DATA SHEET

TELX family
Low voltage 8-bit microcontrollers

Preliminary specification
File under Integrated Circuits, IC20

1997 Jul 02

Philips
Semiconductors

PHILIPS

CONTENTS

1 FEATURES

- Full static 80C51 CPU (8-bit CPU) with a minimum 6 clocks per instruction
- OTP/ROM program memory
- RAM, expandable externally to 64 kbytes (only on certain devices)
- DTMF generator
- MSK modem including Manchester encoder/decoder for analog cordless telephones (standards CT0/CT1/CT1+)
- Pulse Width Modulated output (8-bit resolution)
- EEPROM data memory, accessed internally via I^2C-bus interface
- 8-bit ports, I/O lines
- Three 16-bit timer/event counters, including one with capture, compare and PWM function
- Watchdog Timer
- External memory expandable up to 128 kbytes external ROM up to 64 kbytes and/or RAM up to 64 kbytes (only possible on certain devices)
- On-chip amplitude controlled oscillator (ACO) suitable for quartz crystal or ceramic resonator
- 32 kHz Real-Time Clock (RTC) with programmable interrupt periods
- Twenty source, twenty vector interrupt structure with two priority levels
- Full duplex enhanced UART with double buffering
- I^2C-bus interface for 2-wire serial transfer, 400 kHz maximum
- Enhanced architecture with:
  - Non-page oriented instructions
  - Direct addressing
  - Four 8 byte RAM register banks
  - Stack depth limited only by available internal RAM (maximum 256 bytes)
  - Multiply, divide, subtract and compare instructions
- Eight additional interrupts on Port 1
  - Edge or level sensitive triggering selectable via software
  - Power-saving use for keyboard control.
- Improved Power-on/Power-off reset circuitry (POR) with 9 hardware programmable levels
- Low Voltage Detection (LVD) with 11 software programmable levels
- Wake-up from Power-down mode via external interrupts at Port 1, via RTC or via LVD
- Very low current consumption.

2 GENERAL DESCRIPTION

The TELX microcontroller family is manufactured in an advanced CMOS technology and is based on MCM (Multi-Chip-Module) technology as the non-volatile memory parts OTP and EEPROM are integrated on a separate chip.

The TELX family are 8-bit microcontrollers especially suited for analog cordless telephones (CT0, CT1, CT1+ standards), mid/high-end corded telephones and pagers. For this purpose, features like DTMF, EEPROM, MSK modem, PWM, POR/LVD, ACO and RTC are integrated on-chip. The device is optimized for low power consumption. The TELX family has two software selectable features for power reduction: Idle and Power-down modes.

The instruction set of the TELX family is based on that of the 8051. The TELX family also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 46 two-byte, and 16 three-byte.

This data sheet details the shared properties of the TELX family. For a particular microcontroller, read this data sheet in conjunction with the individual data sheet of the specific device. For details on the I^2C-bus functions see *“Data Handbook IC12”*.

3 BLOCK DIAGRAM

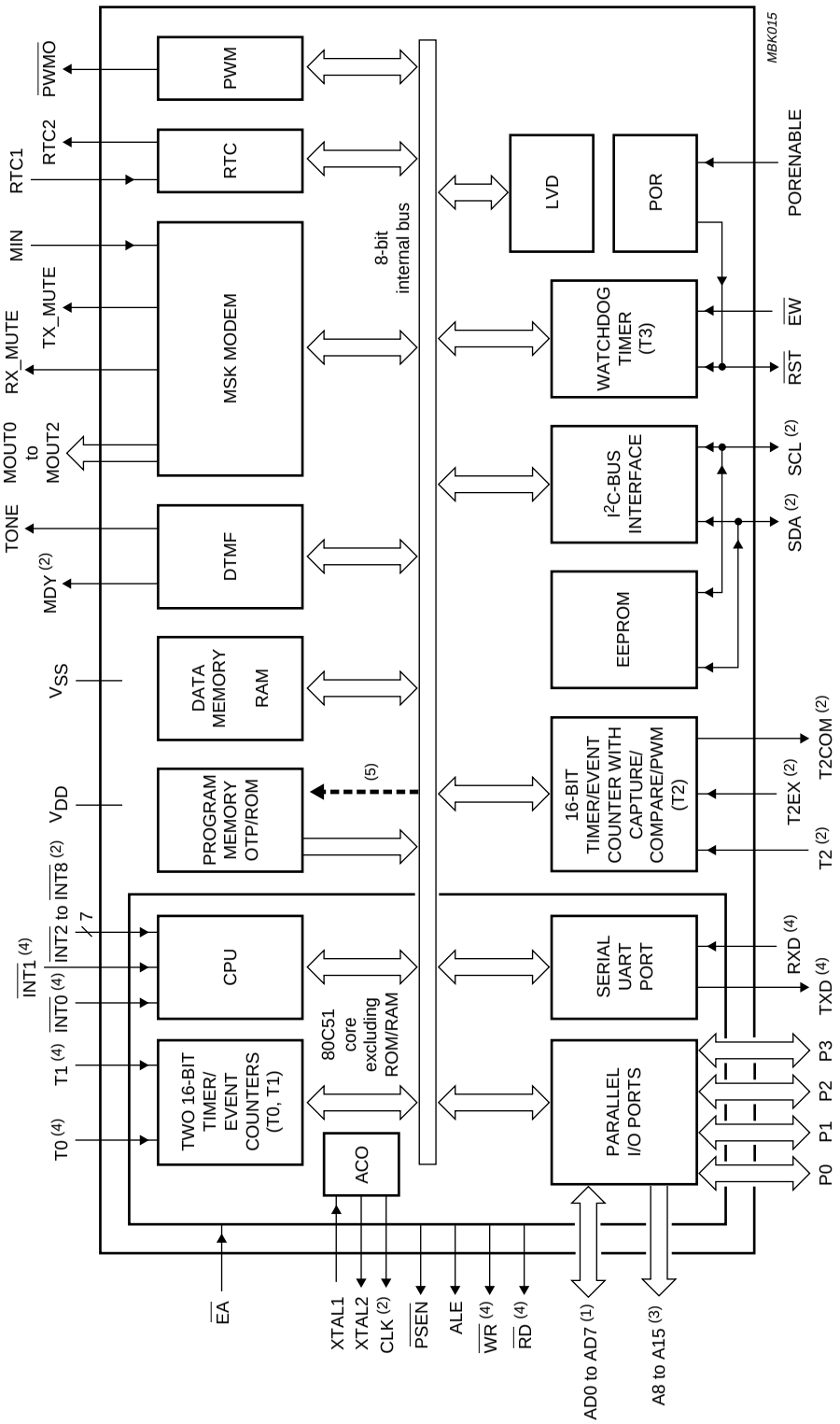


Fig.1 Block diagram.

(1) alternative function of Port 0
(2) alternative function of Port 1
(3) alternative function of Port 2
(4) alternative function of Port 3
(5) in-circuit OTP programming

4 FUNCTIONAL DESCRIPTION

4.1 General

The TELX family provides stand-alone high-performance CMOS microcontrollers designed for use in mid/high-end corded telephones, analog cordless telephones (CT0, CT1, CT1+ standards) and pagers. For this purpose, features such as DTMF, MSK modem, EEPROM, Real-Time-Clock and PWM have been integrated on-chip. The devices provide hardware features, architectural enhancements and new instructions to function as a controller for applications requiring up to 64 kbytes of program memory and/or up to 64 kbytes of data storage.

The TELX family contains ROM or OTP program memory; a static read/write data memory; I/O lines; three 16-bit timer/event counters; a twenty-source two priority-level, nested interrupt structure and on-chip oscillator and timing circuit.

The TELX devices have two software selectable modes of reduced activity for power reduction: Idle and Power-down. The Idle mode freezes the CPU while allowing the RAM, timers, serial I/O and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative.

Two serial interfaces are provided on-chip; a standard UART serial interface and an I^2C-bus serial interface. The I^2C-bus serial interface has byte orientated master and slave functions allowing communication with the whole family of I^2C-bus compatible devices.

4.2 CPU timing

A machine cycle consists of a sequence of 6 states. Each state lasts for one oscillator period, thus a machine cycle takes 6 oscillator periods or e.g. 1.68 µs if the oscillator frequency is 3.58 MHz. This means that the TELX family is twice as fast as a standard 80C51 based on the same oscillator frequency.

4.3 Clocking concept with prescaler PSC

The clocking concept of the TELX family is shown in Fig.2. The on-chip oscillator directly clocks the CPU (including timers T0 and T1), timer T2, PWM, the Watchdog Timer and the Analog-to-Digital Converter (ADC).

The DTMF block requires an input frequency of 3.58 MHz for correct operation. For this purpose a prescaler (PSC) has been included to enable multiples of 3.58 MHz to be used as the oscillator frequency. The blocks I^2C-bus, UART special purpose baud rate generator, MSK modem, Watchdog Timer and EEPROM are also clocked via the PSC prescaler to minimize the number of prescalers on-chip and thereby reducing the power consumption. The division factors 1:1 through to 1:8 of the PSC prescaler are software programmable via the PRESC register, see Tables 2 and 3. The PSC division factor should only be set in the initialization routine directly after start-up. The prescaler and the special baud rate timer for the UART are described in Section 4.13.

In order to minimize power consumption, the individual blocks automatically switch-off their clock (gated clock) when they are not enabled.

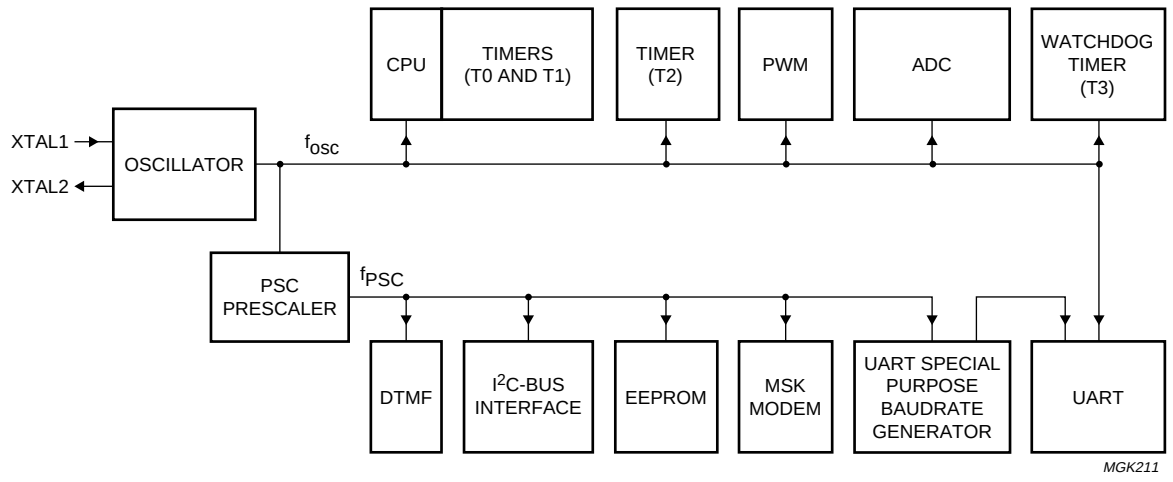


Fig.2 Clocking concept.

4.3.1 PRESCALER REGISTER (PRESC)

Table 1 Prescaler Register (SFR address F3H)

7	6	5	4	3	2	1	0
–	PTWO2	PTWO1	PTWO0	P3	PS2	PS1	PS0

Table 2 Description of PRESC bits

BIT	SYMBOL	DESCRIPTION
PRESC.7	–	This bit will always read a logic 0.
PRESC.6	PTWO2	These 3-bits specify the power of two in the division factor of the UART baud rate timer; see Section 4.13.3.1.
PRESC.5	PTWO1	
PRESC.4	PTWO0	
PRESC.3	P3	This bit specifies the power of three in the division factor of the UART baud rate timer; see Section 4.13.3.1.
PRESC.2	PS2	These 3-bits select the PSC division factor; see Table 3.
PRESC.1	PS1	
PRESC.0	PS0	

Table 3 Selection of PSC division factors

PS2	PS1	PS0	DIVISION RATIO (f_{PSC} : f_{osc})
0	0	0	1:1
0	0	1	1:2
0	1	0	1:3
0	1	1	1:4
1	0	0	1:5
1	0	1	1:6
1	1	0	1:7
1	1	1	1:8

4.4 32 kHz Real-Time Clock

The Real-Time Clock (RTC) consists of a 32 kHz crystal oscillator, a 32 kHz to 1 second or 1 minute divider chain, an 8-bit Frequency Adjustment Register (FAR) and the Clock Control Register (CLCR). The complete RTC section works independent of the microcontroller status, even in Idle and Power-down mode.

The RTC can generate an interrupt periodically every 1 minute or every 1, 2 or 4 seconds. This interrupt can be used to wake-up the microcontroller from the Power-down mode without resetting it. This feature is especially useful in CT0/CT1 cordless phone applications to wake-up the microcontroller periodically to perform channel scanning.

The internal 32 kHz oscillator requires an external quartz crystal with a frequency of 32.768 kHz (a positive deviation up to +200 ppm is allowed) and an external feedback resistor connected between pins RTC1 and RTC2; 4.7 MΩ is recommended.

The oscillator is controlled by the RUN bit in the Clock Control Register.

The divider chain operates with the 32 kHz oscillator output and divides this signal down to two clocks with a period of 1, 2 or 4 seconds or 1 minute respectively. Depending on the state of the ITS1 and ITS0 bits in the Clock Control Register, the falling edge of the seconds or minutes clock is used to set the Clock Interrupt Flag (CIF) in the Clock Control Register.

Additionally, the divider chain generates a 16 kHz clock (RTCLK) that can be routed through the port line P1.3/RTCLK, controlled by the ERCO bit in the Clock Control Register.

Frequency adjustment is used to extend the interrupt time by defining the number of 16 kHz clocks in the Frequency Adjustment Register (FAR) that will be counted twice within the first second period after a minute interrupt.

4.4.1 CLOCK CONTROL REGISTER (CLCR)

Table 4 Clock Control Register (address ACH; access type R/W)

7	6	5	4	3	2	1	0
TST2	TST1	ERC0	RUN	CIF	–	ITS1	ITS0

Table 5 Description of CLCR bits

BIT	SYMBOL	DESCRIPTION
7	TST2	**Test 2 input.** This is a testing bit and must be fixed at a logic 0 by user software.
6	TST1	**Test 1 input.** This is a testing bit and must be fixed at a logic 0 by user software.
5	ERCO	**Enable 16 kHz Clock Output.** If ERCO = 0, then P1.3/RTCLK is a port line. If ERCO = 1, then P1.3/RTCLK is a 16 kHz clock output. ERCO = 1 does not inhibit the port instructions for P1.3/RTCLK. Therefore, the state of both port line and flip-flop may be read in and the port flip-flop may be written by derivative port instructions. However, the port flip-flop of P1.3/RTCLK must remain set to avoid conflicts between the 16 kHz clock and port outputs.
4	RUN	**Clock Run/Stop.** If RUN = 0, then the oscillator is stopped and the clock is reset. If RUN = 1, then the oscillator and the clock are both running.
3	CIF	**Clock Interrupt Flag.** Set by hardware, if RTC divider chain overflows (period depending on the state the ITS1 and ITS0 bits) or by software. Reset by software.
2	–	Not used.
1	ITS1	**Interrupt Time Select bits.** The state of ITS1 and ITS0 determine the interrupt period, see Table 6.
0	ITS0	

Table 6 Selection of the RTC Interrupt period

ITS1	ITS0	INTERRUPT PERIOD
0	0	1 second; note 1
0	1	2 seconds
1	0	4 seconds
1	1	1 minute

Note

1. If the 1 second interrupt is used, every 60th interval may be up to 15.3 ms longer than the others as a result of the frequency adjustment. The adjusted Minute Interrupt Time (MIT) shows now a maximum deviation of 0.5 ppm.

4.4.2 Frequency Adjustment Register (FAR)

The frequency adjustment value of the RTC section is defined by the 8-bit Frequency Adjustment Register. The register access type is R/W. The significance of the individual bits of the FAR register can be illustrated by the following equation:

$$\text{Minute interrupt time (MIT)} = 60 \times 2^{\frac{14}{f_{RTCLK}}} + \frac{FAR}{2^{14}}$$

where f_{RTCLK} = RTC frequency and FAR represents the decimal value of the contents of the Frequency Adjustment Register.

