ACN = 1 on the positive-going edge of the 8th clock pulse. The bit A_6 is then located in R_7 and so on until A_0 is located in R_1, and W = 0 (by convention) in R0. As the address transmitted by the bus corresponds to that displayed by the circuit, the AND-gate 15 transmits a DVA = 1, which is authorised by the gate 65 because the output P from the pointer PNT is in the "1" state, starting from the 8th clock pulse SCL. ADC changes to the state "1". At the next clock pulse, DVA again changes to "0" but ADC remains at the level "1" because the outputs P from STR and STP are both at the level "1", so that STN = 1.

The simultaneous presence of ACN and ADC, detected by the gate 16, produces at the output P of R_9 the signal $\overline{ACK}$ on the negative-going edge of the 8th clock pulse, the latter being erased by the negative-going edge of the 9th clock pulse because the output of the gate 16 at that instant is zero; ACK is, therefore, a signal identical to ACN but delayed by half a clock pulse. It is therefore present when the clock is at its high level for the ninth pulse, hence in the space provided for that purpose between the address sequence A and the data sequence D. The way in which the bus system deals with this signal ACK is described in the cited European patent.

The signal ACN also sets the flip-flop DC1 (DC1Q) to the state "1" upon the negative-going edge of the 8th clock pulse. DC1Q changes back to zero on the negative-going edge of the 9th clock pulse,because ACN has changed back to zero

on the positive-going edge of this clock pulse. This negative-going edge of DC1Q causes the output of the pointer PNT to change to the state "1" (PNTQ = 1). The pointer PNT thus notes the fact that the data sequence D of the cycle is now about to be received.

For the duration of the 9th clock pulse SCL = 1, DC1Q = 1 and PNTQ = 0, hence RST1 = 1, which causes the initialisation of the register R, R_0 being in the state "1" and R_1 to R_8 in the state "0".

The arrival of the data sequence thus occurs with ADC = 1 and PNT = 1.

As for the address sequence A, the logic "1" stored in R_0 at the time of initialisation of transferred by shifting to the register R_8 so as to produce an ACN = 1 on the positive-going edge of the 8th clock pulse in the sequence D, ACN returning to 0 on the positive-going edge of the next clock pulse. The bit D_7 is then stored in R_7 and so on until D_0 is stored in R_0. Thus, DC1Q = 1 on the negative-going edge of the 8th clock pulse in the sequence D, DC1Q returning to 0 in the negative-going edge of the next clock pulse.

Between the negative-going edge of the 8th clock pulse of the sequence D and the next positive-going edge, there occurs simultaneously DC1Q = PNTQ = ACN = ADC = 1, hence LDA = 1 for the duration of half a clock pulse. The contents of R_0 to R_7 are transferred in parallel into the memory flip-flops M_0 to M_7. In particular, M_0 receives from R_0 the bit D_0, a "0" in principle making it possible to enable the command OFF at the input of gate 90 (figure 3).

The open or closed position of switches I_1 to I_9 is then altered in accordance with the new contents of the memories M_0 to M_7 if the command OFF is inactive or inhibited by D_0.

As for the address sequence A, R_9 produces a signal ACK = 1 between the negative-going edge of the 8th clock pulse and that of the 9th clock pulse in the sequence D. The change to zero of DC1Q on the negative-going edge

of the 9th clock pulse in the sequence D causes the resetting to zero of the pointer (PNPQ = 0). At the end of the cycle the bus transmits a stop signal formed by a positive-going edge SDA on a clock signal SCL at the level "1", resulting in STPQ = 1, hence STN = 0, therefore ADC = 0, and RST1 = 1 (initialisation of the register R). An improved version of the interface circuit according to the invention caters for the case in which the bus might fail with regard to the production of such a stop signal.

To that end, advantage is taken of the fact that, in the terminal phase of a cycle, a transitory and characteristic situation occurs in which DC1Q = PNTQ = 1. The gate 69 then switches the flip-flop STP to the state "1" (STPQ = 1) on the positive-going edge of the 9th clock pulse in the sequence D.

The stop signal delivered by the bus consists of a positive-going edge on the signal SDA, while SCL = 1 on the 10th clock pulse in the sequence D. It is detected by the flip-flop STP which changes to the state "1" (STPQ =1) in the circuit versions not incorporating a gate 69. This is represented by dashes in figure 5b, and also applies to the signals ADC and RST1.

The invention is not restricted to the embodiments described or shown. Thus, the inverted output P of the control flip-flop STR may be connected directly to the corresponding input of the inverting AND-gate 63, the use of the AND-gate 70 nevertheless giving the advantage of initialising the register REG in the event of unwanted coincidence corresponding to the start of a cycle, while such a cycle is actually in progress. On the other hand, all D-type flip-flops in the description have been chosen of the type which load on a negative-going edge applied to their clock input CK. It will be obvious that flip-flops which load on a positive-going edge could also be employed.

