



Electronic
components
and materials

PHILIPS

Serial I/O section
MAB8400 family

Purchase of Philips' I²C components conveys a license under the Philips' I²C patent to use the components in the I²C-system provided the system conforms to the I²C specifications defined by Philips.

Archief:
Ir. A.P.M. Moelands

CONTENTS LIST SERIAL I/O

1.0 INTRODUCTION

2.0 SERIAL INPUT/OUTPUT

3.0 SERIAL BUS STRUCTURE

4.0 SERIAL DATA TRANSFER

5.0 SERIAL DATA FORMATS

6.0 OPERATING MODES OF THE SERIAL I/O INTERFACE

6.1 Master transmitter

6.2 Master receiver

6.3 Slave receiver

6.4 Slave transmitter

7.0 SERIAL I/O INTERFACE

7.1 Data shift register S0

7.2 Address register S0'

7.3 Status register S1

7.4 Clock control register S2

8.0 SERIAL I/O OPERATIONS

8.1 Arbitration procedures

8.2 Clock synchronisation

9.0 PROGRAMMING

9.1 State after RESET

9.2 Initialisation procedure

9.2.1 Clock control register S2

9.2.2 Address register S0'

9.2.3 Status register S1

9.2.4 Enable/disable serial I/O interrupt

9.2.5 Example of an initialisation procedure

9.3 Generation of a 'start' condition and the first byte

9.4 Software responses after transmission or reception of a byte

9.5 Generation of the 'stop' condition

9.6 Generation of a repeated 'start' condition

9.7 Generation of the 'stop' condition by a 'master receiver'

9.8 Flow charts for the 'master' and 'slave' modes of the SIO interface

9.9 Equate list for serial I/O control bits

MEMORANDUM FOR THE DIRECTOR

DATE: 10/10/68

TO: DIRECTOR, FBI

FROM: SAC, NEW YORK

SUBJECT: [Illegible]

RE: [Illegible]

1. [Illegible]

2. [Illegible]

3. [Illegible]

4. [Illegible]

5. [Illegible]

6. [Illegible]

7. [Illegible]

8. [Illegible]

9. [Illegible]

10. [Illegible]

11. [Illegible]

12. [Illegible]

13. [Illegible]

14. [Illegible]

15. [Illegible]

16. [Illegible]

17. [Illegible]

18. [Illegible]

19. [Illegible]

20. [Illegible]

1.0 INTRODUCTION

The increasing demand for microcomputers in domestic and industrial applications where control is needed rather than computation, has led to the modification of some microcomputer hardware functions. One of these modifications has been the introduction of serial input/output.

Normally, serial data transfer imposes a heavy processing load on microcomputers, with the serial data bus having to be regularly monitored for information. To overcome this problem the Serial I/O (SIO) interface was introduced. A SIO interface can detect, receive and convert the serial data stream to parallel format without interrupting current program execution.

The use of serial data communication between devices greatly reduces the number of connections, leading to simpler circuit layouts and economies in both connector size and printed circuit board area. A hardware serial I/O on a microcomputer allows the interconnection of any number of devices by the two-line serial bus.

To demonstrate how serial communication works, consider two microcomputers connected via an I²C (Inter-IC) bus to a serial memory and a display (Fig.1). In a typical system such as this, the device which controls the transfer of messages is called a 'master', the device (or devices) controlled by the master is called a 'slave'. So, in our example, the master would be either of the two microcomputers, with the serial memory and display acting as slaves.

A master generates timing when in control of the bus. Only one master can eventually control the I²C bus at one time, and can act as either a transmitter or receiver. In the same way, the slave under control of the master can also transmit or receive.

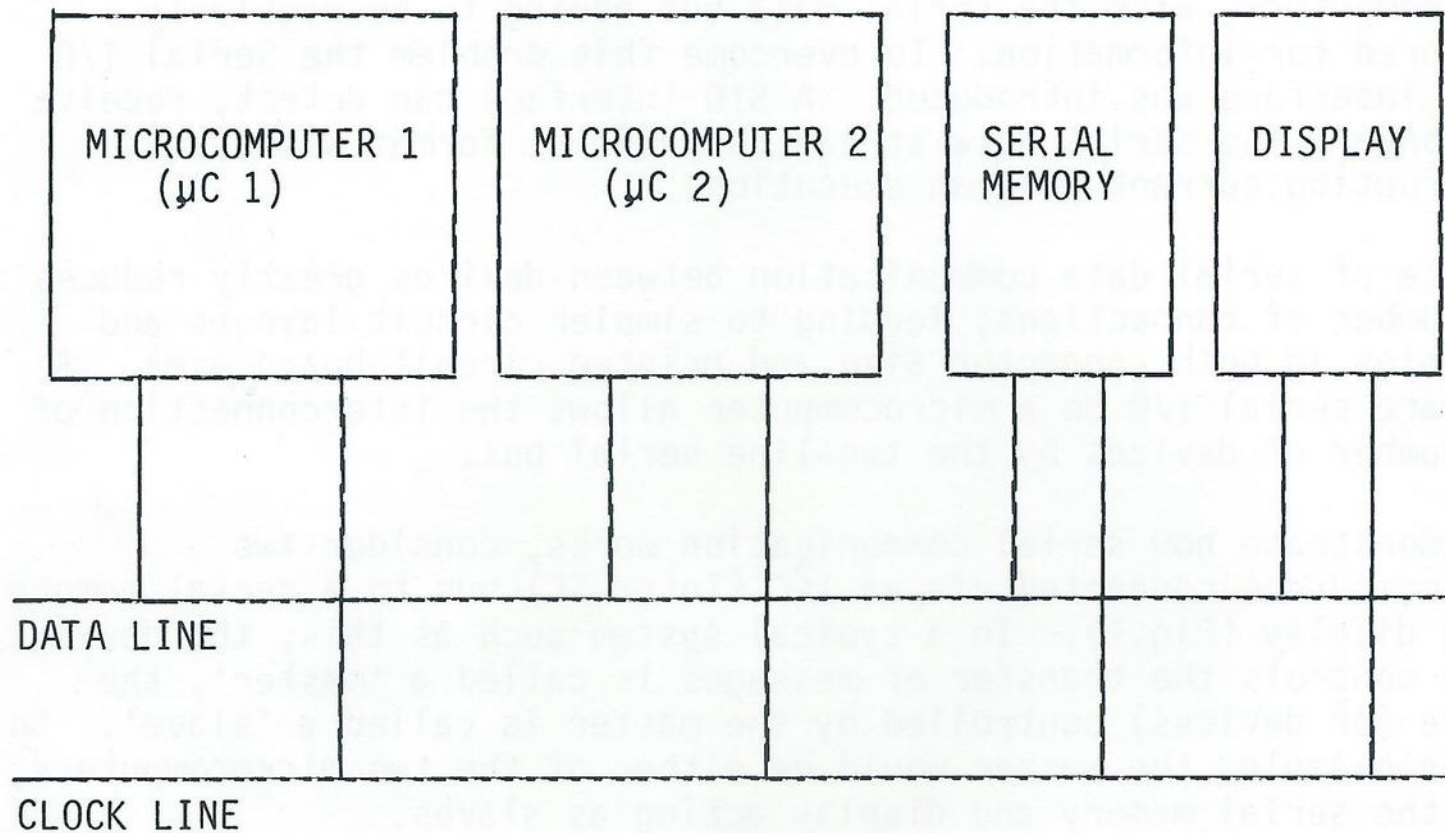
An important feature of the I²C bus is that it is a true multimaster bus, which means that more than one master can try to control the bus at the same time without corrupting the message. During transmission, arbitration takes place among the masters to ensure this. Therefore there is no central master in the bus structure and any master transceiver if it loses the arbitration can be addressed as a slave by the winning master.

To explain what is meant by slave transmitters and receivers and master receivers and transmitters consider Fig.1.

If either of the microcomputers is sending information to the serial memory then the memory is a 'slave receiver' and the microcomputer is the 'master transmitter'.

If a microcomputer is receiving information from the serial memory then the memory is a 'slave transmitter' and the microcomputer is the 'master receiver', and so on.

Of course, if after arbitration the losing microcomputer is addressed by the winner, then the loser is considered a 'slave transmitter' or 'slave receiver' and the winner, a 'master transmitter' or 'master receiver'.



	MASTER	SLAVE	RECEIVER	TRANSMITTER
MICROCOMPUTER 1	* *		*	*
		*1 *1	*	*
MICROCOMPUTER 2	* *		*	*
		*2 *2	*	*
SERIAL MEMORY		* *	*	*
DISPLAY		*	*	

- 1 - when μC 2 controls the bus
- 2 - when μC 1 controls the bus

Fig.1 Example of typical serial I/O connection.

2.0 SERIAL INPUT/OUTPUT

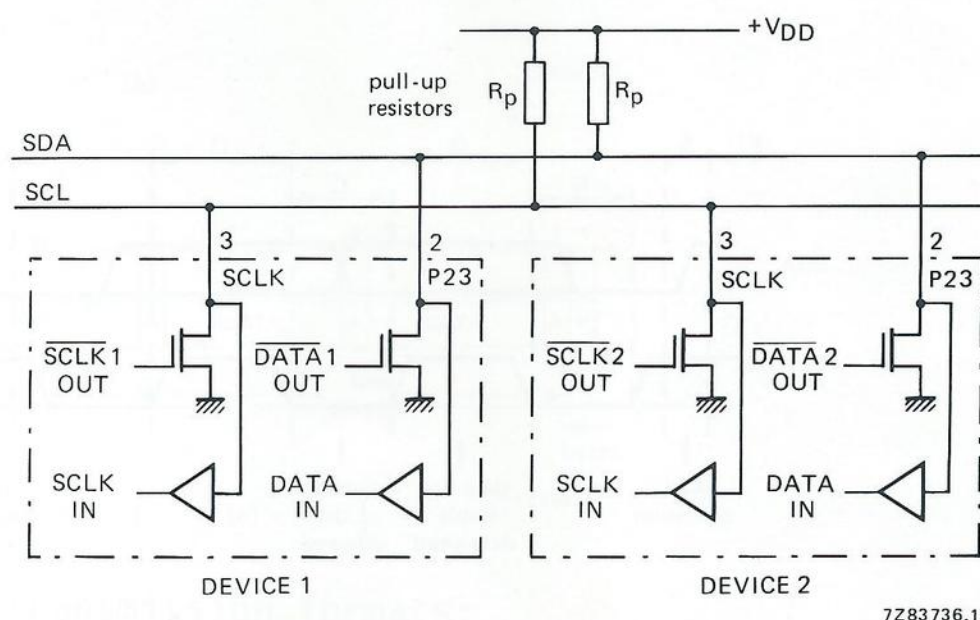
The ability of any two devices to communicate, without the slightest interruption to the operation of any other devices on the bus, is an outstanding attribute of the serial I/O system. Communication is achieved by allocating a specific 7-bit address to each device and providing a system whereby a device reacts only to messages prefixed with its own address or the 'general call' address (H'00').