Table 7 Frequency Adjustment Register (address ADH)

7	6	5	4	3	2	1	0
FAR7	FAR6	FAR5	FAR4	FAR3	FAR2	FAR1	FAR0

Table 8 Description of FAR bits

BIT	SYMBOL	DESCRIPTION
7	FAR7	The state of these 8-bits determine the frequency adjustment value for the Real-Time Clock; see Table 9.
6	FAR6	
5	FAR5	
4	FAR4	
3	FAR3	
2	FAR2	
1	FAR1	
0	FAR0	

Table 9 Selection of FAR value based on f_{RTCLK}

FAR (HEX)	f_{RTCLK}
00	16384.000
01	16384.018
02	16384.033
03	16384.051
04	16384.066
05	16384.084
06	16384.100
07	16384.117
08	16384.135
09	16384.150
0A	16384.168
0B	16384.184
0C	16384.201
0D	16384.217
0E	16384.234
0F	16384.250
10	16384.268
11	16384.283
12	16384.301
13	16384.316
14	16384.334
15	16384.350
16	16384.367
17	16384.385
18	16384.400
19	16384.418
1A	16384.434
1B	16384.451
1C	16384.467
1D	16384.484
1E	16384.500
1F	16384.518
20	16384.533

FAR (HEX)	f_{RTCLK}
21	16384.551
22	16384.566
23	16384.584
24	16384.600
25	16384.617
26	16384.635
27	16384.650
28	16384.668
29	16384.684
2A	16384.701
2B	16384.717
2C	16384.734
2D	16384.750
2E	16384.768
2F	16384.783
30	16384.801
31	16384.816
32	16384.834
33	16384.850
34	16384.867
35	16384.885
36	16384.900
37	16384.918
38	16384.934
39	16384.951
3A	16384.967
3B	16384.984
3C	16385.000
3D	16385.018
3E	16385.033
3F	16385.051
40	16385.066
41	16385.084

FAR (HEX)	f_{RTCLK}
42	16385.100
43	16385.117
44	16385.135
45	16385.150
46	16385.168
47	16385.184
48	16385.201
49	16385.217
4A	16385.234
4B	16385.250
4C	16385.268
4D	16385.283
4E	16385.301
4F	16385.316
50	16385.334
51	16385.350
52	16385.367
53	16385.385
54	16385.400
55	16385.418
56	16385.434
57	16385.451
58	16385.467
59	16385.484
5A	16385.500
5B	16385.518
5C	16385.533
5D	16385.551
5E	16385.566
5F	16385.584
60	16385.600
61	16385.617
62	16385.635
63	16385.650

FAR (HEX)	f_{RTCLK}
64	16385.668
65	16385.684
66	16385.701
67	16385.717
68	16385.734
69	16385.750
6A	16385.768
6B	16385.783
6C	16385.801
6D	16385.816
6E	16385.834
6F	16385.850
70	16385.867
71	16385.885
72	16385.900
73	16385.918
74	16385.934
75	16385.951
76	16385.967
77	16385.984
78	16386.000
79	16386.018
7A	16386.033
7B	16386.051
7C	16386.066
7D	16386.084
7E	16386.100
7F	16386.117
80	16386.135
81	16386.150
82	16386.168
83	16386.184
84	16386.201
85	16386.217

FAR (HEX)	f_{RTCLK}
86	16386.234
87	16386.250
88	16386.268
89	16386.283
8A	16386.301
8B	16386.316
8C	16386.334
8D	16386.350
8E	16386.367
8F	16386.385
90	16386.400
91	16386.418
92	16386.434
93	16386.451
94	16386.467
95	16386.484
96	16386.500
97	16386.518
98	16386.533
99	16386.551
9A	16386.566
9B	16386.584
9C	16386.600
9D	16386.617
9E	16386.635
9F	16386.650
A0	16386.668
A1	16386.684
A2	16386.701
A3	16386.717
A4	16386.734
A5	16386.750
A6	16386.768
A7	16386.783

FAR (HEX)	f_{RTCLK}
A8	16386.801
A9	16386.816
AA	16386.834
AB	16386.850
AC	16386.867
AD	16386.885
AE	16386.900
AF	16386.918
B0	16386.934
B1	16386.951
B2	16386.967
B3	16386.984
B4	16387.000
B5	16387.018
B6	16387.033
B7	16387.051
B8	16387.066
B9	16387.084
BA	16387.100
BB	16387.117
BC	16387.135
BD	16387.150
BE	16387.168
BF	16387.184
C0	16387.201
C1	16387.217
C2	16387.234
C3	16387.250
C4	16387.268
C5	16387.283
C6	16387.301
C7	16387.316
C8	16387.334
C9	16387.350

FAR (HEX)	f_{RTCLK}
CA	16387.367
CB	16387.385
CC	16387.400
CD	16387.418
CE	16387.434
CF	16387.451
D0	16387.467
D1	16387.484
D2	16387.500
D3	16387.518
D4	16387.533
D5	16387.551
D6	16387.566
D7	16387.584
D8	16387.600
D9	16387.617
DA	16387.635
DB	16387.650
DC	16387.668
DD	16387.684
DE	16387.701
DF	16387.717
E0	16387.734
E1	16387.750
E2	16387.768
E3	16387.783
E4	16387.801

FAR (HEX)	f_{RTCLK}
E5	16387.816
E6	16387.834
E7	16387.850
E8	16387.867
E9	16387.885
EA	16387.900
EB	16387.918
EC	16387.934
ED	16387.951
EE	16387.967
EF	16387.984
F0	16388.002
F1	16388.018
F2	16388.035
F3	16388.051
F4	16388.068
F5	16388.084
F6	16388.102
F7	16388.117
F8	16388.135
F9	16388.152
FA	16388.168
FB	16388.186
FC	16388.201
FD	16388.219
FE	16388.234
FF	16384.000

4.5 Memory organization

The TELX family has Program Memory (OTP or ROM) plus Data Memory (RAM) on-chip. The device has separate address spaces for Program and Data Memory as shown in Fig.3. On devices with ports P0 and P2 available, up to 64 kbytes of external memory can be addressed. In this case, the CPU generates both read ($\overline{RD}$) and write ($\overline{WR}$) signals for external Data Memory accesses, and the read strobe ($\overline{PSEN}$) for external Program Memory.

4.5.1 Program memory

After reset the CPU begins execution at location 0000H of the Program Memory. The Program Memory can be implemented in either internal OTP/ROM or external memory. If the $\overline{EA}$ pin is tied to V_{DD}, then program memory fetches are directed to the internal program memory. If the $\overline{EA}$ pin is tied to V_{SS} and if the security bits are not set, then program memory fetches are directed to external memory.

4.5.2 Data memory

The data memory organisation of the TELX family is exactly the same as for the P8xCE558. The TELX family contains a maximum of 512 bytes internal RAM (consisting of 256 bytes standard RAM and 256 bytes AUX-RAM) and Special Function Registers (SFRs). Figure 3 shows the internal Data Memory space divided into the lower 128 bytes, the upper 128 bytes, the SFR space and 256 bytes Auxiliary RAM. Internal RAM locations 0 to 127 are directly and indirectly addressable. Internal RAM locations 128 to 255 are only indirectly addressable. The SFRs locations 128 to 255 bytes are only directly addressable and the Auxiliary RAM is indirectly addressable as external RAM (MOVX) unless it is disabled by setting ARD = 1.

4.5.3 Special Function Registers

The second 128 bytes are the address locations of the SFRs. Figure 4 and Table 10 define the SFRs memory space. The SFRs include the port latches, timers, peripheral control, serial I/O registers, etc. These registers can only be accessed by direct addressing. There are 128 addressable locations in the SFR address space (those SFRs whose addresses are divisible by eight). Refer to the product specifications for the precise list of the SFRs implemented and their value directly after reset.

4.6 Addressing

The TELX family has five methods for addressing source operands:

- Register
- Direct
- Register-Indirect
- Immediate
- Base-Register plus Index-Register-Indirect.

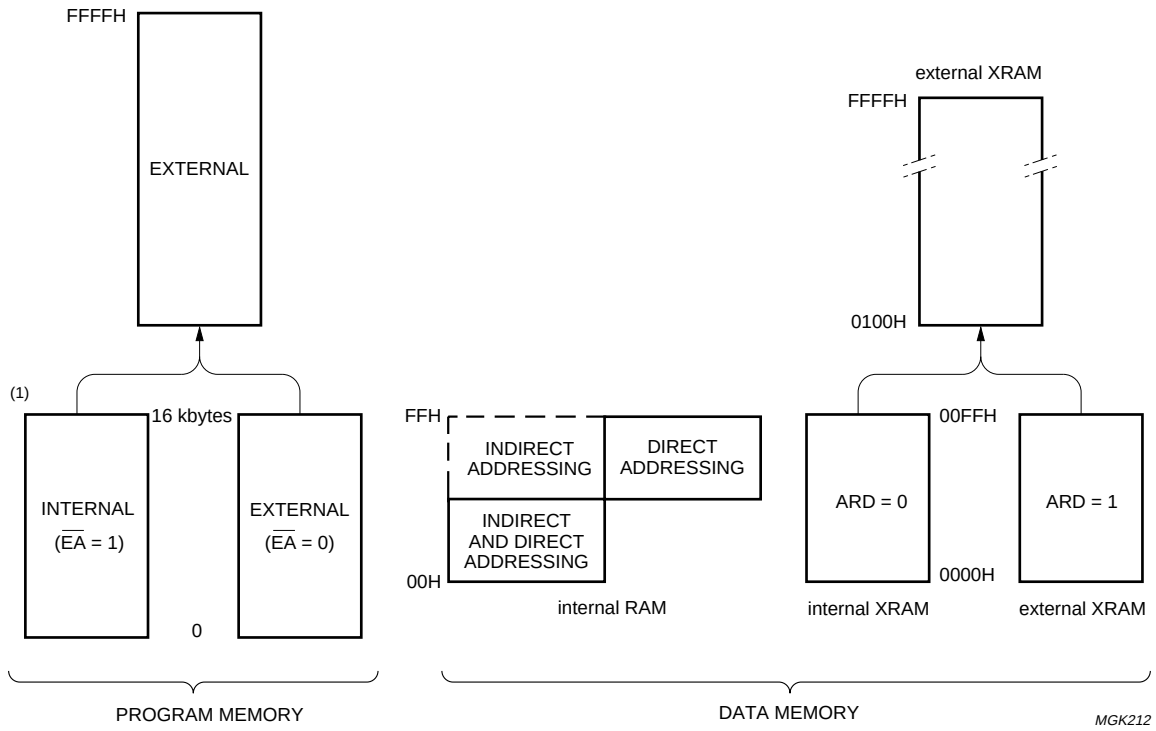
The first three methods can be used for addressing destination operands. Most instructions have a ‘destination/source’ field that specifies the data type, addressing methods and operands involved. For operations other than MOVs, the destination operand is also a source operand.

Access to memory addressing is as follows:

- Registers in one of the four 8-register banks through Register Direct or Register-Indirect
- A maximum of 1024 bytes of internal data RAM through Direct or Register-Indirect
  - Bytes 0 to 127 of internal RAM may be addressed directly or indirectly. Bytes 128 to 255 of internal RAM share their address location with the Special Function Registers and so may only be addressed indirectly as data RAM
  - Bytes 0 to 256 of AUX-RAM can only be addressed indirectly via MOVX instructions.
- Special Function Registers through Direct
- External data memory through Register-Indirect
- Program memory look-up tables through Base-Register plus Index-Register-Indirect.

The members of the TELX family are classified as 8-bit devices since their internal ROM, RAM, Special Function Registers, Arithmetic Logic Unit and external data bus are all 8-bits wide. All perform operations on bit, nibble, byte and double-byte data types.

Facilities are available for byte transfer, logic and integer arithmetic operations. Data transfer, logic and conditional branch operations can be performed directly on Boolean variables to provide excellent bit handling.



(1) The upper limit is device dependent.

Fig.3 Memory map.

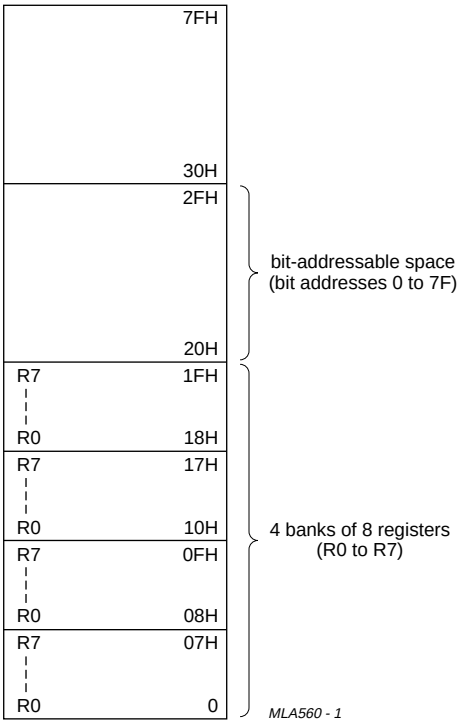


Fig.4 The lower 128 bytes of internal RAM.

Table 10 Special Function Register memory map (bit addressing)

REGISTER MNEMONIC	START ADDRESS	REGISTER BIT								END ADDRESS
		0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
IP1	F8H	PX2	PX3	PX4	PX5	PX6	PX7	PX8	PX9	FFH
B	F0H	B0	B1	B2	B3	B4	B5	B6	B7	F7H
IEN1	E8H	EX2	EX3	EX4	EX5	EX6	EX7	EX8	EX9	EFH
ACC	E0H	ACC0	ACC1	ACC2	ACC3	ACC4	ACC5	ACC6	ACC7	E7H
S1CON	D8H	CR0	CR1	AA	SI	STO	STA	ENS1	CR2	DFH
PSW	D0H	P	–	0V	RS0	RS1	F0	AC	CY	D7H
T2CON	C8H	CP/$\overline{RL2}$	C/$\overline{T2}$	TR2	EXEN2	ECOMP	COMP	EXF2	TF2	CFH
IRQ1	C0H	IQ2	IQ3	IQ4	IQ5	IQ6	IQ7	IQ8	IQ9	C7H
IP0	B8H	PX0	PT0	PX1	PT1	PS0	PS1	PT2	–	BFH
P3	B0H	P30	P31	P32	P33	P34	P35	P36	P37	B7H
IEN0	A8H	EX0	ET0	EX1	ET1	ES0R	ES1	ET2	EA	AFH
P2	A0H	P20	P21	P22	P23	P24	P25	P26	P27	A7H
S0CON	98H	RI	TI	RB8	TB8	REN	SM2	SM1	SM0	9FH
P1	90H	P10	P11	P12	P13	P14	P15	P16	P17	97H
TCON	88H	IT0	IE0	IT1	IE1	TR0	TF0	TR0	TF1	8FH
P0	80H	P00	P01	P02	P03	P04	P05	P06	P07	87H

4.7 I/O facilities

4.7.1 PORTS

The TELX family has 32 I/O lines treated as 32 individually addressable bits or as four parallel 8-bit addressable ports. Ports 0 to 3 perform the following alternative functions.

Port 0 Provides the multiplexed low-order address and data bus for expanding the device with standard memories and peripherals.

Port 1 Used for a number of special functions:

- Provides the inputs for the external interrupts INT2 to INT9
- External inputs of Timer 2
- External activation and compare output of Timer 2
- Real-Time Clock output (16 kHz)
- DTMF melody output
- CLK/P1.4 for the clock output
- SCL/P1.6 and SDA/P1.7 for the I^2C-bus interface are open-drain outputs.

Port 2 Provides the high-order address bus when expanding the device with external program memory and/or external data memory.

Port 3 Pins can be configured individually to provide:

- Serial port receiver input and transmitter output (UART)
- External interrupt request inputs
- Counter inputs
- Control signals to read and write to external memories.

To enable a port pin alternative function, the port bit latch in its SFR must contain a logic 1.

Each port consists of a latch (SFRs P0 to P3), an output driver and input buffer. All ports have internal pull-ups. Figure 5(a) shows that the strong transistor p1 is turned on for only 1 oscillator period after a LOW-to-HIGH transition in the port latch. When on, it turns on p3 (a weak pull-up) through the inverter IN1. This inverter and p3 form a latch which holds the logic 1.

The Alternative Port Function Register (ALTP) is described in Section 4.10.4.

4.7.2 PORT I/O CONFIGURATION

I/O port output configurations are determined by the settings in the port configuration SFRs. Each port has two associated SFRs: PnCFGA and PnCFGB, where ‘n’ indicates the specific port number (0 to 3). One bit in each of the 2 SFRs relates to the output setting for the corresponding port pin, allowing any combination of the 2 output types to be mixed on those port pins. For example, the output type of Port 1 pin 3 is controlled by setting bit 3 in the SFRs P1CFGA and P1CFGB.

The port pins may be individually configured with one of the following modes (P1.6 and P1.7 can be open-drain or high-impedance but never have any diodes against V_{DD}).

Mode 0 Open-drain; quasi-bidirectional I/O with n-channel open-drain output. Use as an output (e.g. Port 0 for external memory accesses ($\overline{EA}$ = 0) or access above the built-in memory boundary) requires the connection of an external pull-up resistor. The ESD protection diodes against V_{DD} and V_{SS} are still present. Except for the I^2C-bus port (P1.6 and P1.7), ports which are configured as open-drain still have a protection diode to V_{DD}. See Fig.5(a).

Mode 1 Standard Port; quasi-bidirectional I/O with pull-up. The strong pull-up p1 is turned on for only two oscillator periods after a LOW-to-HIGH transition in the port latch. After these two oscillator periods the port is only weakly driven through p2 and ‘very weakly’ driven through p3. See Fig.5(b).

Mode 2 High-impedance; this mode turns all port output drivers off. Thus, the pin will not source or sink current and may be used as an input-only pin with no internal drivers for an external device to overcome. See Fig.5(c).

Mode 3 Push-pull; output with drive capability in both polarities. In this mode, pins can only be used as outputs. See Fig.5(d).

Tables 11 and 12 show the configuration register settings for the four output configurations.

The electrical characteristics of each output type may be found in the DC characteristics in the specific product data sheet.

The default port configuration after reset is also given in the specific product data sheet.