CLAIMS:

1. A slave-type interface circuit intended for the switching of a plurality of channels and comprising logic inputs capable of receiving information signals and clock signals supplied by a bus, a register (REG) for receiving the signals (SDA, SCL) received from the bus via the logic inputs (L_1, L_2), and a memory (M) for the data received by the register (REG), characterized in that the circuit comprises:

- a first decoder (CDEC) which controls the switching of the channels on the basis of logic signals stored in the memory (M),

- a plurality of addressing inputs (S_0, S_1, S_2), each of which can be set to a logic level so as to generate a displayed address (A_0, A_1, A_2),

- a second decoder (FDEC) which signals whether said displayed address corresponds to a first operating state of the circuit, on line with said bus, or to a second state, off-line, on the basis of logic control signals applied directly to the logic inputs (L_1, L_2),

- a pointer block (SBL) which directs the signals available at the logic inputs (L_1, L_2) either to said register (R) if said displayed address corresponds to operation on-line with said bus (first state), or to the first decoder (CDEC) so as to control directly the switching of at least some of said channels (I_1, ..., I_9) on the basis of said logic control signals if said displayed address corresponds to off-line operation (second state),

- a blocking circuit (BC) which comprises a first series of logic gates (60 - 87) which inhibit the passage of the logic signals stored in the memory towards said first decoder (CDEC) when the circuit is in its second state of operation (off-line).

2. An interface circuit as claimed in Claim 1, characterized in that it includes a second series of logic gates (101, 109) which inhibit the passage of the logic signals available at the output of the first decoder (CDEC)

towards the channels to be switched in the presence of a logic stop signal from the interface circuit (OFF).

3. An interface circuit as claimed in any one of the preceding Claims, characterized in that it includes a pulse generator (PG) which, when voltage is applied to the circuit, generates a pulse (PON) which causes the initialisation of the memory (M) and of at least a part of the register (REG).

4. An interface circuit as claimed in anyone of the preceding Claims, characterized in that said register is a shift register (R_0, R_7) which receives, in series at an input located at an upstream end, series logic information (SDA) supplied by the bus in the form of cycles, each of which presents at least one sequence synchronized by a clock signal (SCL) supplied by the bus when the circuit is in its first state, and produces, at an output (QR_8) located at a downstream end, an end-of-sequence acknowledge signal (ACN) which is applied to the input(D) of a first control flip-flop (DC_1) which is synchronized (CK) by said clock signal (SCL) and whose output (Q) synchronizes (CK) a second control or pointer flip-flop (PNT) whose inverted output (P) is looped to its input (D), in that the presence of a logic level "1" at the output (Q) of the first flip-flop (DC_1) and at the output (P) of the pointer (PNT) enables initialisation of said register, for example by setting its first flip-flop to the state "1" and resetting of its other flip-flops to zero, in that it comprises a logic comparator (10, 11, 12) which identifies, in a given part of an address sequence, series logic information supplied by the bus if said address sequence corresponds to said displayed address, and which produces an identification logic signal (DVA), in that it comprises a circuit (66, 67) for storing said identification signal (DVA), and in that it comprises an AND-gate (33) enabling the loading into the memory (M) of the data from the register (R) when the output (Q) of the first control flip-flop (DC1) and of the pointer (PNT) are in the logic state "1", when a stored identification signal (ADC) is present and also a said end-of-sequence acknowledge signal

(ACN), which thus corresponds to the end of a data sequence which follows said address sequence.

5. An interface circuit as claimed in Claim 4, characterized in that it is designed so that it produces said initialisation by setting the first flip-flop (R_0) of the register (R) to state "0" and by the resetting of its other flip-flops (R_1 ... R_8) to zero when the output of the first control flip-flop (DC_1) is in the state "1" and the output (Q) of the pointer (PNT) in the state zero.

6. An interface circuit as claimed in one of the Claims 4 or 5, characterized in that it comprises a third (STR) and a fourth (STP) control flip-flop which, when the interface circuit is in its first state, receive at their input (D) the clock signal (SCL) and are synchronized (CK) one by the information signal (SDA) and the other by its inverse ($\overline{SDA}$) so that the output (Q) of the third flip-flop (STR) is set to the state "1" by a first type of coincidence between the clock (SCL) and information (SDA) signals corresponding to the start of a said cycle, and the fourth flip-flop (STP) is set to the state "1" by a second type of coincidence between the clock (SCL) and information (SDA) signals corresponding to the end of a cycle, the third flip-flop (STR) being reset to zero by the inverse ($\overline{SCL}$) of the clock signal (SCL) and the fourth flip-flop (STP) by the output (Q) of the third flip-flop (STR), the output (Q) of the fourth flip-flop (STP) being designed so as to produce, when in the logic state "1", the setting to the state "1" of the first flip-flop (R) of the register (R) and the resetting to zero of its other flip-flops (R_1 ... R_8).