The address recognition procedure is performed by the interface hardware so that operation of the microcomputer need only be interrupted when a valid address has been received. This results in a significant saving of processing time and memory space compared with a conventional microcomputer employing a software serial interface. When the addressing facility is not required, for instance in a system with only two microcomputers, direct data transfer without addressing can be performed i.e. the receiving device reacts after each received byte from the serial bus.

The serial input/output system also provides simple mechanisms for implementation of master/slave operation of devices, acknowledgement of data reception and an asymmetrical clock signal for use with microcomputer systems without on-chip serial I/O. When multimaster systems are used, an arbitration procedure is automatically invoked, to prevent two or more devices from continuing simultaneous transmission.

3.0 SERIAL BUS STRUCTURE

The serial bus shown in Fig.2 consists of two bidirectional lines; a serial data line (SDA) and a serial clock line (SCL). Pin 2 (SERIAL DATA I/O) of each microcomputer is connected to the SDA line, pin 3 (SCLK) is connected to the SCL line. A pull-up resistor R_p is connected between each line and $+V_{DD}$ so that the open-drain outputs



Archief
Ir. A.P.M. Moelands

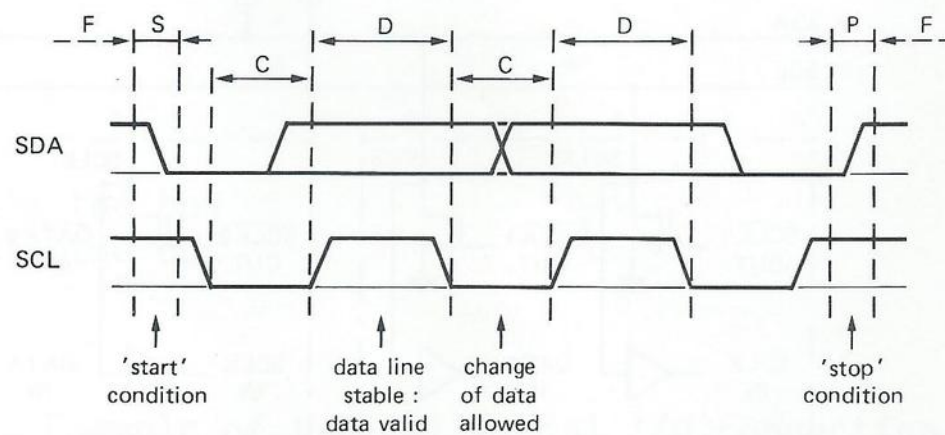
Fig.2 Connection of two microcomputers to the serial bus.

of the microcomputers connected to the bus, together form a wired-AND function on each line. A HIGH level is logic 1, a LOW level is logic 0. Each microcomputer can be software programmed to function as a 'transmitter' or a 'receiver' operating in either a 'master' or a 'slave' mode.

4.0 SERIAL DATA TRANSFER

Fig.3 shows the serial data transfer sequence for the SIO interface. The following conditions can be distinguished:

- F (free): The bus is free; data line SDA and clock line SCL are both HIGH
- S (start): A data transfer commences with a 'start' condition during which the level of data line SDA changes from HIGH to LOW, and clock line SCL remains HIGH. The bus is now busy.
- C (change): During the LOW period of the clock, the data bit to be transmitted is applied to data line SDA. The level on the SDA line may therefore change during this period
- D (data): One data bit is transmitted during the HIGH period of the clock. The level (bit state) on line SDA must remain stable during this period to prevent it being interpreted as a 'start' or 'stop' condition
- P (stop): A data transfer concludes with a 'stop' condition during which the level on data line SDA changes from LOW to HIGH and clock line SCL remains HIGH. The bus is now free again



7Z83731.1

Fig.3 Sequence of bit transfer on the serial bus.

5.0 SERIAL DATA FORMATS

The three data formats that can be serviced by the serial I/O are shown in Fig.4. The first byte (SLA and $R/\bar{W}$) after a 'start' condition (S) always consists of 8 bits. By programming an internal flag, an acknowledgement bit (A) may be inserted after each byte.

In the first two formats (addressing formats (a) and (b)), byte SLA is the address of the 'slave' to be selected by a 'master'. The least significant bit $R/\bar{W}$ of the address byte indicates the direction of transmission of the following data bytes. If $R/\bar{W}$ is LOW, the 'master' will write (transmit) data into the selected 'slave'; if it is HIGH, the 'master' will read (receive) data out of the 'slave'. In the last case the direction of a transfer changes after transmission of the first byte SLA.

In the free data format (c), the direction of transmission remains constant throughout the data transfer.

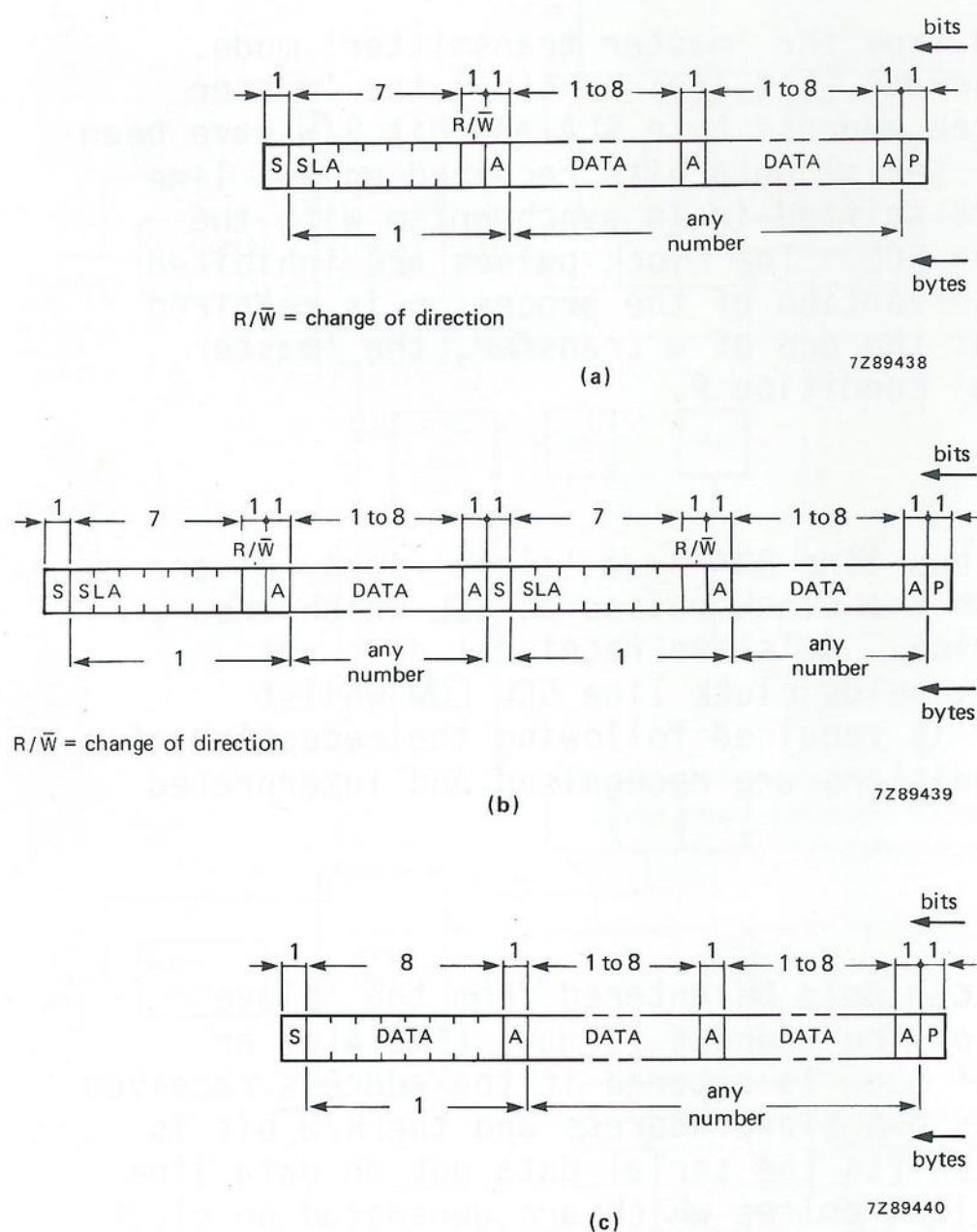


Fig.4 Serial bus transmission formats:

- (a) addressing format (ALS = 0)
- (b) addressing format with repeated 'start' condition (ALS = 0)
- (c) free data format (ALS = 1).

6.0 OPERATING MODES OF THE SERIAL I/O INTERFACE

The serial I/O can function in the following four basic operating modes to service all facilities of the I²C (Inter IC) bus, P/C bus or point to point connections:

- 'master transmitter'
- 'master receiver'
- 'slave receiver'
- 'slave transmitter'

6.1 Master transmitter

In this mode, data assembled in one of the previously described data formats is shifted-out on the serial data line in synchronism with the self-generated clock pulses on SCL. The clock pulses are inhibited and SCL held LOW when the intervention of the processor is required after a byte has been transmitted.