Table 11 Selection of the port output configuration

MODE	PnCFGA	PnCFGB	PORT OUTPUT CONFIGURATION	
			NORMAL PORTS	I^2C-BUS PORTS (P1.6 AND P1.7)
0	0	0	open-drain	open-drain (port data and I^2C-bus output)
1	1	0	quasi-bidirectional	open-drain (port data only)
2	0	1	high-impedance	high-impedance
3	1	1	push-pull	open-drain (port data only)

Note

1. If P1CFGA.7 is set the I^2C-bus interfaces of the microcontroller and other on-chip blocks with an I^2C-bus interface (e.g. EEPROM) are connected internally. This means that the microcontroller can access these blocks via the I^2C-bus without using P1.6 and P1.7.

Table 12 Special Function Registers for port configurations; note 1

REGISTER NAME	REGISTER MNEMONIC	SFR ADDRESS (HEX)	STATE AFTER RESET
Port P0 Configuration A Register	P0CFGA	8E	depending on product, refer to product specification
Port P0 Configuration B Register	P0CFGB	8F	
Port P0 output data Register	P0	80	
Port P1 Configuration A Register	P1CFGA	9E	
Port P1 Configuration B Register	P1CFGB	9F	
Port P1 output data Register	P1	90	
Port P2 Configuration A Register	P2CFGA	AE	
Port P2 Configuration B Register	P2CFGB	AF	
Port P2 output data Register	P2	A0	
Port P3 Configuration A Register	P3CFGA	BE	
Port P3 Configuration B Register	P3CFGB	BF	
Port P3 output data Register	P3	B0	

Note

1. Mode changes may cause glitches to occur during transitions. When modifying both registers, WRITE instructions should be carried out consecutively.

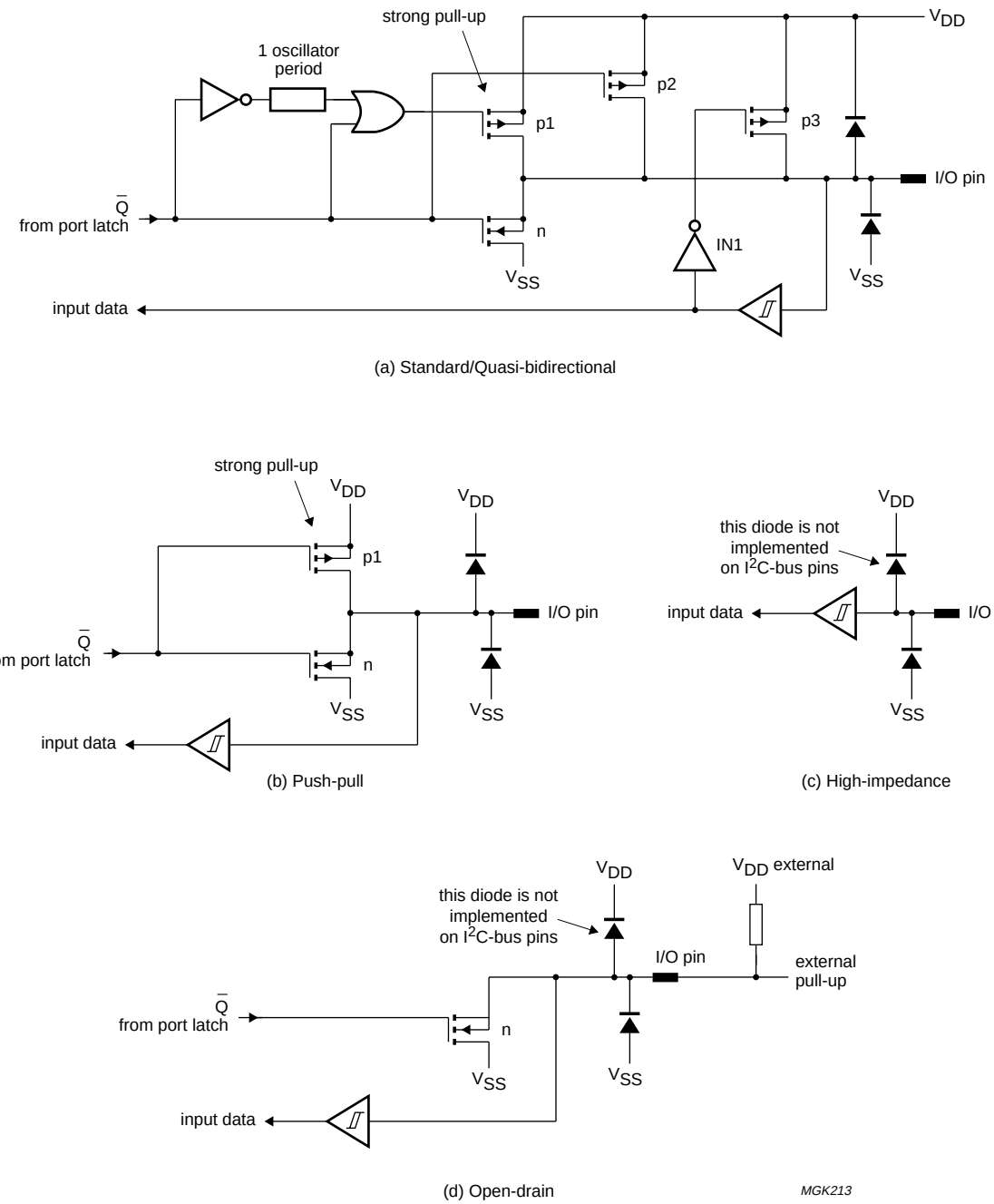


Fig.5 Port configuration options.

4.8 Timer/event counters

The TELX family contains three 16-bit timer/event counters: Timer 0, Timer 1 and Timer 2 which perform the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests
- Generate output on comparator match
- Generate a pulse width modulated output signal.

Timer 0 and Timer 1 can be programmed independently to operate in four modes:

Mode 0 8-bit timer or 8-bit counter each with divide-by-32 prescaler

Mode 1 16-bit time-interval or event counter

Mode 2 8-bit time-interval or event counter with automatic reload upon overflow

Mode 3 Timer 0 establishes TL0 and TH0 as two separate counters.

In the timer mode the register is incremented every machine cycle. Since a machine cycle consists of a minimum of 6 oscillator periods, the maximum count rate is $\frac{1}{6} \times f_{osc}$.

In the counter mode, the register is incremented in response to a HIGH-to-LOW transition. Since it takes one machine cycle (minimum 6 oscillator periods) to recognize a HIGH-to-LOW transition, the maximum count rate is $\frac{1}{6} \times f_{osc}$. To ensure a given level is sampled, it should be held for at least one complete machine cycle.

4.8.1 TIMER T2

Note that the function of Timer 2 may deviate from the following description for certain products in the TELX family. In such a case, the deviation is described in the specific product data sheet.

Timer 2 is a 16-bit timer/counter that can operate as a timer, as an event counter or as a pulse width modulator.

The following operating modes are available: External interrupt, T2-only, Auto-Reload and Capture mode.
If Timer 2 is in the OFF state, its clock is switched off and the timer has an extremely low power consumption.
Parallel to these operating modes, a Compare function and/or a pulse generator function is provided.

The operating modes are selected via the T2CON bits TR2, CP/$\overline{RL2}$ and EXEN2 (see Table 13).

Table 13 Selection of Timer 2 operating modes

TR2	EXEN2	CP/$\overline{RL2}$	MODE
0	0	X	OFF
0	1	X	External interrupt
1	X	0	Auto-reload
1	0	1	T2-only
1	1	1	Capture

In the T2-only mode, TH2 and TL2 function as a 16-bit timer or counter which upon overflowing sets the Timer 2 overflow bit TF2. This may then be used to generate an interrupt.

In the Capture mode, TH2 and TL2 function as a 16-bit timer or counter which upon overflowing sets the Timer 2 overflow bit TF2. This may then be used to generate an interrupt. Additionally a HIGH-to-LOW transition at external input T2EX causes the current value in TL2 and TH2 to be captured into registers RCAP2L and RCAP2H respectively. In addition, the transition at T2EX causes the EXF2 interrupt flag in T2CON to be set, this may also be used to generate an interrupt. The Capture mode and the T2-only mode are shown in Fig.6.

In the Auto-Reload mode the 16-bit counter (TH2, TL2) does not continue counting at the value 0000H, after an overflow occurred, but will be reloaded with the 16-bit value stored in the SFRs RCAP2H and RCAP2L. If in Auto-Reload mode, the EXEN2 bit is set, a HIGH-to-LOW transition at external input T2EX will set the EXF2 bit and will also trigger the reloading of TH2,TL2. The Auto-Reload mode is shown in Fig.7.

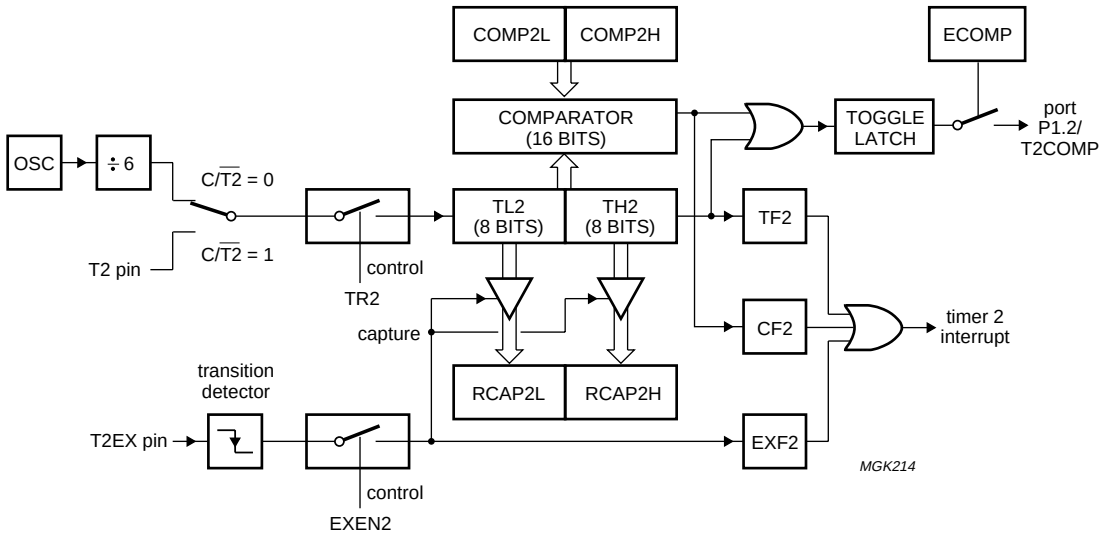


Fig.6 Timer 2 in T2-only and/or Capture mode.

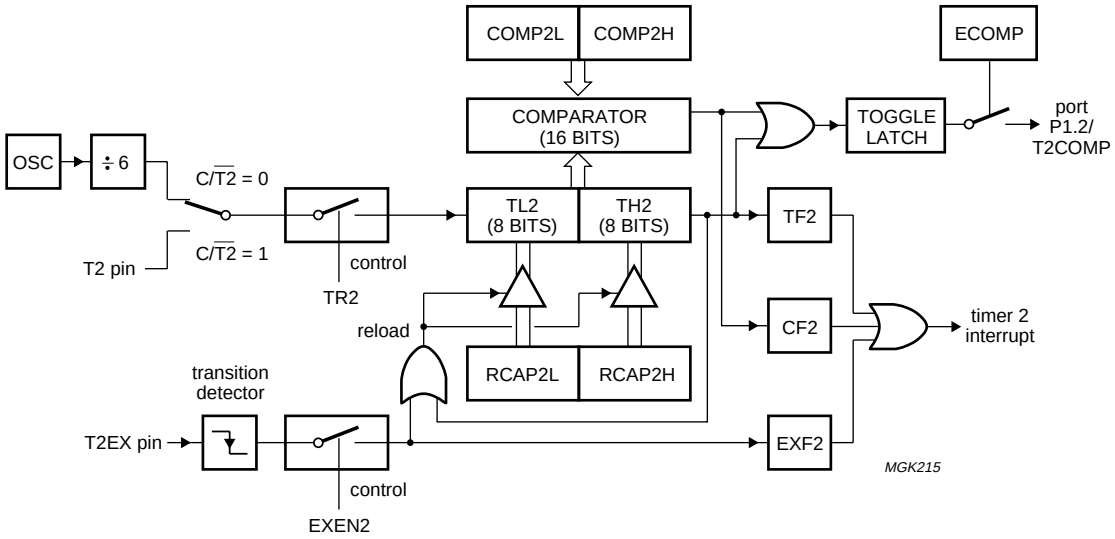


Fig.7 Timer 2 in Auto-Reload mode.

Parallel to the T2-only, Capture and Auto-Reload mode, a compare function can be activated by writing a value other than 0000H to the compare SFRs COMP2H and COMP2L. A compare match is generated when the timer register TL2/TH2 increments to the value of the compare register COMP2L/COM2H. A compare match will set the compare flag CF2, this may also be used to generate an interrupt.

Parallel to the T2-only, Capture and Auto-Reload mode, a Pulse Width Modulation function can be activated by setting the ECOMP bit in the T2CON register. This will activate the alternative port function T2COMP for port bit P1.2. Every time a compare match or a timer overflow occurs, P1.2 (T2COMP) is toggled. The initial state of P1.2 after setting ECOMP is LOW.

If this pulse function is used in conjunction with the Auto-Reload mode and the compare function, a Pulse Width Modulation (PWM) function is realized. The PWM frequency is given by the reload value stored in register RCAP2L/RCAP2H. The PWM duty cycle is given by the value stored in register COMP2L/COMP2H. In Fig.8 an example of this is given with a 25/75% duty cycle.

As a special case, if both registers COMP2H and COMP2L are reset, the frequency on pin P1.2/T2COMP will be given only by the value of the reload register RCAP2L/RCAP2H and is half the frequency for an active compare. The duty cycle will be 50% as shown in Fig.9.

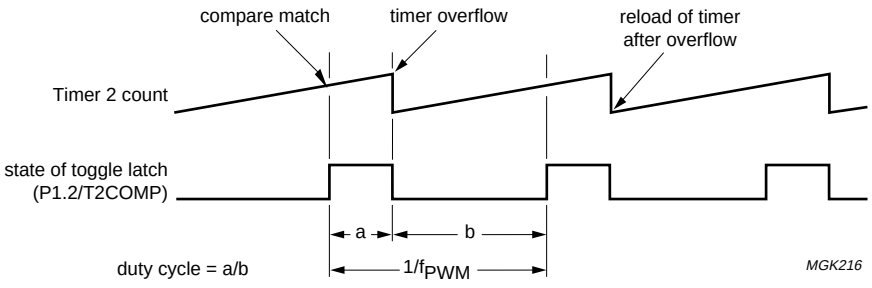


Fig.8 Timer 2: PWM output function with duty cycle 25/75%.

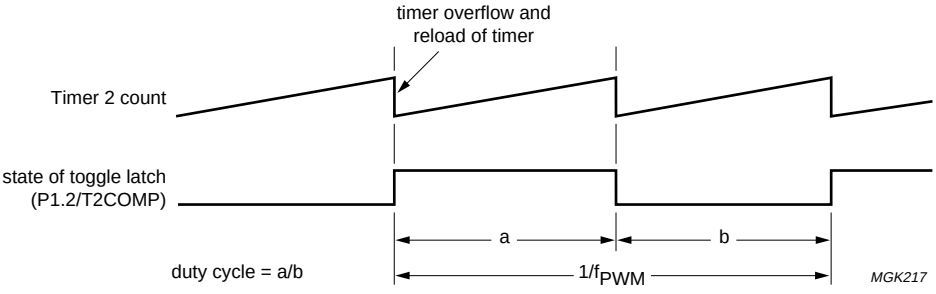


Fig.9 Timer 2: PWM output function with duty cycle 50/50%.

4.8.2 TIMER/COUNTER 2 CONTROL REGISTER (T2CON)

Table 14 Timer/Counter 2 Control Register (SFR address C8H)

7	6	5	4	3	2	1	0
TF2	EXF2	CF2	ECOMP	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$

Table 15 Description of T2CON bits

BIT	SYMBOL	DESCRIPTION
T2CON.7	TF2	**Timer 2 overflow flag.** TF2 is set by a Timer 2 overflow and must be cleared by software. When Timer T2 interrupt is enabled, TF2 = 1 will cause the CPU to vector to Timer 2 interrupt routine. TF2 must be cleared by software. TF2 can also be set by software.
T2CON.6	EXF2	**Timer 2 external flag.** EXF2 is set when either a capture or reload is caused by a negative transition on T2EX and when EXEN2 = 1. When Timer T2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to Timer 2 interrupt routine. EXF2 must be cleared by software. EXF2 can also be set by software.
T2CON.5	CF2	**Compare flag.** CF2 is set by hardware if the timer register TL2/TH2 increments to the value of the compare register COMP2L/COMP2H. CF2 must be cleared by software. When Timer T2 interrupt is enabled, CF2 = 1 will cause the CPU to vector to Timer 2 interrupt routine. CF2 must be cleared by software. CF2 can also be set by software.
T2CON.4	ECOMP	**Enable compare.** When set by software, the controller toggles port bit P1.2 (T2COMP) every time a compare match or a timer overflow occurs. The toggle latch connected to port P1.2/T2COMP is reset when bit ECOMP is cleared.
T2CON.3	EXEN2	**Timer 2 external enable flag.** When set, allows a capture or reload to occur as a result of a negative transition on T2EX. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.
T2CON.2	TR2	**Timer 2 start/stop control.** If TR2 is set, the 16-bit counter (TH2,TL2) will start counting.
T2CON.1	C/$\overline{T2}$	**Timer 2 timer or counter select.** C/$\overline{T2}$ = 0 selects the internal timer with a clock frequency of $\frac{1}{6} \times f_{osc}$. C/$\overline{T2}$ = 1 selects the external event counter; negative edge-triggered.
T2CON.0	CP/$\overline{RL2}$	**Capture/reload flag.** When set captures will occur on negative transitions at T2EX, if EXEN2 = 1. When cleared, auto-reloads will occur either with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1.