7. An interface circuit as claimed in Claim 6, characterized in that the fourth flip-flop (STP) has its output set to the logic state "1" when the stored identification signal (ADC) is in the logic state "0" and when the first flip-flop (DC_1) has its output in the logic state "1".

8. An interface circuit as claimed in any one of the Claims 4 to 7, characterized in that the circuit for storing

the identification signal (DVA) comprises two inverting AND-gates (66, 67), the output of one of which is looped to one of the inputs of the other and vice versa, the other input of the first of the two inverting AND-gates (66) receiving the inverse ($\overline{DVA}$) of the identification signal (DVA), be it only if the pointer (PNT) has its input in the logic state "0" and if the clock signal is in the logic state "1", the other input of the second of the two inverting AND-gates (67) receiving the logic product of the inverses (P) of the outputs (Q) of the third (STR) and the fourth (STP) flip-flop so as to permit the storage (ADC) of the identification signal (DVA) during each cycle.

9. An interface circuit as claimed in Claim 3, in combination with any one of the Claims 6 to 8, characterized in that the fourth flip-flop (STP) also has its output set to the logic state "1" by said pulse (PON), its output, however, being self-maintained when it is in the state "1".

10. An interface circuit as claimed in anyone of the Claims 7 to 9, characterizedin that it is designed so that the fourth flip-flop (STP) also has its output set to the logic state "1" when the outputs from the first (DC_1) and the second flip-flop (PNT) and the clock signal (SCL) all have the logic level "1".

11. An interface circuit as claimed in any one of the Claims 4 to 10, characterized in that it includes an acknowledge flip-flop (R_9) which is synchronized by the clock signal (SCL) and which receives at its input the logic product of the acknowledge signal (ACN) and the stored identification signal (ADC), and produces at its output an authorisation signal (ACK) directed towards the bus.

12. An interface circuit as claimed in the Claims 3 and 11 taken together, characterized in that it is designed so that said pulse also produces the resetting to zero of the acknowledge flip-flop (R_9).

13. An interface circuit as claimed in the Claims 3, 4 and 6 taken together, characterized in that it is designed so that said pulse also produces the resetting to zero of

the first control flip-flop (DC_1) and the setting to the state "1" of the fourth control flip-flop (STP).

14. An interface circuit as claimed in any one of the Claims 6 to 13, characterized in that it is designed so that it produces an initialization of the register (REG) when at least one of the third (STR) and fourth (STP) control flip-flops has its output (Q) in the state "1".

15. An interface circuit as claimed in any one of the preceding Claims, characterized in that it comprises two logic inputs (L_1, L2) which are capable of being connected, in the first state of the circuit, to a bus which supplies an information signal (SCL) and a clock signal (SDA), and in that the memory (M) comprises a plurality of memory flop-flops (M_0 ... M_7), in that the first series of logic gates (80 ... 87) of the pointer circuit (SBL) comprises a third and a fourth inverting AND-gate (81, 82), one input of each of which is connected to the output of a first and second said memory flip-flop (M_1, M_2), respectively, and the other input of which receives from the second decoder (FDEC) a signal ($\overline{CS}$) having a logic level "1" in the first state of the circuit and "0" in the second state of the circuit, the output from the third and the fourth inverting AND-gate being connected to an input of a fifth (88) and a sixth (89) inverting AND-gate, respectively, whose output is connected to the first decoder (CDEC), and in that the other input of the fifth (88) and the sixth (89) inverting AND-gate receives the output from an eighth (78) and a ninth (72) inverting AND-gate, respectively, enabling the signals (SCL, DSA) available at the logic inputs (L_1, L_2) when the circuit is in its second state.

16. An interface circuit as claimed in Claim 2 in combination with any one of the Claims 4 to 15, characterized in that said first series of logic gates (80 ... 87) includes a tenth inverting AND-gate (80), one input of which is connected to the output of a third memory flip-flop (M_0), its other input receiving from the second decocer (FDEC) a signal ($\overline{CS}$) with a logic level "1" in the first state of

the circuit and "0" in the second state, and whose output is connected to the input of a stop enabling gate (90) so that the logic stop signal (OFF) is enabled or not in accordance with the contents of the third memory flip-flop (M_0) when the circuit is in its first state and is always enabled in the second state.