6.2 Master receiver

This mode can only be entered from the 'master transmitter' mode. With either of the address formats (Fig.4(a) or 4(b)), the 'master receiver' mode is entered after address byte SLA and bit R/W have been transmitted if R/W was HIGH. Serial data bits received on bus line SDA by a 'master receiver' are shifted-in in synchronism with the self-generated clock pulses on SCL. The clock pulses are inhibited and SCL held LOW when the intervention of the processor is required after reception of a byte. At the end of a transfer, the 'master receiver' generates the 'stop' condition P.

6.3 Slave receiver

Serial data bits received on bus line SDA by a 'slave receiver' are shifted-in in synchronism with the clock pulses at SCL which are generated by the 'master' device. A 'slave receiver' does not generate clock pulses, it only holds clock line SCL LOW whilst intervention of the processor is required following the reception of a byte. 'Start' and 'stop' conditions are recognised and interpreted accordingly.

6.4 Slave transmitter

The 'slave transmitter' mode can only be entered from the 'slave receiver' mode. With either of the address formats (Fig.4(a) or 4(b)), the 'slave transmitter' mode is entered if the address received in byte SLA is the same as its own slave address and the R/W bit is HIGH. A 'slave transmitter' shifts the serial data out on data line SDA in synchronism with the clock pulses which are generated on clock line SCL by the 'master' device. A 'slave transmitter' does not generate clock pulses, but only holds clock line SCL LOW whilst intervention of the processor is required after transmission of a byte. 'Start' and 'stop' conditions are recognised and interpreted accordingly.

A block diagram of the SIO interface is shown in Fig.5. The clock line of the serial bus has exclusive use of pin 3, while the data line shares pin 2 with I/O signal P23 of port 2. Consequently, only three I/O lines are available for port 2 when the SIO interface is enabled.

- data shift register S0
- address register S0'
- status register S1
- clock control register S2.



Fig.5 Block diagram of the MAB8400 SIO interface.

Eight instructions are available so that these four registers can be written into or read via the internal microcomputer bus. All the bits in the registers, except the PIN (pending interrupt not) bit in the status register are set to 0 by a RESET. The PIN bit is set to 1 by a RESET. The address comparator shown in Fig.5 can instruct the interrupt logic block (if enabled) to generate an SIO interrupt request to the microcomputer.

Communications with the four registers and the function of the address comparator will now be described in more detail.

7.1 Data shift register S0

S0 is the data shift register used to perform the conversion between serial and parallel data format. Data to be transmitted is loaded in S0 by the microcomputer and shifted out serially, most significant bit first. Data received on the serial bus is shifted into S0, most significant bit first. After transmission of a complete byte or reception of a complete data byte, the specific address, or the general address, a pending interrupt is generated.

To address this 8-bit data shift register, the ESO (enable serial output) bit in status register must be a 1. The write instruction MOV S0,A or MOV S0,# data causes data shift register S0 to be parallel-loaded via the internal bus with the data for transmission. During transmission, the contents of the register are shifted out, most significant bit first, onto data bus line SDA.

During reception, the serial data is shifted into the data shift register S0. The last received data bit present on data line SDA is shifted into the least-significant bit position in the data shift register. In the acknowledgement mode, acknowledgement bit ACK is not shifted into register S0 but into the LRB (last received bit) position of S1. The read instruction MOV A,S0 causes the contents of register S0 to be parallel-loaded into the accumulator via the internal bus.

7.2 Address register S0'

The address register contains the 7-bit address back-up latches and the ALS flag, as shown in Fig.6. The address latches hold the slave address allocated to the device, while the ALS (Always Selected) flag is used to enable/disable the address recognition mode.

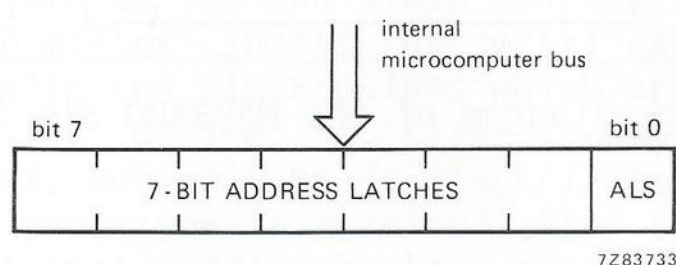


Fig.6 Address register in the SIO interface.

- ALS=1, The address recognition has been disabled. The SIO interface will respond to the free data formats i.e it reacts after each received byte from the serial bus.
- ALS=0 The address recognition has been enabled. The SIO interface will respond to the addressing format i.e it reacts only to messages containing its own slave address or general address (H'00'). In this mode, the least significant bit of the first byte received performs the function of read/write command (write=0).

The address register can only be loaded from the internal bus of the microcomputer when the SIO interface is disabled (ESO=0). Under this condition, a MOV S0,A instruction will move the contents of the accumulator to the address register instead of S0, as would normally be the case.

7.3 Status register S1

The status word in status register S1 is shown in Fig.7. It contains all information regarding the status of the SIO interface. Read instruction MOV A,S1 causes the status word to be read out via the internal bus. To control the SIO interface, information is written into the status register with the write instruction MOV S1,A or MOV S1,# data. There are two latches at each of the four least significant bit positions in the status register so that two bits can be stored at each position. The four least-significant bits ESO, BC2, BC1, and BC0 are control bits that can only be written. The four low order bits AL, AAS, AD0 and LRB are status bits that can only be read. The four high order bits are MST, TRX, BB and PIN which can be both written and read. The function of each bit in the status register will now be described.

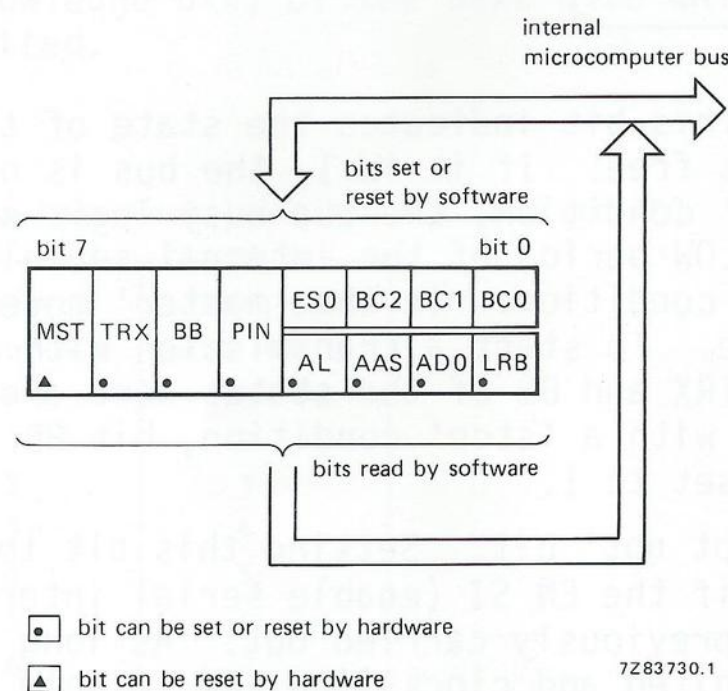


Fig.7 Bit allocation of the status word in register S1.

MST: 'master' bit. If this bit is 1, the SIO interface is in the 'master' mode and it generates the clock pulses on clock line SCL for timing the transmission or reception of the serial data. If the MST bit is a 0, the SIO interface is in the 'slave' mode and the clock pulses are received from the 'master' device on clock line SCL. The MST bit is set to 0 by the hardware if an arbitration* of this master device is lost. When a data transfer has been completed, a 'stop' condition is generated by the 'master' device and the hardware resets the MST bit of this master device to 0.

TRX: 'transmitter' bit. If this bit is 1, the SIO interface is in the 'transmitter' mode and the data in register S0 is shifted out on data line SDA in synchronism with the pulses on clock line SCL. If TRX is 0, the SIO interface is in the 'receiver' mode and the data on bus line SDA is shifted into data register S0 in synchronism with the pulses on clock line SCL. The TRX bit is set to 0 by the hardware if an arbitration* is lost. When a data transfer has been completed, a 'stop' condition is generated by the 'master' device and the hardware resets the TRX bit of this master device to 0. In either of the addressing formats (Fig.4(a) or 4(b)), the state of the TRX bit is set by the hardware in accordance with the state of read/write direction bit $\overline{R}/W$. The operating modes set by the MST and TRX bits of the status word are summarised in Table 1.

Table 1 Operating modes set by bits MST and TRX

MST	TRX	operating code
0	0	'slave receiver'
1	0	'master receiver'
0	1	'slave transmitter'
1	1	'master transmitter'

BB: 'bus busy' bit. This bit indicates the state of the serial bus. If it is 0, the bus is free. If it is 1, the bus is occupied. On reception of a 'start' condition, the bus busy logic sets BB to 1. BB is reset to 0 one LOW period of the internal serial clock after reception of a 'stop' condition. In the 'master' mode, BB is controlled by software. To start a transmission with a 'start' condition, bits MST, TRX and BB of the status word are set to 1. To finish a transmission with a 'stop' condition, bit BB is set to 0 and bits MST and TRX are set to 1.

PIN: 'pending interrupt not' bit. Setting this bit to 0 initiates a serial I/O interrupt if the EN SI (enable serial interrupt) instruction has been previously carried out. As long as PIN is 0, the clock pulses are inhibited and clock line SCL is held LOW. The PIN bit is only set to 0 when:

* The arbitration procedure is described in section 8.1.

- A complete byte has been transmitted (even if arbitration has been lost).
- The address comparator has recognised its own slave address in one of the addressing formats shown in Fig.4(a) or 4(b), or the general address of all (8) zeros has been received (ALS flag = 0).
- Another byte has been received after an address has been previously recognised.
- A byte has been received in the free data format shown in Fig.4(c) (ALS flag = 1).