4.8.3 WATCHDOG TIMER

In addition to the standard timers and Timer 2, a Watchdog Timer consisting of an 13-bit prescaler and an 8-bit timer WDTIM is also incorporated. The prescaler is incremented by the external clock. The 8-bit timer is incremented every 8192 clock cycles.

If the clock frequency is 3.58 MHz, the Watchdog Timer can operate in the range of 2.3 ms up to 0.56 s. The Watchdog Timer is disabled after reset. It can be enabled by writing any value to the WDCON register. A running Watchdog Timer will only be disabled if the microcontroller enters Power-down mode or if the microcontroller is reset.

When a timer overflow occurs and the Watchdog Enable pin ($\overline{EW}$) is LOW, the reset pin ($\overline{RST}$) will be activated (pulled-down) and the microcontroller will be reset. To prevent an overflow of the Watchdog Timer, the user program must reload the Watchdog register within a period shorter than the programmed timer interval.

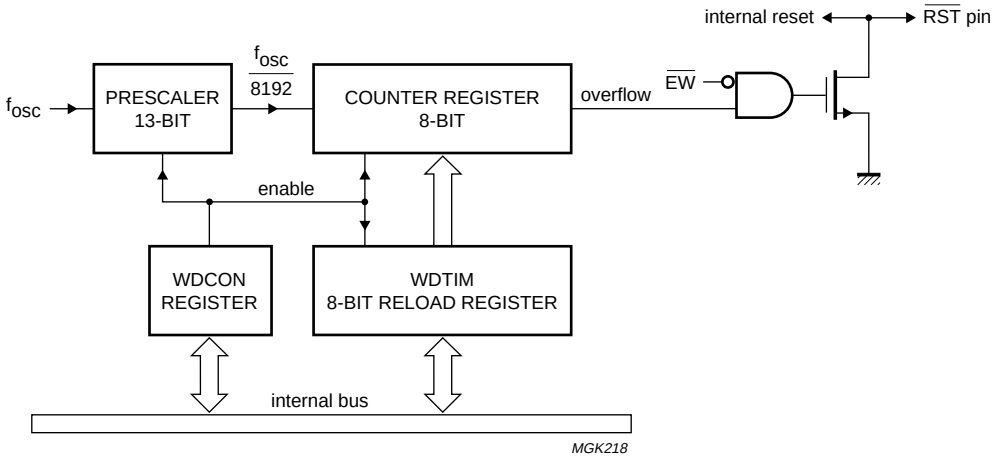


Fig.10 Functional diagram of the Watchdog Timer (T3).

4.8.4 WATCHDOG TIMER INTERVAL REGISTER (WDTIM)

The reset value of WDTIM is 00H. The WDTIM register can only be written to if the WDCON register contains the value 5AH. The Watchdog Timer period can be calculated as follows:

$$\text{Watchdog period} = \frac{(256 - \text{WDTIM}) \times 8192}{f_{osc}}$$

Table 16 Watchdog Timer Interval Register (SFR address FFH)

7	6	5	4	3	2	1	0
WD7	WD6	WD5	WD4	WD3	WD2	WD1	WD0

Table 17 Description of WDTIM bits

BIT	SYMBOL	DESCRIPTION
WDTIM.7 to WDTIM.0	WD7 to WD0	The state of these 8-bits determine the Watchdog Timer period.

4.8.5 WATCHDOG TIMER CONTROL REGISTER (WDCON)

The Watchdog Timer is controlled by the WDCON register. A value of A5H in WDCON clears both the prescaler and the timer WDTIM. After reset WDCON contains the value A5H. Every value other than A5H in WDCON enables the Watchdog Timer. Since the WD0 bit of the WDCON input is tied to a logic 0 by hardware during write operations to WDCON, the reset value A5H can not be programmed again and can only be restored by a reset.

Timer WDTIM can be written only if WDCON has previously been loaded with 5AH, otherwise WDTIM and the prescaler are not affected. A successful write operation to WDTIM also clears the prescaler and clears WDCON.

Only the values A5H and 5AH are stored, all other values are stored with a dummy value 00H.

Table 18 Watchdog Control Register (SFR address A5H)

7	6	5	4	3	2	1	0
WC7	WC6	WC5	WC4	WC3	WC2	WC1	WC0

Table 19 Description of WDCON bits

BIT	SYMBOL	DESCRIPTION
WDCON.7 to WDCON.0	WC7 to WC0	Watchdog Timer control bits

4.8.6 PULSE WIDTH MODULATED OUTPUT

One pulse width modulated output channel is provided which outputs pulses of programmable length and interval. The repetition frequency is defined by an 8-bit prescaler (PWMP) that generates the clock for the counter. The 8-bit counter counts modulo 255 i.e. from 0 to 254 inclusive. The value held in the 8-bit counter is compared to the contents of the register PWM0. Provided the contents of this register are greater than the counter value, the $\overline{PWM0}$ output is set LOW. If the contents of register PWM0 are equal to, or less than the counter value, the $\overline{PWM0}$ output is set HIGH. The pulse-width-ratio is therefore defined by the contents of register PWM0. The pulse-width-ratio will be in the range 0 to 255/255 and may be programmed in increments of $1/255$.

The repetition frequency at the $\overline{PWM0}$ output is given by:

$$f_{PWM} = \frac{f_{osc}}{[(1 + PWMP) \times 255]}$$

When using an oscillator frequency of 3.58 MHz for example, the above formula gives a repetition frequency range of 55 Hz to 14 kHz.

By loading the PWMO register with either 00H or FFH, the $\overline{PWM0}$ output can be maintained at a constant HIGH or LOW level respectively. When loading FFH into the PWM0 register, the 8-bit counter will never actually reach this value.

The $\overline{PWM0}$ output pin is driven by push-pull drivers and is not shared with any other function.

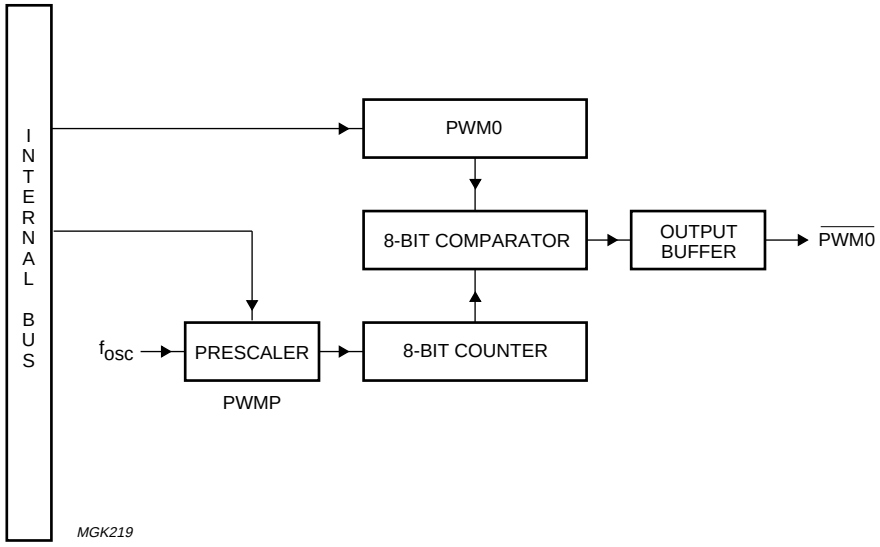


Fig.11 Functional diagram of pulse width modulated output.

4.8.7 Prescaler Frequency Control Register (PWMP)

Table 20 Prescaler Frequency Control Register (SFR address FEH)

7	6	5	4	3	2	1	0
PWMP.7	PWMP.6	PWMP.5	PWMP.4	PWMP.3	PWMP.2	PWMP.1	PWMP.0

Table 21 Description of PWMP bits

BIT	SYMBOL	DESCRIPTION
7 to 0	PWMP7 to PWMP.0	prescaler division factor = (PWMP) + 1

4.8.8 Pulse Width Modulated Register (PWM0)

Table 22 Pulse Width Modulated Register (SFR address FCH)

7	6	5	4	3	2	1	0
PWM0.7	PWM0.6	PWM0.5	PWM0.4	PWM0.3	PWM0.2	PWM0.1	PWM0.0

Table 23 Description of PWM0 bits

BIT	SYMBOL	DESCRIPTION
7 to 0	PWM07 to PWM0.0	These 8-bits define the pulse-width-ratio.

4.9 EEPROM

4.9.1 General description

Most microcontrollers in the TELX family contain an on-chip low-power Electrically Erasable Programmable ROM (EEPROM) memory for non-volatile data storage. The memory offers the following features:

- Low power consumption
- No current consumption if the EEPROM is disabled (see Section 4.9.6)
- Single supply programming; the programming voltage is generated internally via an on-chip voltage multiplier
- Automatic ERASE before WRITE when programming
- User defined programming time (see Section 4.9.6)
- Page programming; 1 to 8 bytes can be programmed simultaneously, reducing programming time
- Accessible via I^2C-bus:
  - Fixed slave address
  - Operates in Slave Transmitter or Slave Receiver modes
  - Can be accessed from a master connected to the external I^2C-bus (EEPROM access in external mode)
  - Can be accessed from the host master even if ports P1.6 and P1.7 are not used as I^2C-bus pins (EEPROM access in internal mode)
  - Supports continuous read and page-write, (word address automatically incremented).

4.9.2 I^2C-bus operation

The operation of the EEPROM memory depends on the state of the I^2C-bus interface (see Section 4.12) and of the port pins P1.6 and P1.7 (see Section 4.7.2). Three situations are possible:

1. **EEPROM access in internal mode.** The I^2C-bus serial I/O interface and the EEPROM memory are active, but the port pins P1.6 and P1.7 are not used as I^2C-bus pins.

   The CPU of the TELX microcontroller can program and read the EEPROM. Port pins P1.6 and P1.7 can be used as open-drain ports for other purposes.
2. **EEPROM access in external mode.** The I^2C-bus serial I/O master is not active, port pins P1.6 and P1.7 are configured as I^2C-bus pins, and the EEPROM is active.

   The EEPROM can be accessed from a master connected to the I^2C-bus (see Fig.13), but not from the TELX CPU.
3. **EEPROM access in mixed mode**. Both the serial I/O interface and EEPROM are active, P1.6 and P1.7 are configured as I^2C-bus pins.

   Both the CPU of the TELX and external master(s) can read/programme the EEPROM.

After reset, the I^2C-bus is in internal mode. In external mode, I^2C-bus pull-up resistors must be connected to P1.6 and P1.7.

Table 24 EEPROM modes of operation

MODE	CONTROL REGISTER					
	PnCFGA(1)		PnCFGB(1)		S1CON(2)	EECON(3)
	P1CFGA.6	P1CFGA.7	P1CFGB.6	P1CFGB.7	ENS1	EEPE
EEPROM disabled(4)	X	X	X	X	X	0
Internal mode	X	1	X	X	1	1
External mode	0	0	0	0	0	1
Mixed mode	0	0	0	0	1	1

Notes

1. See Section 4.7.2, Table 11.
2. See Section 4.12.
3. See Section 4.9.6.
4. When disabled, the EEPROM will not acknowledge any I^2C-bus request, and consumes no power.

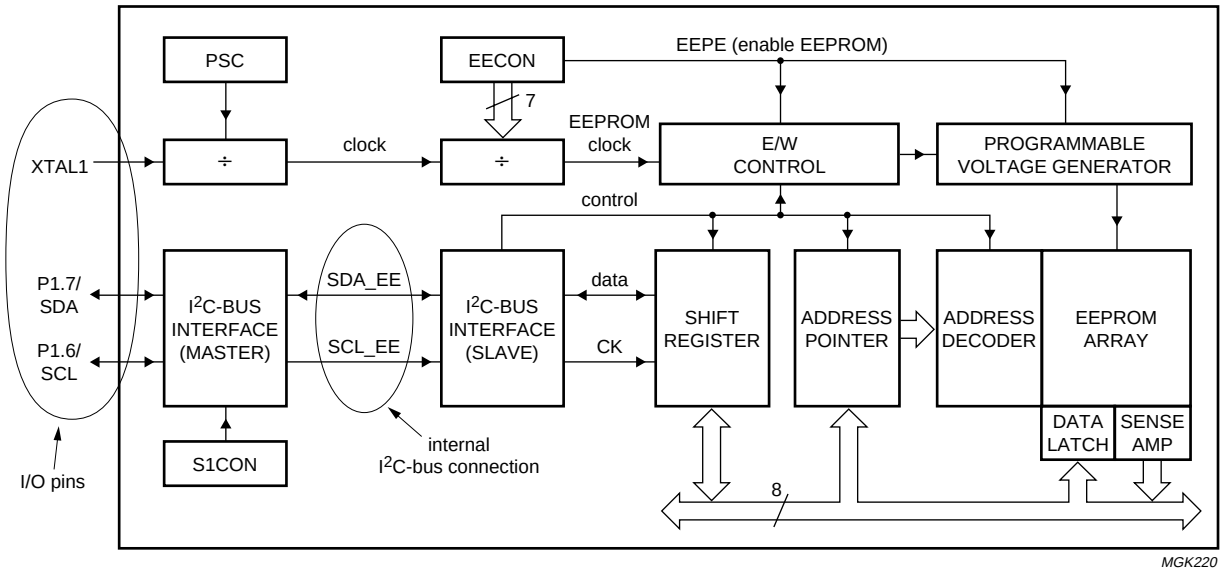


Fig.12 Block diagram of the EEPROM with I²C-bus interface.

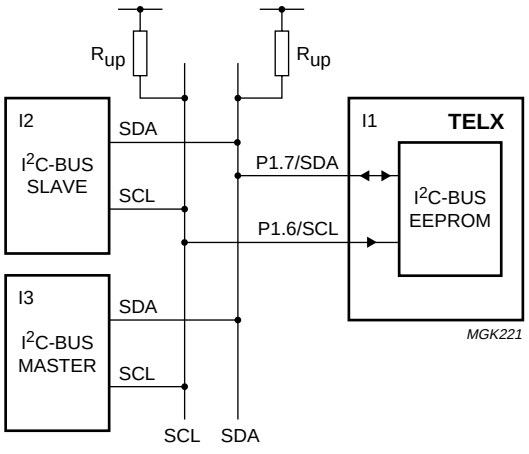


Fig.13 EEPROM access in external mode. The device I3 can access the EEPROM of the TELX.

4.9.3 EEPROM Addressing And Operation

The EEPROM is accessed with an I²C Start (S) condition, followed by a 7-bit slave address and a control bit (R/$\overline{W}$). Upon successful decoding of the address, the EEPROM answers with an I²C Acknowledge (A). Figure 14 shows the slave addresses for the different EEPROM sizes.

In microcontrollers with a 512-byte EEPROM, the 7th bit (A8) sent after the 6-bit slave address is part of the word address (A8 is the most significant address bit of the 512-byte EEPROM array).

The last bit of the slave address (R/$\overline{W}$) defines the operation to be performed. When set to logic 1 a read operation is selected (the EEPROM will output the addressed data onto SDA at every SCL pulse), and when set to logic 0 the EEPROM will be ready to accept 7 bits of EEPROM address, possibly followed by data bytes to be stored in the EEPROM.

The master can abort any Read or Write operation at any time during I²C-bus data transfer by generating a new Start (S) without generating a Stop (P) condition.

4.9.4 Write Operations

4.9.4.1 Byte Write

After addressing the EEPROM with the R/$\overline{W}$ bit set to a logic 0, the EEPROM responds with an acknowledge and expects to receive a word address, followed by a byte of data to be written. In the case of a 512-byte EEPROM, the bit before R/$\overline{W}$ is the MSB of the word address of the byte to be written (A8). The master then sends the word address (A0 to A7), to which the EEPROM sends an acknowledge (A). Finally the master sends the data to be written, acknowledged by the EEPROM. The master sends a Stop condition (P) to start an Erase/Write cycle. The cycle takes typically 10 ms and is controlled by the E/W control circuitry (see Fig.12). The byte write sequence is shown in Fig.15, for the case n = 1.

Note that a Write to the EEPROM is implemented as a logical OR with the previously stored data; a Write operation must therefore be preceded with an Erase to clear the byte first. The E/W control logic will automatically generate the necessary Erase followed by the Write when a Stop condition is generated. The write time is specified for the complete Erase/Write cycle.

During the Erase/Write cycle the I²C-bus interface of the EEPROM is idle, i.e., it does not acknowledge when addressed (see also Section 4.9.4.3).

4.9.4.2 Page Write

In order to reduce total programming time when several bytes of data are to be written to the EEPROM, a page-write operation is available. Up to 8 bytes of data can be programmed with a single Erase/Write cycle, as long as all bytes are on the same page, i.e., their addresses only differ on the 3 lowest bits (A0 to A2). The sequence is similar to the byte-write: the master sends a Start (S) and slave address with the R/$\overline{W}$ bit set to logic 0, followed by the word address of the first data byte to be programmed.

Then the first data byte is sent, and instead of immediately generating a Stop condition, the master sends up to 7 additional bytes; the 3 lowest bits of the address are automatically incremented, the highest bits remain fixed. The EEPROM acknowledges each data byte. Finally a Stop (P) is generated to start an Erase/Write cycle. This sequence is shown Fig.15.

Any number of bytes from 1 to 8 can be written, but their low addresses (A0 to A2) must be sequential. When the page addresses reaches end-of-page (A0 to A2 = 111), the address will wrap around to ‘000’ (binary). Fig.16 shows two examples of possible page set-up; the first example shows eight bytes written starting from the beginning of the page (address 00H), and the second example shows six bytes written starting in the middle of the page, at address 15H.

If more then 8 bytes are sent by the master, the EEPROM will ignore and will **not** acknowledge the 9th, 10th etc., bytes. The master can proceed in one of two ways:

- Abort the write procedure, by sending a Start (S) and repeating the complete page-write procedure of Fig.15
- Start and Erase/Write cycle by generating a Stop (P). The first 8 bytes transmitted will be written into the EEPROM cells.

4.9.4.3 Acknowledge Polling

During programming, the EEPROM does not acknowledge when addressed by an I²C-bus master. To find out when the EEPROM is again accessible, the microcontroller must perform ACK polling, i.e. repeatedly send a Start and slave address and check if an acknowledge is generated.

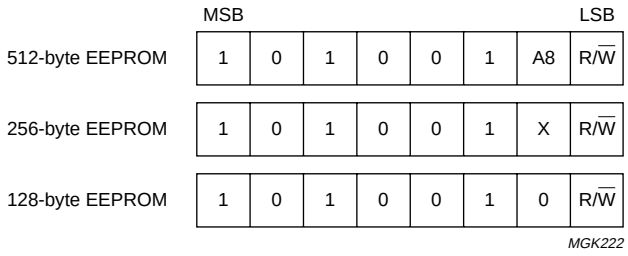


Fig.14 Slave address format for the EEPROM.

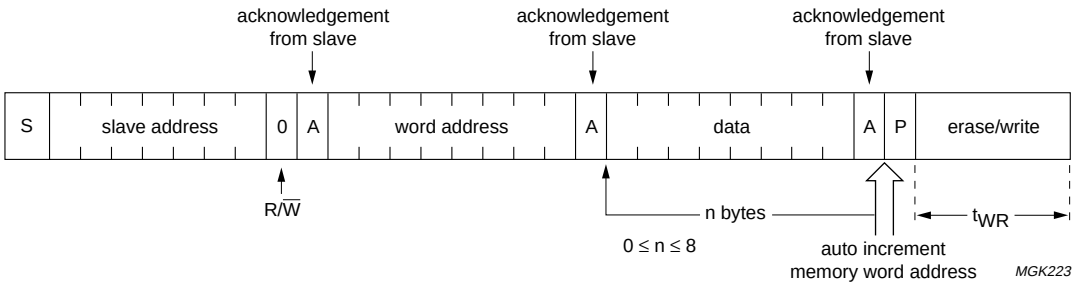


Fig.15 Master transmits to slave receiver (Write mode).

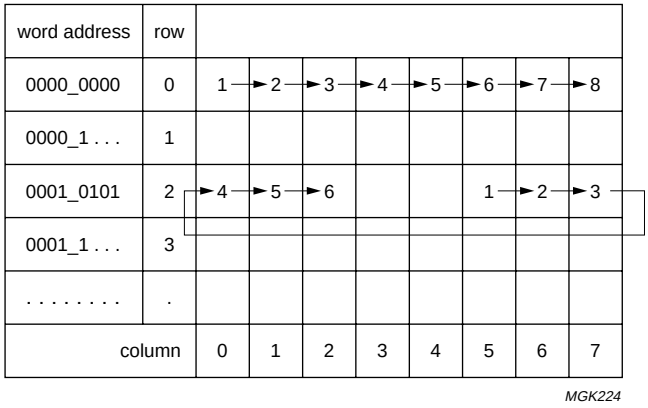


Fig.16 Two examples of writing eight or six bytes with different word addresses.

4.9.5 READ OPERATIONS

An unlimited number of data bytes can be read, the address being automatically incremented after each byte is transmitted. A Read can be done in two ways: by first setting the word address (Random Read), or without setting the word address (Current Address Read). Both ways allow to sequentially read any number of bytes (Sequential Read).

4.9.5.1 Current Address Read

The master addresses the EEPROM slave with the R/$\overline{W}$ bit set to a logic 1. The EEPROM acknowledges, transmits the data byte addressed by the current contents of the address pointer, and increments it by 1. The master ends the read operation by generating a No-acknowledge (A = 1) and Stop (P).

If the master wishes to read more than one byte (Sequential Read), it generates an Acknowledge (A = 0) after receiving the data byte, and does not generate a Stop. Any number of bytes can be read with this procedure; the address pointer will wrap-around to address 00H when the highest address is read. To end Sequential Read, the master generates a No-acknowledge (A = 1) and a Stop (P). Figure 17 illustrates the Current Address Read and Sequential Read procedure.

When using the Current Address Read, the contents of the address pointer are equal to the address of the byte previously accessed (either by a previous read or write) incremented by 1; e.g., if the previous action was writing or reading byte addressed by ‘n’, using Current Address Read will retrieve the byte addressed by ‘n + 1’.

Note there is an exception to the above rule: when using Current Address Read with 512-byte EEPROMs, the MSB of the current read address (A8) is overwritten each time the Slave address is sent. For example, if the byte previously accessed was addressed by 000H, and the EEPROM is selected using A8 = 1, the data retrieved will be in address 101H (A8 = 1, A0 to A7 incremented by 1) and not 001H.

4.9.5.2 Random Read

In Random Read mode, the address of the byte to be read is sent prior to the read. The master selects the EEPROM with the R/$\overline{W}$ bit set to logic 0 (write), and upon Acknowledge from the EEPROM sends the 8-bit word address, which is loaded into the Address pointer and acknowledged by the EEPROM. The master then sends a Repetitive Start (Sr, a Start without previously having generated a Stop) selecting again the slave with the R/$\overline{W}$ bit to logic 1 (read). The EEPROM transmits the byte addressed by the address pointer and increments it at the end. The master ends Random Read by generating a No-acknowledge (A = 1) and a Stop (S).

If the master wishes to read more bytes, it generates an Acknowledge (A = 0) after receiving each byte (Sequential Read). Any number of bytes can be read; the address pointer will wrap-around to address 00H when the highest address is read. To end Sequential Read, the master generates a No-acknowledge (A = 1) and a Stop (P). Figure 18 shows the Random Read and Sequential Read procedure.

Note when using Random Read with EEPROM memories with 512-bytes, the MSB of the word address (A8) sent during the Write frame is overwritten when the slave is addressed a second time, after the Repetitive Start (Sr). For example, if the word-address sent is 000H (A8 = 0), and the slave address after the Repetitive Start contains A8 = 1, then the data retrieved is addressed by 100H and not 000H.

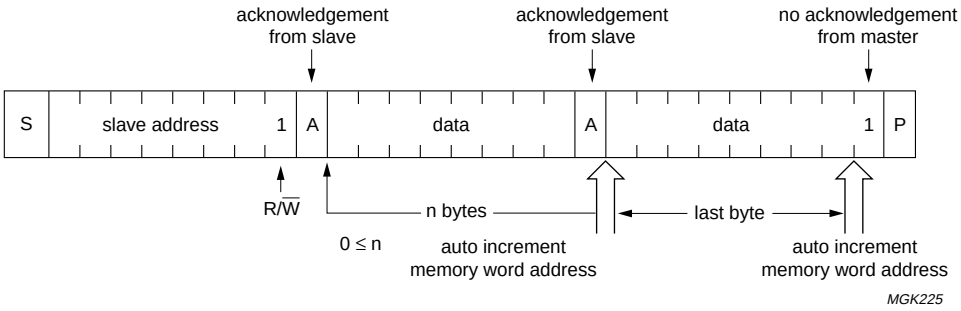


Fig.17 Current Address Read.

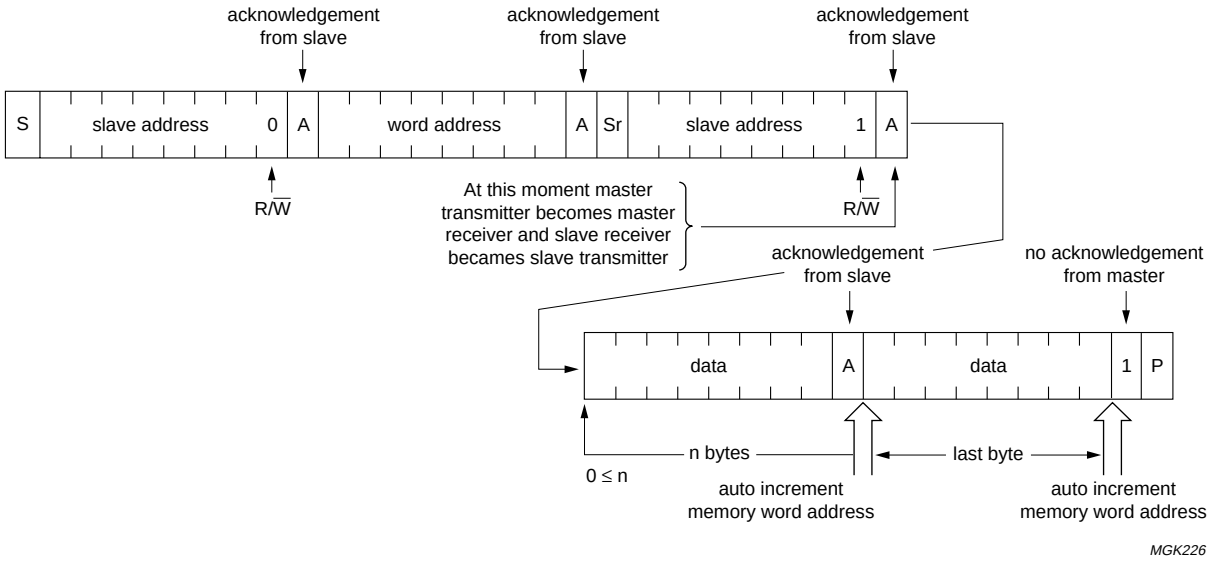


Fig.18 Random Read.

4.9.6 EEPROM CONTROL REGISTER (EECON)

This Special Function Register controls the operation of the EEPROM. The programming time (Erase/Write cycle) is defined by the crystal frequency, the pre-scaler division factor and the EECON bits (EEC0 to EEC6). Erase and Write operations take the same time. Typically 4 to 5 ms are necessary to erase or write the EEPROM cells giving a total Erase/Write cycle time of 8 (minimum) to 10 ms.

To achieve these programming times, the E/W control circuit requires a clock (EEPROM clock, see Fig.12) with a frequency between 51 and 63.75 kHz (max.). Frequencies lower than 51 kHz are acceptable, but result in a programming time greater than 10 ms. The frequency of XTAL is divided by the contents of the PSC register to define the internal clock frequency. This clock is then divided by the contents of the EEC0 to EEC6 bits to generate the 51 kHz EEPROM clock. The Erase and Write times are obtained by further dividing this clock by 255.

The total Erase/Write time is given by the relationship shown below:

$$t_{WR} = 2 \times 255 \times \frac{XTAL1}{PSC \times EECON}$$

To determine the EECON value given the XTAL frequency and the PSC factor the relationship shown below should be used:

$$EECON = \frac{XTAL1 \times t_{WR}}{PSC \times 510}$$

Note that EECON can only take integer values between 2 and 127. Table 27 shows some examples of XTAL frequencies, Prescaler Division Factor and EECON values and the resulting Erase/Write times.

Table 25 EEPROM Control Register (see Section 4.6 for the SFR addressing)

7	6	5	4	3	2	1	0
EEPE	EEC6	EEC5	EEC4	EEC3	EEC2	EEC1	EEC0

Table 26 Description of EECON bits

BIT	SYMBOL	DESCRIPTION
7	EEPE	**EEPROM Enable.** When EEPE = 1, the EEPROM is enabled and can be read and written to. When EEPE = 0, the EEPROM is disabled and in this state consumes no power; reading and writing operations are not possible and the I^2C-bus interface will not acknowledge any request.
6 to 0	EEC6 to EEC0	The decimal value of these 7-bits determines the divider value for the EEPROM programming clock.

Table 27 EEPROM division factor examples

XTAL FREQUENCY (MHz)	PSC DIVISION FACTOR	f_{PSC} (kHz)	EECON VALUE(1)	EEPROM CLOCK FREQUENCY (kHz)	E/W TIME (ms)
30	8	3.75	65 (1100 0001)	57.7	8.84
20	6	3.33	57 (1011 1001)	58.4	8.73
8.0	2	4.00	69 (1100 0101)	57.8	8.80
3.58	1	3.58	62 (1011 1110)	57.7	8.84
1.0	1	1.00	17 (1001 0001)	58.8	8.67
0.102	1	0.102	2 (1000 0010)	51.0	10.0

Note

1. The EECON division value is in decimal notation and between brackets in binary. The EEPE bit is set to a logic 1 in all cases. The smallest EECON VALUE is 2 (1000 0010). EECON VALUE = 1 (1000 0001) or 0 (1000 0000) generates no EEPROM clock and should not be used.

4.10 DTMF generator section

A versatile frequency generator section is provided and is shown in Fig.19. For normal operation use a 3.579545 MHz (or a multiple of this frequency) quartz crystal or PXE resonator. The frequency generator includes precision circuitry for dual-tone multi-frequency (DTMF) signals, which is typically used for tone dialling telephone sets. The TONE output can alternatively issue twelve modem frequencies for data rates between 300 and 1200 bits/s. In addition to DTMF and modem frequencies, two octaves of musical scale in steps of semitones are available.

4.10.1 Frequency registers

The two frequency registers (LGF and HGF) define two frequencies and from these, the digital sine wave synthesizers together with the Digital-to-Analog Convertors construct two sine waves. Their amplitudes are precisely scaled according to the bandgap voltage reference. This ensures tone output levels independent of supply voltage and temperature.

The amplitude of the Low Group Frequency sine wave is attenuated by 2 dB compared to the amplitude of the High Group Frequency sine wave.

The two sine waves are summed and then filtered by on-chip switched capacitor and RC low-pass filters. These guarantee all DTMF tones generated fulfil the CEPT CS203 recommendations with respect to amplitude, frequency deviation, total harmonic distortion and suppression of unwanted frequency components. A value of 00H in a frequency register stops the corresponding digital sine wave synthesizer. If both frequency registers contain 00H, the whole frequency generator is shut off, resulting in lower power consumption. A decimal value of ‘x’ in a frequency register yields a digital sine wave signal with frequency:

$$f = \frac{f_{xtal}}{[23\,(x+2)]}\ ;\ \text{where } 60 \le x \le 255$$

The frequency limitation given by $x \geq 60$ is due to the low-pass filters which would attenuate higher frequency sine waves.

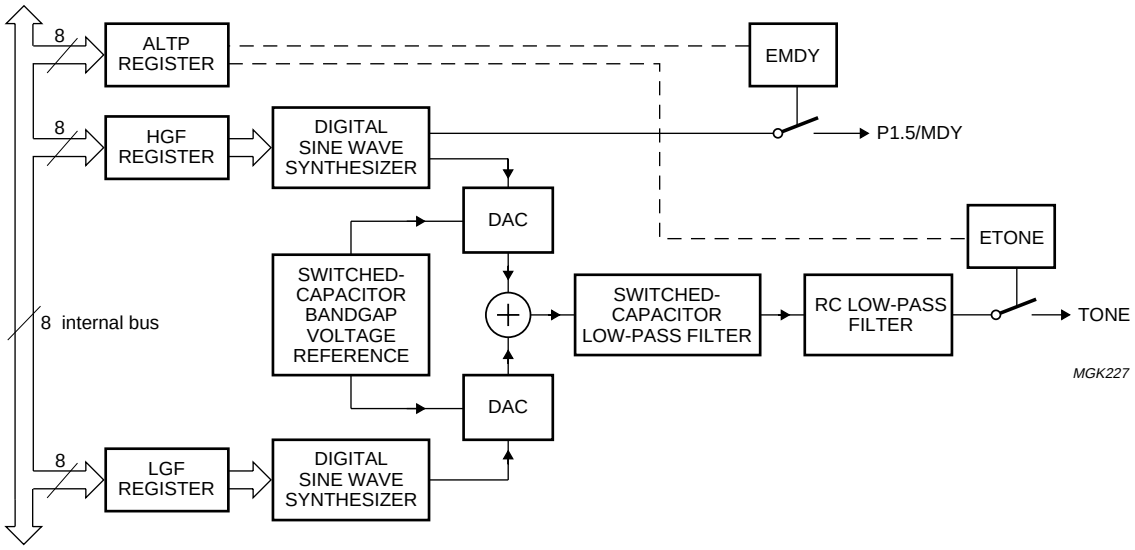


Fig.19 Block diagram of the Frequency Generator section.

4.10.2 Low Group Frequency Register (LGF)

Table 28 Low Group Frequency Register (address A1H; access type W)

7	6	5	4	3	2	1	0
L7	L6	L5	L4	L3	L2	L1	L0

Table 29 Description of LGF bits

BIT	SYMBOL	DESCRIPTION
7 to 0	L7 to L0	The decimal value of these 8-bits determines the Low Group Frequency.

4.10.3 High Group Frequency Register (HGF)

Table 30 High Group Frequency Register (address A2H; access type W)

7	6	5	4	3	2	1	0
H7	H6	H5	H4	H3	H2	H1	H0

Table 31 Description of HGF bits

BIT	SYMBOL	DESCRIPTION
7 to 0	H7 to H0	The decimal value of these 8-bits determines the High Group Frequency.