The PIN bit is reset to 1 by:

- Read instructions MOV A,S0 or write instruction MOV S0,A or MOV S0,# data.
- Loading a 1 into the PIN bit position in status register S1 as a result of write instruction MOV S1,A or MOV S1,# data.

ESO: 'enable serial output' bit. When this bit is 1, the SIO interface is enabled and can receive or transmit on serial data line SDA, in synchronism with the pulses on clock line SCL. When the ESO bit is 0, the SIO interface is disabled, pin 2 reverts to its P23 (I/O line of port 2) function and pin 3 (SCLK) is in the high impedance state. Whilst ESO is 0, a 7-bit slave address, plus ALS bit, can be loaded into address register S0' with write instruction MOV S0,A or MOV S0,# data. As long as ESO remains 0, PIN is held HIGH, AL is held LOW by hardware and the contents of data shift register S0 are not affected by a write instruction to S0.

BC2, BC1 and BC0: 'bit count'. As shown in Table 2, the binary-coded number preset in bit position BC2, BC1 and BC0 is the number of bits (excluding the acknowledge bit) of the next byte which are yet to be received or transmitted.

Table 2 Binary numbers in bit-count locations BC2, BC1 and BC0

BC2	BC1	BC0	bits/byte without ACK	bits/byte with ACK
0	0	1	1	2
0	1	0	2	3
0	1	1	3	4
1	0	0	4	5
1	0	1	5	6
1	1	0	6	7
1	1	1	7	8
0	0	0	8	9

A 'start' condition resets the bit count to 000, so the first byte to be received or transmitted will always consist of eight bits excluding the acknowledge bit. After each complete byte has been received or transmitted, the bit count has reduced to 000.

AL: 'arbitration lost' bit. This bit is set to 1 when the arbitration logic senses the SIO interface has lost an arbitration* when it is in the 'master transmitter' mode. It is also set to 1 if an attempt is made to occupy the bus by writing to S1 while the BB has already been set to 1. After a byte has been transmitted, PIN is set to 0 and the status word in register S1 indicates in which mode the SIO has been addressed. AL is reset to 0 by read instruction MOV A,S0 or any of the write instructions MOV S0,A MOV S0,# data MOV S1,A or MOV S1,# data.

AAS: 'addressed as slave' bit. This bit is set to 1 when the address comparator has recognised its own slave address or an address of all (8) zeros. The AAS flag is also set if the first byte has been received in the free data format (ALS = 1). The AAS bit is reset to 0 by the read instruction MOV A,S0 or write instruction MOV S0,A or MOV S0,# data. The device remains selected until the STOP condition.

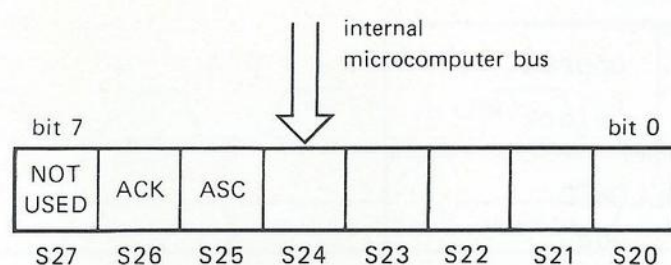
ADO: 'address zero' bit. This bit is set to 1 if the address comparator detects the address of all (8) zeros. The ADO bit is reset to 0 when a 'start' or 'stop' condition is detected.

LRB. 'last received bit'. If a byte is transmitted with an acknowledge (ACK) bit, the LRB bit position in status register S1 of the 'transmitter' contains the acknowledgement of the 'receiver'. When LRB is 0, reception of the transmission has been acknowledged. If a transmission does not include an acknowledgement bit, the last bit of the transmitted or received byte will be in the LRB position.

7.4 Clock control register S2

This is an 8-bit write-only register with bit positions as shown in Fig.8. The seven bits S20 to S26 can be parallel-loaded into the register with write instruction MOV S2,A or MOV S2,# data. The eighth bit position (S27) is not used. Bits S20 to S24 control the frequency of the clock pulses for the SIO interface. Bit S25 (ASC) determines whether the high/low ratio of the clock pulses is 1:1 or 3:1. Bit S26 (ACK) determines whether or not the device is operating in the acknowledgement mode.

* The arbitration procedure is described in section 8.1



7Z83732.1

Fig.8 Bit allocation in clock control register S2.

S20 to S24: clock frequency control bits. These bits form a binary-coded frequency control number (0 to 31) which determines the factor by which the microcomputer clock frequency will be divided to obtain the required frequency of pulses on clock line SCL. Table 3 lists the divisors together with hexadecimal equivalents of the frequency control numbers, and shows the resulting frequency of the clock pulses on SCL if the microcomputer clock is controlled by a 4,43 MHz crystal.

ASC: 'asymmetrical clock' bit at S23. If this bit is 1, the clock pulses on SCL have a high/low ratio of 3:1 and the clock frequency control number may only be HEX'18', HEX'19', HEX'1A' or HEX'1B' which gives a clock-pulse frequency range of 1,4 kHz to 2,3 kHz. This is the low-speed mode of the SIO interface. For example, if the clock-frequency control number HEX'19' is in bit positions S20 to S24, the frequency of the pulses on clock line SCL will be about 1,9 kHz ($1/f = 520 \mu s$) with an input microcomputer clock frequency of 4,4 MHz. Since ASC is 1, the mark/space ratio will be 3:1 so that the HIGH time of the clock pulses is 390 μs and the LOW time is 130 μs .

If ASC is 0, the pulses on clock line SCL have a high/low ratio of 1:1 and any of the clock frequency control numbers, except 0, may be loaded into bit positions S20 to S24. The clock-pulse frequency may therefore be selected within the range 700 Hz to 114 kHz. This is the high-speed mode of the SIO interface.

Table 3 Clock pulse frequency control when using a 4,43 MHz crystal

HEX S20-S24 code	divisor	approx. f_{clock} (kHz)
0	NOT ALLOWED	
1	39	114
2	45	98
3	51	87
4	63	70
5	75	59
6	87	51
7	99	45
8	123	36
9	147	30
A	171	26
B	195	23
C	243	18
D	291	15
E	339	13
F	387	11
10	483	9,2
11	579	7,7
12	675	6,6
13	771	5,8
14	963	4,6
15	1155	3,8
16	1347	3,3
17	1539	2,9
18*	1923	2,3
19*	2307	1,9
1A*	2691	1,7
1B*	3075	1,4
1C	3843	1,2
1D	4611	1,0
1E	5379	0,8
1F	6147	0,7

* only values that may be used in the low speed mode (ASC = 1).

ACK: 'acknowledge' bit at S26. For operation in the acknowledgement mode, the ACK bit position in register S2 must be loaded with a 1. The acknowledgement procedure of the SIO interface is illustrated in Fig.9. Its operation for devices in each of the four SIO interface modes will now be described.

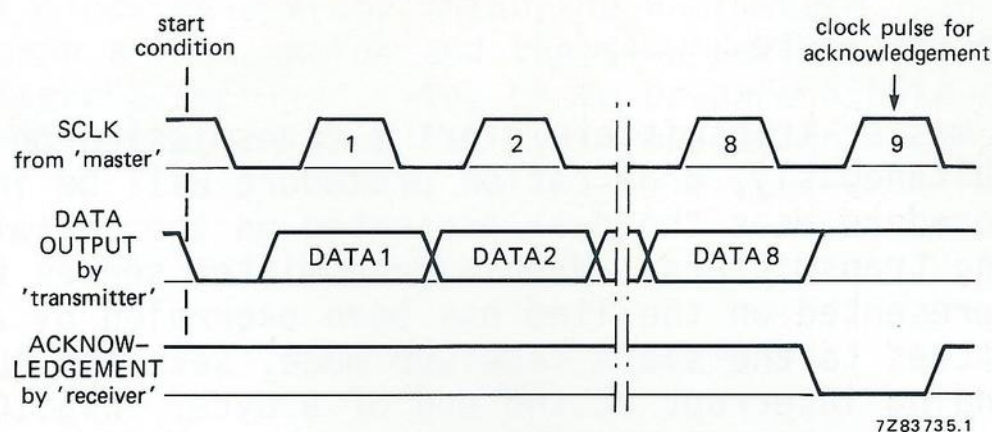


Fig.9 Acknowledgement between 'receiver' and 'transmitter'.

If the ACK bit in register S2 of a 'master transmitter' is 1, the device generates an extra clock pulse on SCL at the end of each transmitted byte. During this clock pulse, the SERIAL DATA I/O (pin 2) becomes an input and, if a 'receiver' generates an acknowledgement (LOW level on data line SDA), it resets the LRB flag to 0. If an acknowledgement is not received, the level on data line SDA remains HIGH and this sets the LRB flag to 1.

A 'slave transmitter' reacts to an acknowledgement in the same manner as a 'master transmitter' except it receives the extra clock pulse from the 'master' device instead of generating it.

If the ACK bit in clock control register S2 of a 'master receiver' is 1, the device generates an extra clock pulse after each received byte and applies it to clock line SCL. During this clock pulse, the SERIAL DATA I/O (pin 2) becomes an output where a LOW level is generated on data line SDA to acknowledge receipt of the transmission.

A 'slave receiver' generates an acknowledgement in the same manner as a 'master receiver' except that it receives the extra clock pulse from the 'master' device instead of generating it.

It should be noted that, if a device changes from a 'receiver' to a 'transmitter' after the R/W bit, an acknowledgement will be generated for the first byte by that device. If a device is changed from a 'transmitter' to a 'receiver' after the R/W bit, the acknowledgement for the first byte will be received by that device.

If the ACK bit position in clock control register S2 is loaded with a 0, the SIO interface is no longer in the acknowledgement mode. A 'master' device does not then generate an extra clock pulse after each byte and a 'receiver' does not generate an acknowledgement LOW level on bus line SDA.