4.10.4 Alternative Port Function Control Register (ALTP)

Table 32 Alternative Port Control Register (address A3H; access type W)

7	6	5	4	3	2	1	0
–	–	–	–	–	ECLK	EMLDY	ETONE

Table 33 Description of ALTP bits

BIT	SYMBOL	DESCRIPTION
7 to 3	–	These 5-bits are not used.
2	ECLK	**Enable Clock.** When ECLK = 1, P1.4 will output the system clock.
1	EMLDY	**Enable MLDY.** When EMLDY = 1, the MLDY output is enabled (multiplexed with P1.5).
0	ETONE	**Enable Tone.** When ETONE = 1, the TONE output is enabled. When ETONE = 0, the TONE output is disabled (default after reset) however the MLDY output can still be active if required.

4.10.5 DTMF FREQUENCIES

The input frequency to the frequency generator is f_{PSC}. Assuming an oscillator frequency of a multiple of f_{DTMF} = 3.579545 MHz, the division factor of the prescaler should be chosen such that $f_{PSC} = f_{DTMF}$. The DTMF standard frequencies can then be implemented as shown in Table 34. The relationship between telephone keyboard symbols and the frequency register contents are given in Table 35.

Table 34 DTMF standard frequencies and their implementation

LGF/HGF VALUE (HEX)	FREQUENCY (Hz)		DEVIATION	
	STANDARD	GENERATED	(%)	(Hz)
DD	697	697.90	0.13	0.90
C8	770	770.46	0.06	0.46
B5	852	850.45	–0.18	–1.55
A3	941	943.23	0.24	2.23
7F	1209	1206.45	–0.21	–2.55
72	1336	1341.66	0.42	5.66
67	1477	1482.21	0.35	5.21
5D	1633	1638.24	0.32	5.24

Table 35 Dialling symbols, corresponding DTMF frequency pairs and frequency registers content

TELEPHONE KEYBOARD SYMBOLS	DTMF FREQUENCY PAIRS (Hz)	LGF VALUE (HEX)	HGF VALUE (HEX)
0	(941, 1336)	A3	72
1	(697, 1209)	DD	7F
2	(697, 1336)	DD	72
3	(697, 1477)	DD	67
4	(770, 1209)	C8	7F
5	(770, 1336)	C8	72
6	(770, 1477)	C8	67
7	(852, 1209)	B5	7F
8	(852, 1336)	B5	72
9	(852, 1477)	B5	67
A	(697, 1633)	DD	5D
B	(770, 1633)	C8	5D
C	(852, 1633)	B5	5D
D	(941, 1633)	A3	5D
•	(941, 1209)	A3	7F
#	(941, 1477)	A3	67

4.10.6 MODEM FREQUENCIES

Assuming an oscillator frequency of $f_{PSC} = f_{DTMF} = 3.579545$ MHz, the standard modem frequency pairs summarized in Table 36 can be implemented. It is suggested to define the frequency using the HGF register while the LGF register contains 00H, disabling Low Group Frequency generation.

Table 36 Standard modem frequency pairs and their implementation

HGF VALUE (HEX)	FREQUENCY (Hz)		DEVIATION	
	MODEM	GENERATED	(%)	(Hz)
9D	980[1]	978.82	–0.12	–1.18
82	1180[1]	1179.03	–0.08	–0.97
8F	1070[2]	1073.33	0.31	3.33
79	1270[2]	1265.30	–0.37	–4.70
80	1200[3]	1197.17	–0.24	–2.83
45	2200[3]	2192.01	–0.36	–7.99
76	1300[4]	1296.94	–0.24	–3.06
48	2100[4]	2103.14	0.15	3.14
5C	1650[1]	1655.66	0.34	5.66
52	1850[1]	1852.77	0.15	2.77
4B	2025[2]	2021.20	–0.19	–3.80
44	2225[2]	2223.32	–0.08	–1.68

Notes

1. Standard is V.21.
2. Standard is Bell 103.
3. Standard is Bell 202.
4. Standard is V.23.

4.10.7 MUSICAL SCALE FREQUENCIES

Finally, two octaves of musical scale in steps of semitones can be realized, again assuming an oscillator frequency $f_{PSC} = f_{DTMF} = 3.579545$ MHz (Table 37). It is suggested to define the frequency using the HGF register while the LGF register contains 00H, disabling Low group frequency generation.

Table 37 Musical scale frequencies and their implementation

NOTE	HGF VALUE (HEX)	FREQUENCY (Hz)	
		STANDARD(1)	GENERATED
D#5	F8	622.3	622.5
E5	EA	659.3	659.5
F5	DD	698.5	697.9
F#5	D0	740.0	741.1
G5	C5	784.0	782.1
G#5	B9	830.6	832.3
A5	AF	880.0	879.3
A#5	A5	923.3	931.9
B5	9C	987.8	985.0
C6	93	1046.5	1044.5
C#6	8A	1108.7	1111.7
D6	82	1174.7	1179.0
D#6	7B	1244.5	1245.1
E6	74	1318.5	1318.9
F6	6D	1396.9	1402.1
F#6	67	1480.0	1482.2
G6	61	1568.0	1572.0
G#6	5C	1661.2	1655.7
A6	56	1760.0	1768.5
A#6	51	1864.7	1875.1
B6	4D	1975.5	1970.0
C7	48	2093.0	2103.3
C#7	44	2217.5	2223.3
D7	40	2349.3	2358.1
D#7	3D	2489.0	2470.4

Note

1. Standard scale based on A4 at 440 Hz.

4.11 MSK modem

The MSK modem is used for in-band signalling between handset and base in analog cordless telephone systems CT0, CT1 and CT1+. The MSK modem's receiver and transmitter can be enabled separately. Receive and transmit interrupts can wake-up the microcontroller during its power saving Idle mode. Baud rates are programmable between 1200 and 4800 Baud. Fig.20 shows the functional diagram of the MSK modem.

The modem has the following features:

- Full duplex operation via 8-bit parallel interface
- The message is fully Manchester coded/decoded
- Automatic detection of 16-bit Manchester preamble pattern
- The last received 4 bits of the preamble pattern are software programmable
- Receiver full, transmitter empty indication bits
- Manchester coding and decoding for clock recovery and early error detection
- Programmable input polarity
- Baud rate selection from 1200, 2400, 3600 and 4800 baud with internal modem timer
- Receiver and transmitter off-states with no power consumption.

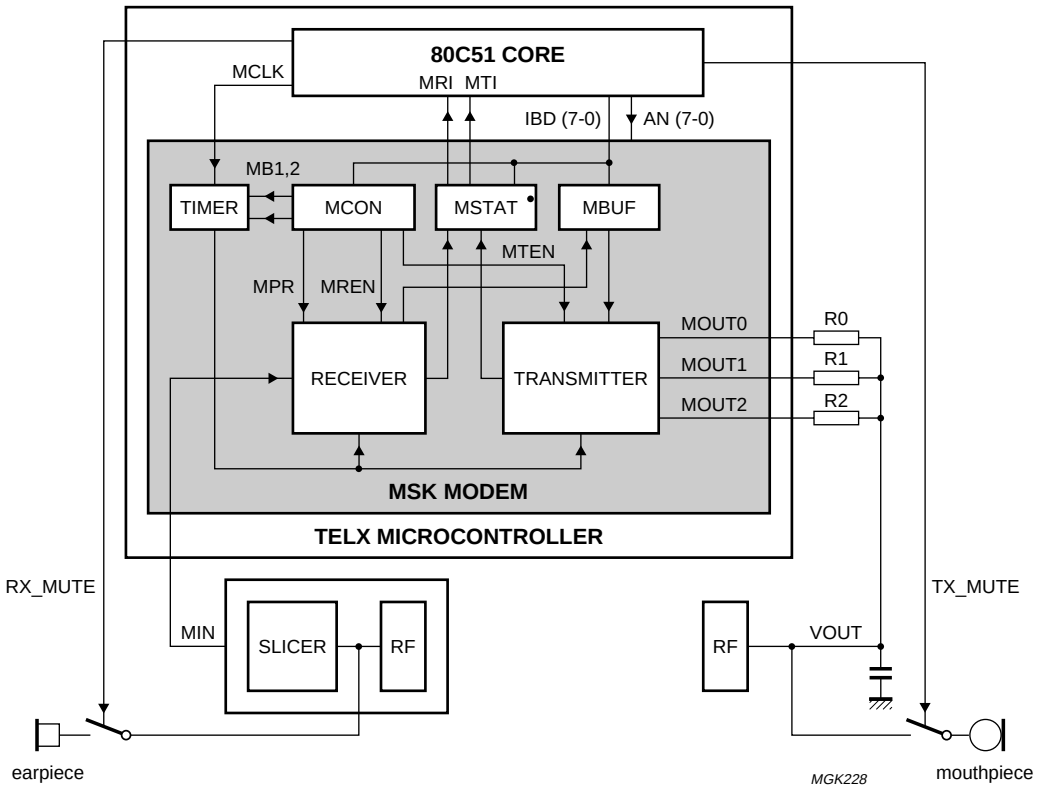


Fig.20 MSK modem functional diagram.

4.11.1 80C51 MICROCONTROLLER INTERFACE

The modem block interfaces to the microcontroller via the interrupt signals MRI and MTI and via the control and data SFRs MCON, MSTAT and MBUF. The MSK modem receive and transmit registers are both accessed via the Special Function Register MBUF. Writing to MBUF loads the transmit register and reading MBUF accesses a physically separate receive register.

4.11.1.1 MSK Modem Control Register (MCON)

Table 38 MSK Modem Control Register (SFR address D3H)

7	6	5	4	3	2	1	0
MPR3	MPR2	MPR1	MPR0	MB1	MB0	MTEN	MREN

Table 39 Description of MCON bits

BIT	SYMBOL	DESCRIPTION
MCON.7	MPR3	**Modem preamble pattern.** These 4 bits define the modem's preamble pattern.
MCON.6	MPR2	
MCON.5	MPR1	
MCON.4	MPR0	
MCON.3	MB1	**Modem transmit/receive frequency.** These 2-bits define the modem's transmit/receive frequency; see Table 40.
MCON.2	MB0	
MCON.1	MTEN	**Modem Transmitter Enable.** If this bit is set the transmitter is active and MOUT<3:1> will get the value <100> if no data is transmitted; if reset, MOUT<3:1> will get the value <111> to zero the currents in the resistive DAC. See note 1.
MCON.0	MREN	**Modem Receiver Enable.** If this bit is set the modem receiver is active and scans for Manchester data. See note 1.

Note

1. If both the transmitter and the receiver are disabled (MTEN = 0 and MREN = 0), the clock of the MSK modem is switched-off. It is advised to use this state for power saving.

Table 40 Selection of the modem's baud rates

MB1	MB0	MODEM BAUD RATE
0	0	1200 Baud
0	1	2400 Baud
1	0	3600 Baud
1	1	4800 Baud

4.11.1.2 MSK Modem Status Register (MSTAT)

Table 41 MSK modem Status Register (SFR address D2H)

7	6	5	4	3	2	1	0
–	–	MRF	MRE	MRP	MRL	MTI	MRI

Table 42 Description of MSTAT bits

BIT	SYMBOL	DESCRIPTION
MSTAT.5	MRF	**Modem Receiver Full flag.** MRF is set when MBUF holds a newly received byte. MRF is reset if the receiver is disabled (MREN = 0) or by reading MBUF. This bit is read-only. Writing to it will have no effect.
MSTAT.4	MRE	**Modem Receiver Error flag.** Indicates the reception of a non-Manchester bit. This bit is set by hardware and is reset by reading MBUF, by disabling the receiver (MREN = 0) or by resetting MRI. This bit is read-only. Writing to it will have no effect.
MSTAT.3	MRP	**Modem Receiver Preamble flag.** MRP is set by hardware when the modem recognizes the programmed preamble pattern (AAAH) after locking the receiver clock (MRL = 1). MRP is reset by hardware if the receiver is disabled (MREN = 0) or if non-Manchester data is received (MRE = 1). This bit is read-only. Writing to it will have no effect.
MSTAT.2	MRL	**Modem Receiver Clock Locked flag.** This bit is set when the clock of the receiver is locked, i.e. when the receiver has detected Manchester data but has not found the preamble pattern yet. MRL is reset when the receiver detects a non-Manchester bit or when the receiver is disabled. This bit is read-only. Writing to it will have no effect.
MSTAT.1	MTI	**Modem Transmit Interrupt flag.** Indicates MBUF is empty and ready to accept a new byte for transmission. MTI is reset by writing to MBUF or by writing a logic 0 to it. Writing a logic 1 to MTI sets the bit and allows a hardware interrupt to be generated by software.
MSTAT.0	MRI	**Modem Receive Interrupt flag.** Indicates Modem Receiver Full (MRF = 1) or Modem Receiver Error (MRE = 1) or Modem Receiver Preamble (MRP = 1) or Modem Receiver Clock Locked (MRL = 1) This bit is reset by reading MBUF or by writing a logic 0 to MRI. A reset of MRI will also reset MRE. Writing a logic 1 to MRI will have no effect.

4.11.1.3 MSK Modem Data Buffer (MBUF)

Table 43 MSK Modem Data Buffer (SFR address D1H)

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Table 44 Description of MBUF bits

BIT	SYMBOL	DESCRIPTION
MBUF.7 to MBUF.0	D7 to D0	Writing to MBUF loads the data into the transmit buffer and starts a transmission at MOUT if the transmitter is enabled (MTEN = 1). A new byte can be loaded after MTI is set. If a new byte is loaded before MTI is set the previous byte will be lost. After data has been received at MIN, indicated by MRI, the received byte can be read from MBUF.

4.11.2 MODEM INTERFACE

The modem block has the following modem interface signals.

- **MIN:** digital MSK Manchester coded input signal from the Data slicer. MIN is the alternative input function of P4.0. If P4.0 is used for MIN, it has to be configured to be an input port. The data that is written to the ports data SFR can be used to switch the polarity of MIN. If P4.0 data is set, the value on the pin is passed to MIN with inverted polarity. If P4.0 data is reset, the value on the pin is passed directly to MIN.
- **MOUT0 to MOUT2:** 3-bit Manchester coded output signal of the modem.

The mute signals RX_MUTE and TX_MUTE must be generated by software. Any standard I/O port pin can be used for this purpose.

4.11.3 DATA TRANSMISSION

Data transmission is enabled if bit MTEN in register MCON is set to a logic 1. If MTEN is a logic 0, data transmission is disabled and MOUT<2:0> is set to <111> to zero the currents in the resistive DAC. Setting MTEN to a logic 1 sets MOUT<2:0> to the idle value <100>. This results in a value close to $\frac{1}{2} \times V_{DD}$ on the output signal of the external DAC. Transmission is started by loading the first byte into register MBUF. All bytes are transmitted starting with the MSB.

A message is transferred in a block of 3 or more bytes, the first two bytes being the programmed Manchester preamble pattern. In order to insert the preamble pattern, the first two bytes AAH and AXH (with X being the MPR3 to MPR0 value programmed in the receiver MSK modem) have to be written to MBUF by software. After this, the first byte of the message is written to MBUF. As soon as MBUF is ready to accept new input, signal MTI is set. A new byte written to MBUF automatically clears MTI. The time between two MTI interrupts is $T = 8 \times 1$/ baud rate (e.g. for baud rate 1200 baud, T = 6.7 ms). If no new byte is written to MBUF at the end of a byte transmission, the modem transmitter stops transmission and MOUT<2:0> is set to the idle state <100>.

In this case MTI must be cleared explicitly. If MTEN is reset during transmission, the transmitter will finish the transmission of the current byte and then will set MOUT<2:0> to the off state <111>. No interrupt on MTI will be generated at the end of the transmission.

4.11.4 DATA RECEPTION

A message is received as a block of one or more data bytes. When enabled, the receiver starts sampling MIN and tries to detect a Manchester pattern. As soon as 3 consecutive Manchester bits are detected the receiver clock is locked (MRL = 1) and the receiver starts scanning the incoming data for the programmed Manchester preamble pattern. When the modem recognizes the preamble pattern, bit MRP is set to a logic 1. If a non-Manchester bit is detected before finding the preamble pattern then MRL is reset and MRE is set to a logic 1. The synchronisation process has to restart. If the preamble pattern has been detected the receiver starts to Manchester decode the incoming data bits and shifts them into an internal register. After eight bits the contents of the internal register are copied to MBUF and MRF bit is set to a logic 1. The received byte can be read from MBUF while receiving continues in the internal register. If a non-Manchester bit is received during data reception then MRE is set to a logic 1 and MRL and MRP are reset. The receiver has to resynchronize before receiving new data.

Whenever one of the bits MRF, MRE, MRP and MRL is set the MRI bit is also set and an RTI interrupt is generated. This means that when an RTI interrupt occurs the 4 status bits have to be polled by software. The bit MRL allows the software to decide very quickly whether an occupied channel contains Manchester coded data or not. The MRP bit is used to find the start of data transmission in a message that is repeated over and over again. MRE is used to detect a Manchester error, which is a violation of the Manchester coding rule that the received level should change in the middle of a bitcell. The MRF bit indicates that the data in MBUF is ready to be read by the software. During data reception the time between two settings of MRF (each one generating an MRI interrupt) is $T = 8 \times 1$/baud rate. Figure 22 shows an example of the timing diagram of data reception.

4.11.5 MANCHESTER CODING OF DATA

The bits of the data byte written in MBUF are Manchester encoded as shown in Fig.22: A ‘1’ is coded as a LOW-to-HIGH transition in the middle of a bitcell, a ‘0’ is coded as a HIGH-to-LOW transition.The Manchester encoded signal contains redundancy for early error detection in received bits. A non-matching HIGH-to-LOW or LOW-to-HIGH pair indicates an error condition.The Manchester encoded signal has a polarity change in each bitcell.

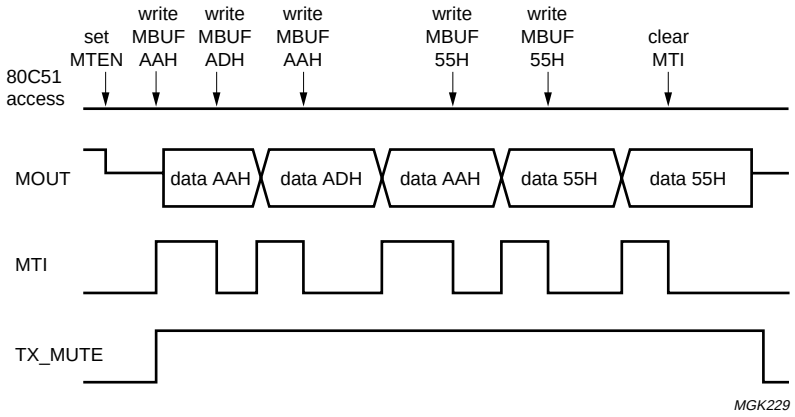


Fig.21 Data transmission timing diagram.

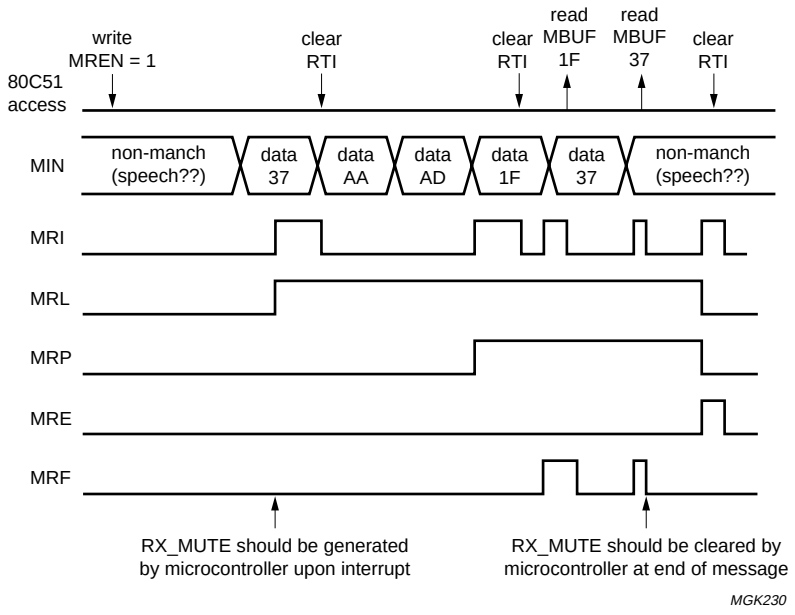


Fig.22 Data reception timing diagram.

4.11.6 WAVEFORM GENERATION WITH MOUT<2:0>

The 3 digital output pins MOUT0 to MOUT2, should be used as an input to a three bit external DAC. The signals can be connected via external resistors R0, R1 and R2 to a summation point and then be filtered with an external capacitor (C1). The 3-bit DAC is shown in Fig.23. Table 45 gives the relationship between the MOUT pins, the resistor values and VOUT.

Figure 24 shows the waveforms that are produced by the waveform generator. The horizontal axis shows the sample counter on which the waveform changes its value. Each bit is built-up out of 2×124 samples. The vertical axis shows the values of MOUT<2:0>, forming the inputs of the resistive DAC. The first half of the waveform is determined by the previous and the current bit, whereas the second half of the waveform is determined by the current and the next bit to be transmitted. The count frequency of the sample counter depends on the programmed baud rate.

If the transmitter is disabled with MTEN set to a logic 0, MOUT<2:0> is <111> to save power in the resistive DAC. If the transmitter is enabled and no data is transmitted, MOUT<2:0> has an idle value of <100>, which corresponds to $0.57 \times V_{DD}$.

4.11.7 SYNCHRONISATION

When enabled the receiver samples MIN with a frequency $f = 8 \times$ baud rate. The sampled values are shifted into an 8-bit shift register. This register is regularly checked to determine whether it contains samples that fulfil the Manchester coding rule i.e. whether there is a LOW-to-HIGH or a HIGH-to-LOW transition in the middle of the bitcell. The receiver searches for 3 consecutive sets of 8 samples that fulfil the Manchester coding rule. If these sets have been found the clock is locked (MRL = 1) and the receiver starts looking for the Manchester preamble pattern. From this point on the receiver uses a PLL (Phase Locked Loop) to adjust the synchronisation after each received Manchester bit.

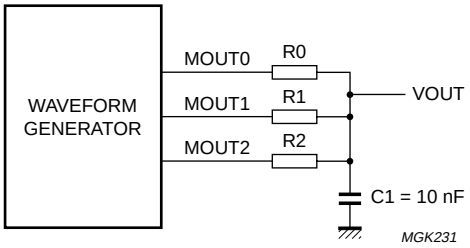


Fig.23 3 bit DAC with MOUT<2:0>.

Table 45 V_{out} as a function of MOUT<2:0> and the resistor values

MOUT2	MOUT1	MOUT0	VOUT	RESISTOR VALUES
0	0	0	0	R0 = R R1 = 0.48 × R R2 = 0.25 × R
0	0	1	$0.14 \times V_{DD}$	
0	1	0	$0.29 \times V_{DD}$	
0	1	1	$0.43 \times V_{DD}$	
1	0	0	$0.57 \times V_{DD}$	
1	0	1	$0.71 \times V_{DD}$	
1	1	0	$0.86 \times V_{DD}$	
1	1	1	V_{DD}	

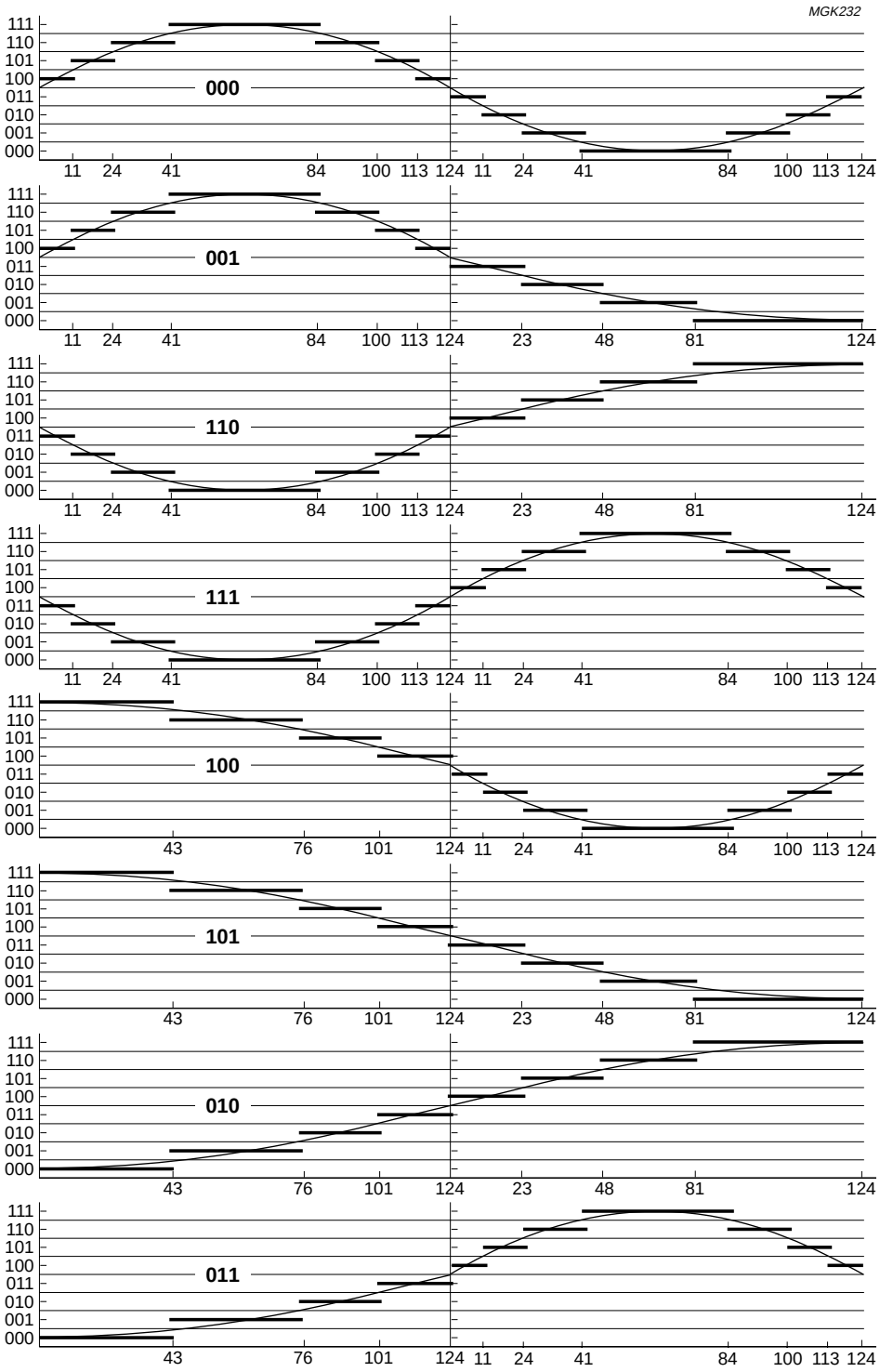


Fig.24 Waveforms with MOUT<2:0> for previous, current and next bits to be transmitted.

4.12 I²C-bus serial I/O

The serial port supports the twin line I²C-bus. The I²C-bus consists of two lines: a data line (SDA) and a clock line (SCL). These lines also function as the I/O port lines P1.7 and P1.6 respectively. The system is unique because data transport, clock generation, address recognition and bus control arbitration are all controlled by hardware.

The I²C-bus serial I/O has complete autonomy in byte handling and operates in 4 modes:

- Master transmitter
- Master receiver
- Slave transmitter
- Slave receiver.

These functions are controlled by the S1CON register. S1STA is the Status Register whose contents may also be used as a vector to various service routines. S1DAT is the Data Shift Register and S1ADR the Slave Address Register. Slave address recognition is performed by on-chip hardware. The block diagram of the I²C-bus serial I/O is shown in Fig.25.

4.12.1 I²C-bus internal mode

A special internal mode is provided. In this mode other on-chip blocks with an I²C-bus interface can communicate without using the I/O port lines P1.7 and P1.6, thus freeing them for other purposes. The microcontroller can be configured to use this internal mode or the normal external mode, with the port configuration bits; see Section 4.7.2.

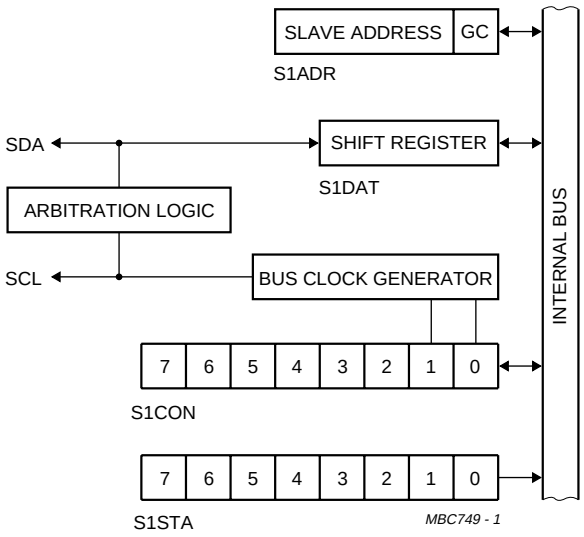


Fig.25 Block diagram of I²C-bus serial I/O.

4.12.2 SERIAL CONTROL REGISTER (S1CON)

Table 46 Serial Control Register (SFR address D8H)

7	6	5	4	3	2	1	0
CR2	ENS1	STA	STO	SI	AA	CR1	CR0

Table 47 Description of S1CON bits

BIT	SYMBOL	DESCRIPTION
S1CON.7	CR2	This bit along with bits CR1 and CR0 determines the serial clock frequency when SIO is in the Master mode. See Table 48. When CR2 = 0 the I^2C-bus is in fast mode.
S1CON.6	ENS1	**ENABLE serial I/O.** When ENS1 = 0, the serial I/O is disabled. SDA and SCL outputs are in the high-impedance state; P1.6 and P1.7 function as open-drain ports. When ENS1 = 1, the serial I/O is enabled. Output port latches P1.6 and P1.7 must be set to logic 1.
S1CON.5	STA	**START flag.** When this bit is set in Slave mode, the SIO hardware checks the status of the I^2C-bus and generates a START condition if the bus is free or after the bus becomes free. If STA is set while the SIO is in Master mode, SIO will generate a repeated START condition.
S1CON.4	STO	**STOP flag.** With this bit set while in Master mode a STOP condition is generated. When a STOP condition is detected on the I^2C-bus, the SIO hardware clears the STO flag. STO may also be set in Slave mode in order to recover from an error condition. In this case no STOP condition is transmitted to the I^2C-bus. However, the SIO hardware behaves as if a STOP condition has been received and releases the SDA and SCL. The SIO then switches to the not addressed slave receiver mode. The STOP flag is cleared by the hardware.
S1CON.3	SI	**SIO interrupt flag.** This flag is set, and an interrupt is generated, after any of the following events occur: • A start condition is generated in Master mode • Own slave address has been received during AA = 1 • The general call address has been received while S1ADR0 and AA = 1 • A data byte has been received or transmitted in Master mode (even if arbitration is lost) • A data byte has been received or transmitted as selected slave • A Stop or Start condition is received as selected slave receiver or transmitter. If this flag is set, the I^2C-bus is halted (by pulling down SCL). Received data is only valid until this flag is reset.
S1CON.2	AA	**Assert Acknowledge.** When this bit is set, an acknowledge (LOW level to SDA) is returned during the acknowledge clock pulse on the SCL line when: • Own slave address is received • General call address is received (S1ADR.0 = 1) • A data byte is received while the device is programmed to be a Master Receiver • A data byte is received while the device is a selected Slave Receiver. When this bit is reset, no acknowledge is returned. Consequently, no interrupt is requested when the own slave address or general call address is received.
S1CON.1	CR1	These two bits along with the CR2 bit determine the serial clock frequency when SIO is in the Master mode. See Table 48.
S1CON.0	CR0	

Table 48 Selection of the serial clock frequency in the Master mode of operation; see notes 1 and 2

CR2	CR1	CR0	f_{osc} DIVISOR	BIT RATE (kHz) at f_{PSC}		
				3.58 MHz	4 MHz	6 MHz
0	0	0	10	358	400	(600)
0	0	1	20	179	200	300
0	1	0	30	119.33	133	199.5
0	1	1	40	89.5	100	150
1	0	0	80	44.75	50	75
1	0	1	120	29.83	33	49.5
1	1	0	160	22.38	25	37.5
1	1	1	(256 – T1 reload value) × 12 24 to 3072	1.17 to 149	1.3 to 167	100

Notes

1. Bit rates greater than 400 kHz are outside the specified frequency range.
2. When the CR (2:0) = 111, the maximum bit rate for the data transfer will be derived from the Timer 1 overflow rate divided by 2, i.e. every time the Timer 1 overflows the SCL signal will toggle.

4.12.3 Data Shift Register (S1DAT)

S1DAT contains the serial data to be transmitted or data which has just been received. Bit 7 is transmitted or received first; i.e. data shifted from left to right.

Table 49 Data Shift Register (SFR address DAH)

7	6	5	4	3	2	1	0
S1DAT.7	S1DAT.6	S1DAT.5	S1DAT.4	S1DAT.3	S1DAT.2	S1DAT.1	S1DAT.0

4.12.4 Address Register (S1ADR)

This 8-bit register may be loaded with the 7-bit slave address to which the controller will respond when programmed as a slave receiver/transmitter.

Table 50 Address Register (SFR address DBH)

7	6	5	4	3	2	1	0
SLA6	SLA5	SLA4	SLA3	SLA2	SLA1	SLA0	GC

Table 51 Description of S1ADR bits

BIT	SYMBOL	DESCRIPTION
S1ADR.7 to S1ADR.1	SLA6 to 0	own slave address
S1ADR.0	GC	this bit is used to determine whether the general CALL address is recognized; when a logic 0, the general CALL address is not recognized; when a logic 1, the general CALL address is recognized

4.12.5 SERIAL STATUS REGISTER (S1STA)

The contents of this register may be used as a vector to a service routine. This optimizes the response time of the software and consequently that of the I^2C-bus. S1STA is a read-only register. The status codes for all possible modes of the I^2C-bus interface are given in Tables 54 to 58.