8.0 SERIAL I/O OPERATIONS

8.1 Arbitration procedure

If two or more master transmitters start a transmission on the same bus almost simultaneously, arbitration procedure will be invoked. The arbitration procedure uses the data presented on the serial data line by the competing transmitters. When a transmitter senses that a HIGH signal it has presented on the line has been overruled by a LOW signal, it switches to the slave receiver mode, sets the AL flag and generates a pending interrupt at the end of a byte. Fig.10 shows the arbitration procedure between two devices. The arbitration procedure will give priority to the device that transmits the serial data stream with the lowest binary value. Should two or more devices send identical first bytes, arbitration will continue on the subsequent bytes.

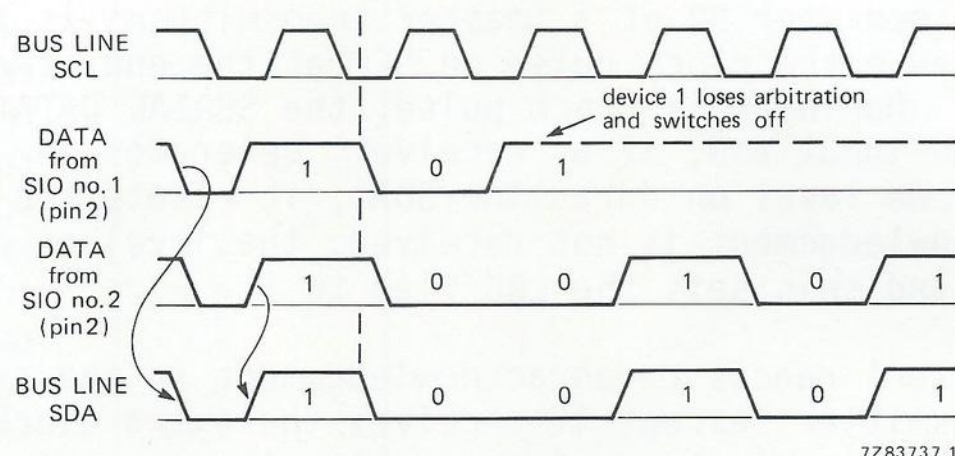


Fig.10 Arbitration procedure between two master transmitters.

8.2 Clock synchronisation

Under normal conditions, only one master device will generate a clock signal. During the arbitration procedure, however, there are two or more master devices and the clocks must be synchronised so that the data outputs can be compared. The wired-AND property of the clock line means that a device that first generates a LOW period of the clock line will overrule the other devices. At this HIGH/LOW transition, the clock generators of the other devices will be forced to start generation of their own LOW period. The clock line will then be held LOW by the device with the longest LOW period, while the other devices that finish their LOW periods must wait for the clock line to be released before starting their HIGH periods. A synchronised signal on the clock line is thus obtained, where the slowest device determines the length of the LOW period and the fastest the length of the HIGH period.

If a device pulls down the clock line for a longer time, the result will be that all clock generators enter the WAIT state. In this way a slave can slow down a fast master and the slow device can create enough time to store a received byte, or to prepare a byte to be transmitted. An illustration of clock synchronisation is given in Fig.11.

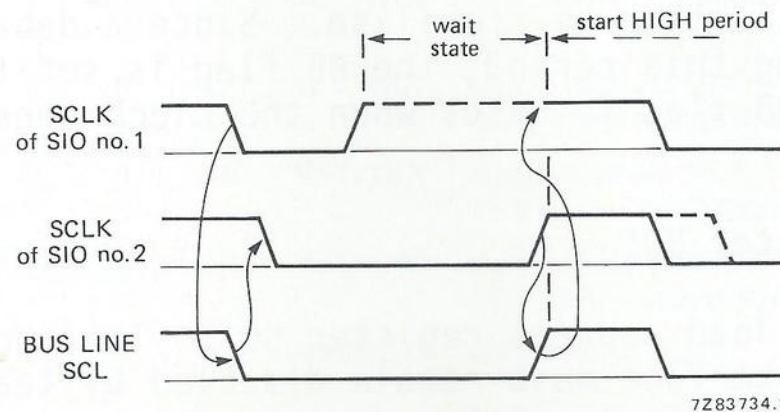


Fig.11 Synchronisation of two SIO clock generators.

9.0 PROGRAMMING

9.1 State after RESET

A reset (HIGH to pin 17 of the microcomputer) clears all the SIO interface registers (S0, S0', S1 and S2) except for the PIN (pending interrupt not) bit in status register S1 which is set to 1. Because the ES0 (enable serial output) bit in status register S1 is 0, the SIO interface is disabled, pin 2 has to its I/O function P23 of port 2 (HIGH after RESET) and pin 3 (SCLK) is in the high impedance state. An initialisation procedure must therefore be carried out before the SIO interface can be used to transfer serial data.

9.2 Initialisation procedure

A flow chart of the initialisation procedure is given in Fig.13. Each step of the procedure will now be explained in more detail.

9.2.1 Clock control register S2

The first step of the initialisation procedure is to load register S2. Bit position S20 to S24 are loaded, as indicated in Table 3, to set the frequency of the pulses on clock line SCL. Bit position ASC and ACK are loaded as previously described to determine whether the SIO interface operates with a symmetrical or asymmetrical clock, and whether or not it operates in the acknowledgement mode. After S2 has been loaded, the clock pulse generator requires one LOW period of its programmed clock cycle-time to stabilise. Since a data transfer must not be started during this period, the BB flag is set to indicate such an occurrence. The BB flag is reset when the clock generator has stabilised.

9.2.2 Address register S0'

The next step is to load address register S0'. To address this register, the SIO interface must remain disabled by leaving the ESO (enable serial output) bit in status register S1 at 0. Address register S0' is loaded with the 7-bit slave address plus the ALS bit. If the ALS bit is 1, the SIO will service free data format so the slave address in S0' is immaterial.

9.2.3 Status register S1

The final step of the initialisation procedure is to change the ESO bit in status register S1 to 1. All the other bits in register S1 remain as they were following the RESET, i.e. all bits 0 except PIN which is 1. The status register therefore contains HEX '18' and the SIO interface is in the 'slave receiver' mode and ready to receive serial data.

9.2.4 Enable/disable serial I/O interrupt

The instruction EN SI must be performed to enable the serial interrupt logic. If the serial interrupt logic is not enabled, the SIO interface can be serviced by polling PIN.

9.2.5 Example of an initialisation procedure

After a RESET and a jump to label INSI, initialisation of the SIO interface can be achieved as follows.

```
INSI MOV S2,# H'43'  WITH ACK, FSCLK ≈ 87 kHz at 4.43 MHz
      MOV S0,# H'84'  LOAD SLAVE ADDRESS HEX '42'(bit7-1) IN S0' ALS=0(bit 0)
      MOV S1,# H'18'  ENABLE SIO, SELECT SLAVE RECEIVER MODE
      EN SI           ENABLE SERIAL I/O INTERRUPT
```

The SIO interface is now in the 'slave receiver' mode. If address HEX'42' is received, the device will generate an acknowledge bit and cause a serial I/O interrupt call. Furthermore software responses depend on the contents of status register S1 as shown in Table 4.

TABLE 4 Possible bit combinations of the status word.

serial I/O status word S1								mode	concerned received or transmitted serial byte	software response
MST	TRX	BB	PIN	AL	AAS	ADO	LRB			
0	0	<u>1</u>	0	0	1	1/0	X	SLV/REC	own slave/all 0's address received with $R/\bar{W} = 0$	write 00111XXX to S1 or a dummy read of S0
0	0	<u>1</u>	0	0	0	1/0	X	SLV/REC	data received in S0	00101XXX to S1 and/or read S0
0	1	<u>1</u>	0	0	1	<u>0</u>	X	SLV/TRX	own slave/all 0's address received with $R/\bar{W} = 1$	write 01101XXX to S1 and/or load S0
0	1	<u>1</u>	0	0	0	<u>0</u>	1/0	SLV/TRX	transmitted out S0. LRB: 1/0 → NO ACK/ACK retruned	write 01101XXX to S1 and/or load S0, or write 00111000 to S1 (end)
1	1	<u>1</u>	0	<u>0</u>	<u>0</u>	<u>0</u>	1/0	MST/TRX	transmitted out S0. LRB = 1/0 → NO ACK/ACK returned	write 11101XXX to S1 and/or load S0, or write 11011000 to S1(STO.C)
1	0	<u>1</u>	0	<u>0</u>	<u>0</u>	<u>0</u>	1/0	MST/REC	slave address with $R/\bar{W} = 1$ transmitted LRB = 1/0 → NO ACK/ACK returned	write 10111XXX to S1 or a dummy read of S0
1	0	<u>1</u>	0	<u>0</u>	<u>0</u>	<u>0</u>	X	MST/REC	data received in S0	write 10101XXX to S1 and/or read S0, or write 11011000 to S1(STO.C)
0	0	<u>1</u>	0	1	0	<u>0</u>	X	MST/TRX has lost arbitration	slave address or data transmitted and arbitration lost	a dummy read of S0
0	0	<u>1</u>	0	1	<u>1</u>	<u>0</u>	X	MST/TRX has lost arbitration now SLV/REC	during transmitting of slave address arbitration lost and own slave/all 0's address received with $R/\bar{W} = 0$	write 00111XXX to S1 or a dummy read of S0
0	1	<u>1</u>	0	1	1	<u>0</u>	X	MST/TRX has lost arbitration now SLV/TRX	during transmitting of slave address arbitration lost and own slave/all 0's address received with $R/\bar{W} = 1$	write 01101XXX to S1 and/or load S0
<u>0</u>	<u>0</u>	0	<u>1</u>	<u>0</u>	<u>0</u>	<u>0</u>	X		the bus is free, a transmission may be started	
X	X	1	X	X	X	X	X		the bus is busy, a transmission may not be started	
X	X	X	1	X	X	X	X		no serial I/O interrupt pending i.e. no complete byte received or transmitted (polling PIN)	

X = 1 or 0, not relevant for status.
Underlined bits not relevant for status.

XXX = BC2, BC1, BC0 according to table 2.
(STO.C) = generation of the STOP CONDITION.

9.3 Generation of a 'start condition' and the first byte.

After completion of the initialisation procedure, serial data can be transmitted by selecting the 'master transmitter' modes as shown by the flow chart in Fig.14. If the device is connected in multi-transmitter bus system, the state of the BB flag must be tested to verify whether the serial bus is free. If the bus is free (BB = 0), data register S0 is loaded with the first byte for transmission. Bit position MST, TRX, and BB in status register S1 must each be loaded with a 1 to immediately start the data transmission beginning with a 'start' condition. When the first byte has been transmitted, the PIN flag is reset to 0. A serial interrupt call is initiated if the instruction EN SI has been performed previously. The status word in register S1 determines the software responses that will ensue as listed in Table 4. An example of a program which generates the 'start' condition and transmits the contents of working register Rr is as follows:

```

MSTX MOV A,S1          LOAD STATUS WORD TO ACCU
      JB5 MSTX          TEST IF BUS IS FREE
      MOV A,Rr          LOAD BYTE TO BE TRANSMITTED FROM Rr TO ACCU
      MOV S0,A          LOAD BYTE TO BE TRANSMITTED FROM ACCU TO S0
      MOV S1,# H'F8'    SELECT MASTER TRANSMITTER MODE AND
*                               START TRANSMISSION

```

After checking that the serial bus is free, but before the transmission starts, another device may engage the bus. If this occurs, the SIO interface will not start its transmission. To prevent received data from being corrupted, the SIO hardware inhibits data from being written by software into data register S0 and status register S1. If an instruction then attempts to write into register S1, the AL (arbitration lost) flag is set to indicate that the attempt to transmit has failed. When the AL flag is set, a serial interrupt request is always generated at the end of the serial byte.

9.4 Software responses after transmission or reception of a byte

After transmission or reception of a byte, the PIN flag is reset to 0. If a serial I/O interrupt call to program memory location 5 (serial interrupt enabled) is made or if the PIN flag is polled (serial interrupt disabled), a software response has to be initiated to service the SIO interface. As long as the PIN flag is 0, transfer of serial data is halted because clock bus line SCL is held LOW. Reading out or writing information to data shift register S0 sets the PIN flag to 1 and allows the transfer of serial data to continue.

The status word in register S1 contains all the information to determine the required software response. It also indicates the operating mode of the SIO interface, and gives information about the transmitted or received serial byte. Table 4 lists all possible bit combinations for the status word. It must be remembered that the four least-significant bits of the status word that can be read are not the same as those that can be written (see Fig.7). The least-significant bit (LRB) of the status word during a read operation only indicates an acknowledgement if the SIO interface is programmed to operate in the acknowledgement mode, i.e. the ACK bit in clock control register S2 is set to 1. If the ACK bit is 0, the LRB bit position in the status register contains the last bit of the byte that has been transmitted or received.

The following is an example of software response by a 'master transmitter'. The next 8-bit byte to be transmitted is in working register Rq.

	ORG H'05'	SERIAL INTERRUPT VECTOR
	JMP SIR	JUMP TO SERIAL INTERFACE ROUTINE
*		
SIR	SEL RB1	SELECT REGISTER BANK 1
	MOV Rp,A	SAVE ACCU CONTENTS IN Rp
	MOV A,S1	LOAD SERIAL STATUS IN ACCU
	JB7 TXTS	TEST MST BIT
	JMP SLVR	JUMP TO SLAVE ROUTINE IF MST=0
TXTS	JB6 TNBY	TEST TRX BIT
	JMP MRRC	JUMP TO MST/REC ROUTINE IF TRX=0
TNBY	MOV A,Rq	LOAD NEXT BYTE TO BE TRANSMITTED
*		FROM Rq TO ACCU
	MOV SO,A	LOAD IT TO SO=START TRANSMISSION
	MOV A,Rp	RESTORE CONTENTS OF ACCU
	RETR	RETURN TO MAIN PROGRAM

Note that bit counter bits BC0, BC1 and BC2 in the status register need not be preset as they can all remain 0 because 8 data bits have to be transmitted (see Table 2). The state of the LRB flag can be tested to check whether the transmitted data has been acknowledged. The result must be interpreted by the programmer.

9.5 Generation of the 'stop' condition

A data transfer ends with a 'stop' condition generated by the 'master' device. To generate the 'stop' condition, the program in section 9.4 is followed until TXTS. After checking that the last byte has been transferred, the program continues with:

	MOV S1,# H'D8'	RESET BB TO 0, MST, TRX, PIN TO 1
*		THIS GENERATES THE STOP CONDITION
	MOV A,Rp	RESTORE CONTENTS OF ACCU
	RETR	RETURN TO MAIN PROGRAM

9.6 Generation of repeated 'start' condition

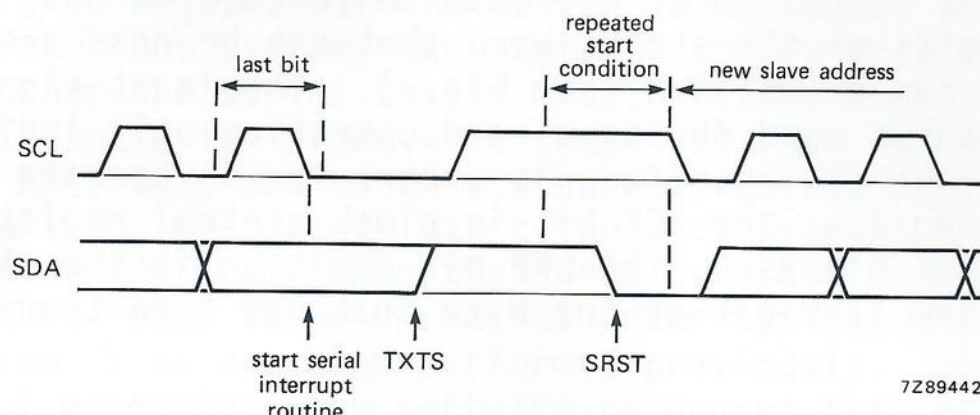


Fig.12 Repeated 'start' condition on the serial bus.

Fig.12 shows the generation of a repeated 'start' condition followed by a new slave address. The addressing format with repeated 'start' condition is shown in Fig.4(b). The serial bus remains busy at the end of the preceding data transfer because a 'stop' condition is not generated. The repeated 'start' condition is generated by a program which follows the program in section 9.4 until TXTS. After a check to verify that a repeated 'start' condition must be transmitted, the program continues with:

```

MOV S1,# H'18' RESET MST, SLV, BB TO 0
*          SET PIN TO 1
*          NOW BOTH SDA AND SCL
*          BECOME HIGH (NO STOP CONDITION), PROVIDED SCL IS
*          NOT PULLED DOWN BY ANOTHER DEVICE
MSTX  MOV A,S1      LOAD STATUS WORD TO ACCU **
      JB5 MSTX      TEST BB. AS LONG AS THE INTERNAL SERIAL
*                  CLOCK IS LOW, BB = 1 **
      CPL A         COMPLEMENT ACCU
      JB1 MSTX      TEST LRB=TEST IF SCL IS HIGH *** LRB IS SET TO 1
*                  AT THE POSITIVE GOING EDGE OF SCL (SDA=1)
      MOV A,Rq      LOAD NEW SLAVE ADD. TO ACCU
      MOV S0,A      LOAD NEW SLAVE ADD. TO S0
SRTS  MOV S1,# H'F8' SELECT MASTER TRANSMITTER MODE
*                  START TRANSMISSION
      MOV A,Rp      RESTORE CONTENTS OF ACCU
      RETR          RETURN TO MAIN PROGRAM

```

** This instruction may be omitted if it is certain that the LOW period of the serial clock finishes before label SRST is reached.

*** As long as a slow slave pulls down the SCL line, the master is not able to generate a repeated 'start' condition.

9.7 Generation of the 'stop' condition by a 'Master Receiver'

If a 'master receiver' wants to conclude a data transfer, it must instruct the 'slave transmitter' to free data bus line SDA so that it can generate the 'stop' condition. The 'master receiver' therefore generates a negative acknowledge (data line SDA held HIGH) after reception of the current data byte. The 'slave transmitter' recognises the negative acknowledge by testing the LRB flag and must then be changed to the 'slave receiver' mode by software. The generation of the 'stop' condition consists of the following three stages:

1. An 8-bit byte is received without generating an acknowledgement bit because bit position ACK in clock control register S2 is loaded with a 0.
2. The 'master receiver' generates a negative acknowledgement by generating an extra clock pulse, i.e. it generates a single HIGH-state bit on data line SDA.
3. The 'stop' condition is generated as a 'master transmitter'

The 'master receiver' program for the last received byte is the same as that given in section 9.4 until TNBY. After a check has verified that the last byte is to be transferred, the program continues with:

```

      MOV S2,# H'03' NO ACK, FSCL = 87 kHz
      MOV A,S0      READ S0 AND START
*
      MOV @R0,A     STORE RECEIVED DATA
      MOV A,Rp      RESTORE CONTENTS OF ACCU
      RETR          RETURN TO MAIN PROGRAM

```

During the next serial interrupt routine, the program in section 9.4 is again followed until TNBY. After a check has verified that the last byte has been transferred, a negative acknowledge and a 'stop' condition must be transmitted. The program continues with:

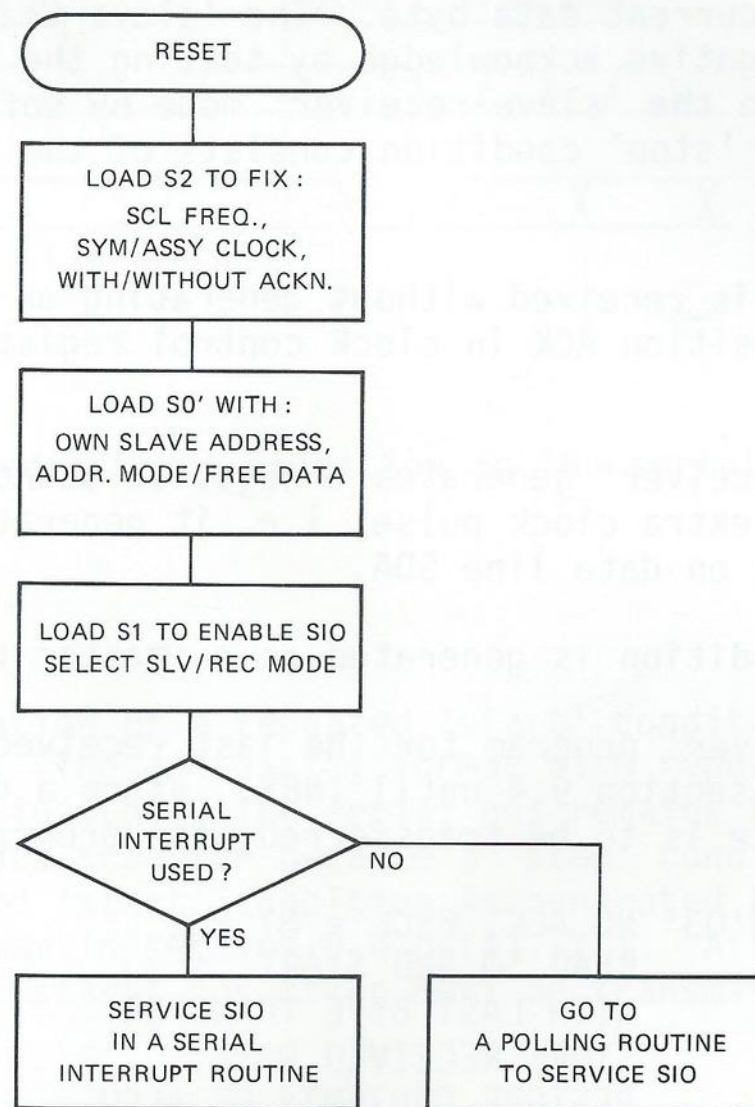
```

      MOV S1,# H'A9' LOAD BIT COUNTER FOR
*
      MOV A,S0      ONE BIT TRANSFER
*
      MOV A,S0      READ S0 AND START
*
      MOV A,S0      ONE BIT TRANSFER =
      MOV A,S0      NEGATIVE ACKNOWLEDGE
      MOV @R0,A     STORE RECEIVED DATA
TSP1  MOV A,S1      LOAD STATUS WORD TO ACCU **
      JB4 TSP1      TEST END OF ONE BIT TRANSFER **
      MOV S2,# H'43' RESTORE ACK BIT IN S2
      MOV S1,# H'D8' GENERATE STOP CONDITION
      MOV A,Rp      RESTORE ACCU
      RETR          RETURN TO MAIN PROGRAM

```

** These instructions can be omitted if it is certain that the negative acknowledge has been transferred before instruction MOV S2,# H'43' is given.

9.8 Flow charts for the 'master' and 'slave' functions of the SIO interface.



7Z89441

Fig.13 Initialisation flow chart. A prior RESET signal has cleared all bits of the SIO registers except PIN = 1 in status registers S1.



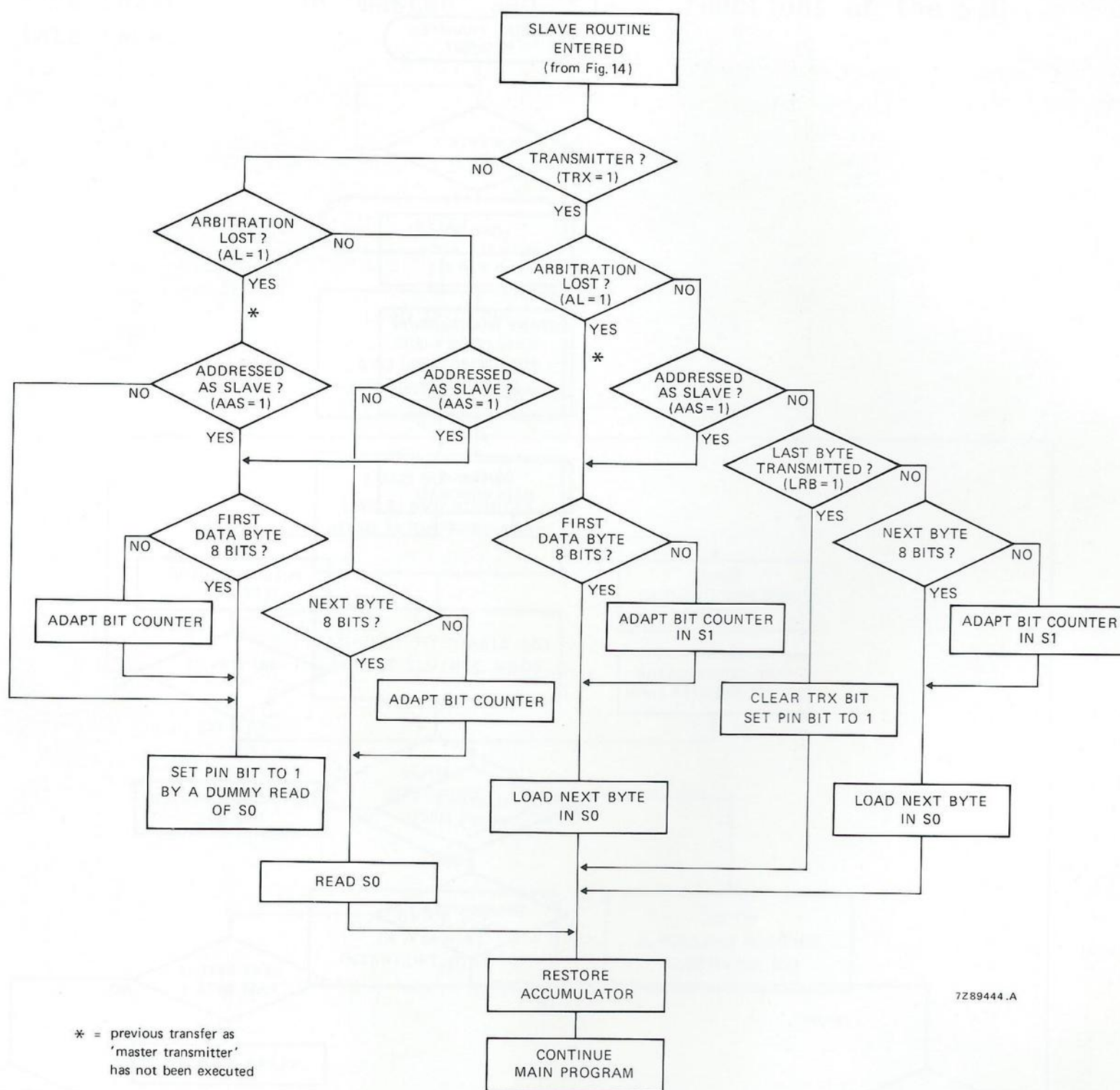


Fig.15 Flow chart of the 'slave' mode after initialisation.

9.9 Equate list for serial I/O control bits

To improve the readability of programs containing serial I/O routines the following list of equates is proposed:

REGISTER S0'

ALS	EQU	H'01'	ALWAYS SELECTED BIT
SLAB	EQU	H'FE'	SLAVE ADDRESS BITS
SLA AD	EQU	H'01' to H'7F'	OWN SLAVE ADDRESS

REGISTER S0

RW	EQU	H'01'	READ/WRITE NOT CONTROL BIT
----	-----	-------	----------------------------

REGISTER S1

READ AND WRITE BITS

MST	EQU	H'80'	MASTER BIT
TRX	EQU	H'40'	TRANSMITTER BIT
BB	EQU	H'20'	BUS BUSY BIT
PIN	EQU	H'10'	PENDING INTERRUPT NOT BIT

WRITE BITS

ESO	EQU	H'08'	ENABLE SERIAL I/O BIT
BCB	EQU	H'07'	SIO BITCOUNTER BITS
BC1	EQU	1	SIO BITCOUNTER = 1
BC2	EQU	2	SIO BITCOUNTER = 2
BC3	EQU	3	SIO BITCOUNTER = 3
BC4	EQU	4	SIO BITCOUNTER = 4
BC5	EQU	5	SIO BITCOUNTER = 5
BC6	EQU	6	SIO BITCOUNTER = 6
BC7	EQU	7	SIO BITCOUNTER = 7
BC8	EQU	8	SIO BITCOUNTER = 8

READ BITS

AL	EQU	H'08'	ARBITRATION LOST BIT
AAS	EQU	H'04'	ADDRESSED AS SLAVE BIT
ADO	EQU	H'02'	ADDRESS ZERO BIT
LRB	EQU	H'01'	LAST RECEIVED BIT

REGISTER S2

ASC	EQU	H'20'	ASYNCHRONOUS CLOCK BIT
ACK	EQU	H'40'	WITH ACKNOWLEDGE BIT
SCLFB	EQU	H'1F'	SCL CLOCK FREQUENCY BITS

2-3 Cascade list for serial I/O control bits

To improve the readability of program containing serial I/O routines
the following list of equates is proposed:

REGISTER 20

20.0	W.00	W.00
20.1	W.01	W.01
20.2	W.02	W.02
20.3	W.03	W.03
20.4	W.04	W.04
20.5	W.05	W.05
20.6	W.06	W.06
20.7	W.07	W.07

REGISTER 21

21.0	W.00	W.00
21.1	W.01	W.01
21.2	W.02	W.02
21.3	W.03	W.03
21.4	W.04	W.04
21.5	W.05	W.05
21.6	W.06	W.06
21.7	W.07	W.07

REGISTER 22

READ AND WRITE BITS

22.0	W.00	W.00
22.1	W.01	W.01
22.2	W.02	W.02
22.3	W.03	W.03
22.4	W.04	W.04
22.5	W.05	W.05
22.6	W.06	W.06
22.7	W.07	W.07

WRITE BITS

23.0	W.00	W.00
23.1	W.01	W.01
23.2	W.02	W.02
23.3	W.03	W.03
23.4	W.04	W.04
23.5	W.05	W.05
23.6	W.06	W.06
23.7	W.07	W.07
23.8	W.08	W.08
23.9	W.09	W.09
23.10	W.10	W.10
23.11	W.11	W.11
23.12	W.12	W.12
23.13	W.13	W.13
23.14	W.14	W.14
23.15	W.15	W.15
23.16	W.16	W.16
23.17	W.17	W.17
23.18	W.18	W.18
23.19	W.19	W.19
23.20	W.20	W.20
23.21	W.21	W.21
23.22	W.22	W.22
23.23	W.23	W.23
23.24	W.24	W.24
23.25	W.25	W.25
23.26	W.26	W.26
23.27	W.27	W.27
23.28	W.28	W.28
23.29	W.29	W.29
23.30	W.30	W.30
23.31	W.31	W.31
23.32	W.32	W.32
23.33	W.33	W.33
23.34	W.34	W.34
23.35	W.35	W.35
23.36	W.36	W.36
23.37	W.37	W.37
23.38	W.38	W.38
23.39	W.39	W.39
23.40	W.40	W.40
23.41	W.41	W.41
23.42	W.42	W.42
23.43	W.43	W.43
23.44	W.44	W.44
23.45	W.45	W.45
23.46	W.46	W.46
23.47	W.47	W.47
23.48	W.48	W.48
23.49	W.49	W.49
23.50	W.50	W.50
23.51	W.51	W.51
23.52	W.52	W.52
23.53	W.53	W.53
23.54	W.54	W.54
23.55	W.55	W.55
23.56	W.56	W.56
23.57	W.57	W.57
23.58	W.58	W.58
23.59	W.59	W.59
23.60	W.60	W.60
23.61	W.61	W.61
23.62	W.62	W.62
23.63	W.63	W.63
23.64	W.64	W.64
23.65	W.65	W.65
23.66	W.66	W.66
23.67	W.67	W.67
23.68	W.68	W.68
23.69	W.69	W.69
23.70	W.70	W.70
23.71	W.71	W.71
23.72	W.72	W.72
23.73	W.73	W.73
23.74	W.74	W.74
23.75	W.75	W.75
23.76	W.76	W.76
23.77	W.77	W.77
23.78	W.78	W.78
23.79	W.79	W.79
23.80	W.80	W.80
23.81	W.81	W.81
23.82	W.82	W.82
23.83	W.83	W.83
23.84	W.84	W.84
23.85	W.85	W.85
23.86	W.86	W.86
23.87	W.87	W.87
23.88	W.88	W.88
23.89	W.89	W.89
23.90	W.90	W.90
23.91	W.91	W.91
23.92	W.92	W.92
23.93	W.93	W.93
23.94	W.94	W.94
23.95	W.95	W.95
23.96	W.96	W.96
23.97	W.97	W.97
23.98	W.98	W.98
23.99	W.99	W.99
23.100	W.100	W.100
23.101	W.101	W.101
23.102	W.102	W.102
23.103	W.103	W.103
23.104	W.104	W.104
23.105	W.105	W.105
23.106	W.106	W.106
23.107	W.107	W.107
23.108	W.108	W.108
23.109	W.109	W.109
23.110	W.110	W.110
23.111	W.111	W.111
23.112	W.112	W.112
23.113	W.113	W.113
23.114	W.114	W.114
23.115	W.115	W.115
23.116	W.116	W.116
23.117	W.117	W.117
23.118	W.118	W.118
23.119	W.119	W.119
23.120	W.120	W.120
23.121	W.121	W.121
23.122	W.122	W.122
23.123	W.123	W.123
23.124	W.124	W.124
23.125	W.125	W.125
23.126	W.126	W.126
23.127	W.127	W.127
23.128	W.128	W.128
23.129	W.129	W.129
23.130	W.130	W.130
23.131	W.131	W.131
23.132	W.132	W.132
23.133	W.133	W.133
23.134	W.134	W.134
23.135	W.135	W.135
23.136	W.136	W.136
23.137	W.137	W.137
23.138	W.138	W.138
23.139	W.139	W.139
23.140	W.140	W.140
23.141	W.141	W.141
23.142	W.142	W.142
23.143	W.143	W.143
23.144	W.144	W.144
23.145	W.145	W.145
23.146	W.146	W.146
23.147	W.147	W.147
23.148	W.148	W.148
23.149	W.149	W.149
23.150	W.150	W.150
23.151	W.151	W.151
23.152	W.152	W.152
23.153	W.153	W.153
23.154	W.154	W.154
23.155	W.155	W.155
23.156	W.156	W.156
23.157	W.157	W.157
23.158	W.158	W.158
23.159	W.159	W.159
23.160	W.160	W.160
23.161	W.161	W.161
23.162	W.162	W.162
23.163	W.163	W.163
23.164	W.164	W.164
23.165	W.165	W.165
23.166	W.166	W.166
23.167	W.167	W.167
23.168	W.168	W.168
23.169	W.169	W.169
23.170	W.170	W.170
23.171	W.171	W.171
23.172	W.172	W.172
23.173	W.173	W.173
23.174	W.174	W.174
23.175	W.175	W.175
23.176	W.176	W.176
23.177	W.177	W.177
23.178	W.178	W.178
23.179	W.179	W.179
23.180	W.180	W.180
23.181	W.181	W.181
23.182	W.182	W.182
23.183	W.183	W.183
23.184	W.184	W.184
23.185	W.185	W.185
23.186	W.186	W.186
23.187	W.187	W.187
23.188	W.188	W.188
23.189	W.189	W.189
23.190	W.190	W.190
23.191	W.191	W.191
23.192	W.192	W.192
23.193	W.193	W.193
23.194	W.194	W.194
23.195	W.195	W.195
23.196	W.196	W.196
23.197	W.197	W.197
23.198	W.198	W.198
23.199	W.199	W.199
23.200	W.200	W.200
23.201	W.201	W.201
23.202	W.202	W.202
23.203	W.203	W.203
23.204	W.204	W.204
23.205	W.205	W.205
23.206	W.206	W.206
23.207	W.207	W.207
23.208	W.208	W.208
23.209	W.209	W.209
23.210	W.210	W.210
23.211	W.211	W.211
23.212	W.212	W.212
23.213	W.213	W.213
23.214	W.214	W.214
23.215	W.215	W.215
23.216	W.216	W.216
23.217	W.217	W.217
23.218	W.218	W.218
23.219	W.219	W.219
23.220	W.220	W.220
23.221	W.221	W.221
23.222	W.222	W.222
23.223	W.223	W.223
23.224	W.224	W.224
23.225	W.225	W.225
23.226	W.226	W.226
23.227	W.227	W.227
23.228	W.228	W.228
23.229	W.229	W.229
23.230	W.230	W.230
23.231	W.231	W.231
23.232	W.232	W.232
23.233	W.233	W.233
23.234	W.234	W.234
23.235	W.235	W.235
23.236	W.236	W.236
23.237	W.237	W.237
23.238	W.238	W.238
23.239	W.239	W.239
23.240	W.240	W.240
23.241	W.241	W.241
23.242	W.242	W.242
23.243	W.243	W.243
23.244	W.244	W.244
23.245	W.245	W.245
23.246	W.246	W.246
23.247	W.247	W.247
23.248	W.248	W.248
23.249	W.249	W.249
23.250	W.250	W.250
23.251	W.251	W.251
23.252	W.252	W.252
23.253	W.253	W.253
23.254	W.254	W.254
23.255	W.255	W.255
23.256	W.256	W.256
23.257	W.257	W.257
23.258	W.258	W.258
23.259	W.259	W.259
23.260	W.260	W.260
23.261	W.261	W.261
23.262	W.262	W.262
23.263	W.263	W.263
23.264	W.264	W.264
23.265	W.265	W.265
23.266	W.266	W.266
23.267	W.267	W.267
23.268	W.268	W.268
23.269	W.269	W.269
23.270	W.270	W.270
23.271	W.271	W.271
23.272	W.272	W.272
23.273	W.273	W.273
23.274	W.274	W.274
23.275	W.275	W.275
23.276	W.276	W.276
23.277	W.277	W.277
23.278	W.278	W.278
23.279	W.279	W.279
23.280	W.280	W.280
23.281	W.281	W.281
23.282	W.282	W.282
23.283	W.283	W.283
23.284	W.284	W.284
23.285	W.285	W.285
23.286	W.286	W.286
23.287	W.287	W.287
23.288	W.288	W.288
23.289	W.289	W.289
23.290	W.290	W.290
23.291	W.291	W.291
23.292	W.292	W.292
23.293	W.293	W.293
23.294	W.294	W.294
23.295	W.295	W.295
23.296	W.296	W.296
23.297	W.297	W.297
23.298	W.298	W.298
23.299	W.299	W.299
23.300	W.300	W.300
23.301	W.301	W.301
23.302	W.302	W.302
23.303	W.303	W.303
23.304	W.304	W.304
23.305	W.305	W.305
23.306	W.306	W.306
23.307	W.307	W.307
23.308	W.308	W.308
23.309	W.309	W.309
23.310	W.310	W.310
23.311	W.311	W.311
23.312	W.312	W.312
23.313	W.313	W.313
23.314	W.314	W.314
23.315	W.315	W.315
23.316	W.316	W.316
23.317	W.317	W.317
23.318	W.318	W.318
23.319	W.319	W.319
23.320	W.320	W.320
23.321	W.321	W.321
23.322	W.322	W.322
23.323	W.323	W.323
23.324	W.324	W.324
23.325	W.325	W.325
23.326	W.326	W.326
23.327	W.327	W.327
23.328	W.328	W.328
23.329</		

© 1982 N.V. Philips' Gloeilampenfabrieken

This information is furnished for guidance, and with no guarantees as to its accuracy or completeness; its publication conveys no licence under any patent or other right, nor does the publisher assume liability for any consequence of its use; specifications and availability of goods mentioned in it are subject to change without notice; it is not to be reproduced in any way, in whole or in part, without the written consent of the publisher.

Printed in The Netherlands

Date of release: October 1982

9398 614 50011