Table 52 Serial Status Register (SFR address D9H)

7	6	5	4	3	2	1	0
SC4	SC3	SC2	SC1	SC0	0	0	0

Table 53 Description of S1STA bits

BIT	SYMBOL	DESCRIPTION
S1STA.3 to S1STA.7	SC4 to SC0	5-bit status code
S1STA.0 to S1STA.2	–	these three bits are held LOW

Table 54 MST/TRX mode

S1STA VALUE	DESCRIPTION
08H	a START condition has been transmitted
10H	a repeated START condition has been transmitted
18H	SLA and W have been transmitted, ACK has been received
20H	SLA and W have been transmitted, $\overline{\text{ACK}}$ received
28H	DATA of S1DAT has been transmitted, ACK received
30H	DATA of S1DAT has been transmitted, $\overline{\text{ACK}}$ received
38H	arbitration lost in SLA, R/W or DATA

Table 55 MST/REC mode

S1STA VALUE	DESCRIPTION
38H	arbitration lost while returning $\overline{\text{ACK}}$
40H	SLA and R have been transmitted, ACK received
48H	SLA and R have been transmitted, $\overline{\text{ACK}}$ received
50H	DATA has been received, ACK returned
58H	DATA has been received, $\overline{\text{ACK}}$ returned

Table 56 SLV/REC mode

S1STA VALUE	DESCRIPTION
60H	own SLA and W have been received, ACK returned
68H	arbitration lost in SLA, R/W as MST; own SLA and W have been received, ACK returned
70H	general CALL has been received, ACK returned
78H	arbitration lost in SLA, R/W as MST; general CALL has been received
80H	previously addressed with own SLA; DATA byte received, ACK returned
88H	previously addressed with own SLA; DATA byte received, $\overline{\text{ACK}}$ returned
90H	previously addressed with general CALL; DATA byte has been received, ACK has been returned
98H	previously addressed with general CALL; DATA byte has been received, $\overline{\text{ACK}}$ has been returned
A0H	a STOP condition or repeated START condition has been received while still addressed as SLV/REC or SLV/TRX

Table 57 SLV/TRX mode

S1STA VALUE	DESCRIPTION
A8H	own SLA and R have been received, ACK returned
B0H	arbitration lost in SLA, R/W as MST. Own SLA and R have been received, ACK returned
B8H	DATA byte has been transmitted, ACK received
C0H	DATA byte has been transmitted, $\overline{\text{ACK}}$ received
C8H	last DATA byte has been transmitted (AA = logic 0), ACK received

Table 58 Miscellaneous

S1STA VALUE	DESCRIPTION
00H	bus error during MST mode or selected SLV mode, due to an erroneous START or STOP condition
F8H	no information available (reset value). The serial interrupt flag SI, is not yet set

Table 59 Symbols used in Tables 54 to 58

SYMBOL	DESCRIPTION
SLA	7-bit slave address
R	read bit
W	write bit
ACK	acknowledgement (acknowledge bit = logic 0)
$\overline{\text{ACK}}$	no acknowledgement (acknowledge bit = logic 1)
DATA	8-bit data byte to or from I^2C-bus
MST	master
SLV	slave
TRX	transmitter
REC	receiver

4.13 Standard serial interface SIO0: UART

This serial port is full duplex which means that it can transmit and receive simultaneously. It is also receive-buffered and can commence reception of a second byte before a previously received byte has been read from the register. (However, if the first byte has not been read by the time the reception of the second byte is complete, the second bytes will be lost). The serial port receive and transmit registers are both accessed via the Special Function Register S0BUF. Writing to S0BUF loads the transmit register and reading S0BUF accesses a physically separate receive register.

The serial port can operate in 4 modes:

Mode 0 Serial data enters and exits through RXD. TXD outputs the shift clock. 8 bits are transmitted/received (LSB first). The baud rate is fixed at $\frac{1}{6}$ the oscillator frequency.

Mode 1 10 bits are transmitted (through TXD) or received (through RXD): a start bit (0), 8 data bits (LSB first) and a stop bit (1). On receive, the stop bit goes into RB8 in Special Function Register S0CON. The baud rate is variable.

Mode 2 11 bits are transmitted (through TXD) or received (through RXD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit and a stop bit (1). On transmit, the 9th data bit (TB8 in S0CON) can be assigned the value of 0 or 1. Or, for example, the parity bit (P, in the PSW) could be moved into TB8. On receive, the 9th data bit goes into RB8 in S0CON, while the stop bit is ignored. The baud rate is programmable to either $\frac{1}{32}$ or $\frac{1}{64}$ of the oscillator frequency.

Mode 3 11 bits are transmitted (through TXD) or received (through RXD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit and a stop bit (1). In fact, Mode 3 is the same as Mode 2 in all respects except baud rate. The baud rate in Mode 3 is variable.

In all four modes, transmission is initiated by any instruction that uses S0BUF as a destination register. Reception is initiated in Mode 0 by the condition RI = 0 and REN = 1. Reception is initiated in the other modes by the incoming start bit if REN = 1.

4.13.1 Multiprocessor communications

Modes 2 and 3 have a special provision for multiprocessor communications. In these modes, 9 data bits are received. The 9th bit goes into RB8. The following bit is the stop bit. The port can be programmed such that when the stop bit is received, the serial port interrupt will be activated, but only if RB8 = 1. This feature is enabled by setting bit SM2 in S0CON. One use of this feature, in multiprocessor systems, is as follows.

When the master processor wants to transmit a block of data to one of several slaves, it first sends out an address byte which identifies the target slave. An address byte differs from a data byte in that the 9th bit is HIGH in an address byte and LOW in a data byte. With SM2 = 1, no slave will be interrupted by a data byte. An address byte, however, will interrupt all slaves, so that each slave can examine the received byte and see if it is being addressed. The addressed slave will clear its SM2 bit and prepare to receive the data bytes that will be sent. The slaves that were not being addressed leave their SM2 bits set and go on about their business, ignoring the coming data bytes.

SM2 has no effect in Mode 0, and in Mode 1 can be used to check the validity of the stop bit. In a Mode 1 reception, if SM2 = 1, the receive interrupt will not be activated unless a valid stop bit is received.

4.13.2 Serial Port Control Register (S0CON)

The Serial Port Control and Status Register is the Special Function Register S0CON; shown in Table 60. The register contains not only the mode selection bits, but also the 9th data bit for transmit and receive (TB8 and RB8), and the serial port interrupt bits (TI and RI).

Table 60 Serial Port Control Register (SFR address 98H)

7	6	5	4	3	2	1	0
SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Table 61 Description of S0CON bits

BIT	SYMBOL	DESCRIPTION
S0CON.7	SM0	These 2 bits are used to select the serial port mode; see Table 62
S0CON.6	SM1	
S0CON.5	SM2	Enables the multiprocessor communication feature in Modes 2 and 3. In these modes, if SM2 = 1, then RI will not be activated if the received 9th data bit (RB8) is a logic 0. In Mode 1, if SM2 = 1, then RI will not be activated unless a valid Stop bit was received. In Mode 0, SM2 should be a logic 0.
S0CON.4	REN	enables serial reception and is set by software to enable reception, and cleared by software to disable reception
S0CON.3	TB8	is the 9th data bit that will be transmitted in Modes 2 and 3. Set or cleared by software as desired
S0CON.2	RB8	in Modes 2 and 3, is the 9th data bit received. In Mode 1, if SM2 = 0 then RB8 is the stop bit that was received; in Mode 0, RB8 is not used
S0CON.1	TI	The transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit time in the other modes, in any serial transmission. Must be cleared by software.
S0CON.0	RI	The receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial transmission (for exception see SM2). Must be cleared by software.

Table 62 Selection of the serial port modes

SM0	SM1	MODE	DESCRIPTION	BAUD RATE
0	0	0	shift register	$\frac{1}{6} \times f_{osc}$
0	1	1	8-bit UART	variable
1	0	2	9-bit UART	$\frac{1}{64} \times f_{osc}$ or $\frac{1}{32} \times f_{osc}$
1	1	3	9-bit UART	variable

4.13.3 BAUD RATES

In Mode 0 the baud rate is fixed as shown in Eqtn.(1):

$$\text{Baud rate} = \frac{f_{osc}}{6} \quad (1)$$

In Mode 2 the baud rate depends on the value of the SMOD bit in the PCON register and is calculated as shown in Eqtn.(2).

$$\text{Baud rate} = \frac{2^{SMOD}}{64} \times f_{osc} \quad (2)$$

For Modes 1 and 3 baud rates see Section 4.13.3.1.

4.13.3.1 Using the special purpose baud rate Timer to generate baud rates

In Modes 1 and 3 the baud rate is determined by the overflow rate of the special purpose baud rate timer and the value of the SMOD bit, as shown in Eqtn. (3):

$$\text{Baud rate} = \frac{2^{SMOD}}{32} \times \text{baud rate timer overflow rate} \quad (3)$$

The baud rate timer overflow rate is controlled by the value of the bits PRESC.7 to PRESC.3 in the PRESC register according to Eqtn.(4)

$$\text{Timer overflow rate} = f_{PSC} \times \frac{1}{2^{PTW0} \times 3^{P3}} \quad (4)$$

This gives the following formula for the baud rate:

$$\text{Baud rate} = \frac{2^{SMOD}}{32} \times f_{PSC} \times \frac{1}{2^{PTWO} \times 3^{P3}} \quad (5)$$

PTWO (PRESC.6 to PRESC.4) defines a power of two in the division factor of the baud rate timer. P3 (PRESC.3) defines a factor of 3 or 1 in the division factor of the baud rate timer. f_{PSC} is the frequency defined by the prescaler (see Section 4.3) and is the DTMF frequency of 3.579545 MHz in typical telecom applications.
The prescaler is controlled with the PS0 to PS2 bits in the PRESC register. Table 63 lists various commonly used baud rates and how they can be obtained with the special purpose baud rate timer and the PRESC register. For detailed description on PRESC see Section 4.3.1.

Table 63 Baud rate timer generated commonly used baud rates, based on f_{PSC} = 3.579545 MHz

BAUD RATE (kbits/s)		SMOD	PRESC.6 TO PRESC.3	DIVISION FACTOR
TARGET	ACTUAL[1]			
115.2[2]	111.8608	1	0010	$2 \times 16 = 32$
57.6	55.9304	0	0010	$2 \times 32 = 64$
38.4	37.2869	0	0001	$3 \times 32 = 96$
28.8	27.9652	0	0100	$4 \times 32 = 128$
19.2	18.6435	0	0011	$6 \times 32 = 192$
14.4	13.9826	0	0110	$8 \times 32 = 256$
9.6	9.3217	0	0101	$12 \times 32 = 384$
7.2	6.9913	0	1000	$16 \times 32 = 512$
4.8	4.6609	0	0111	$24 \times 32 = 768$
3.6	3.4956	0	1010[3]	$32 \times 32 = 1024$
2.4	2.3304	0	1001	$48 \times 32 = 1536$
1.2[4]	1.1652	0	1011[5]	$96 \times 32 = 3072$

Notes

1. Error compared to the target values is less then 3%.
2. Maximum baud rate.
3. And also 1100 and 1110.
4. Minimum baud rate.
5. And also 1101 and 1111.

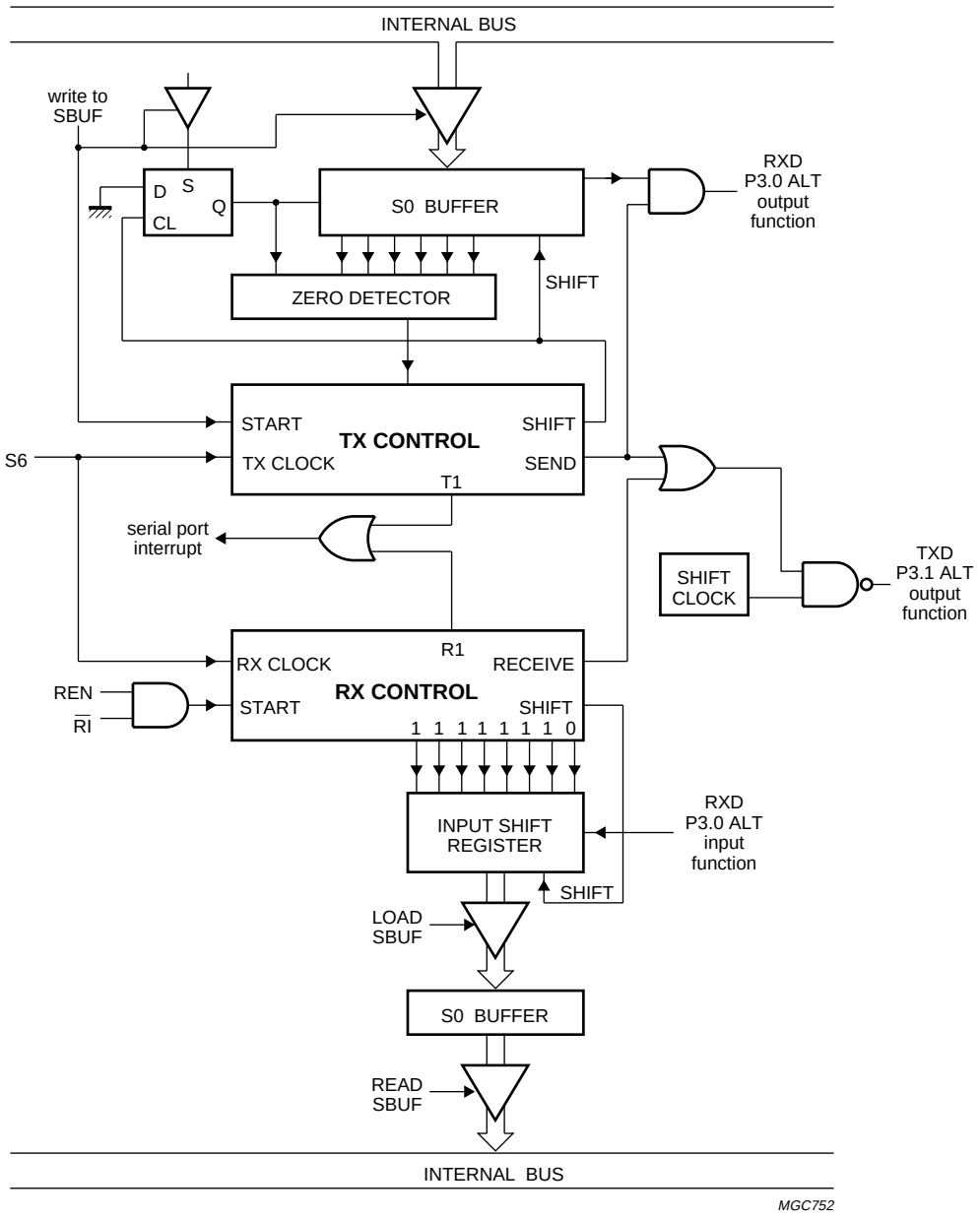


Fig.26 Serial port Mode 0.

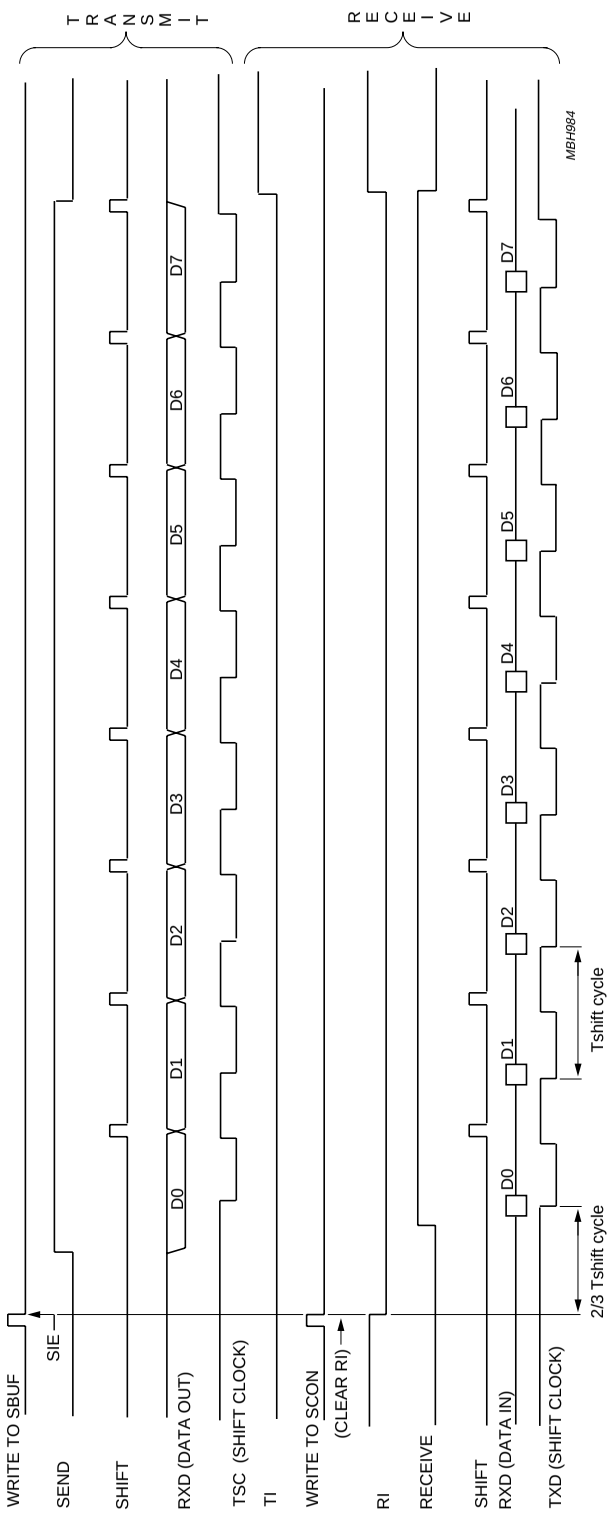


Fig.27 Serial port Mode 0 timing.

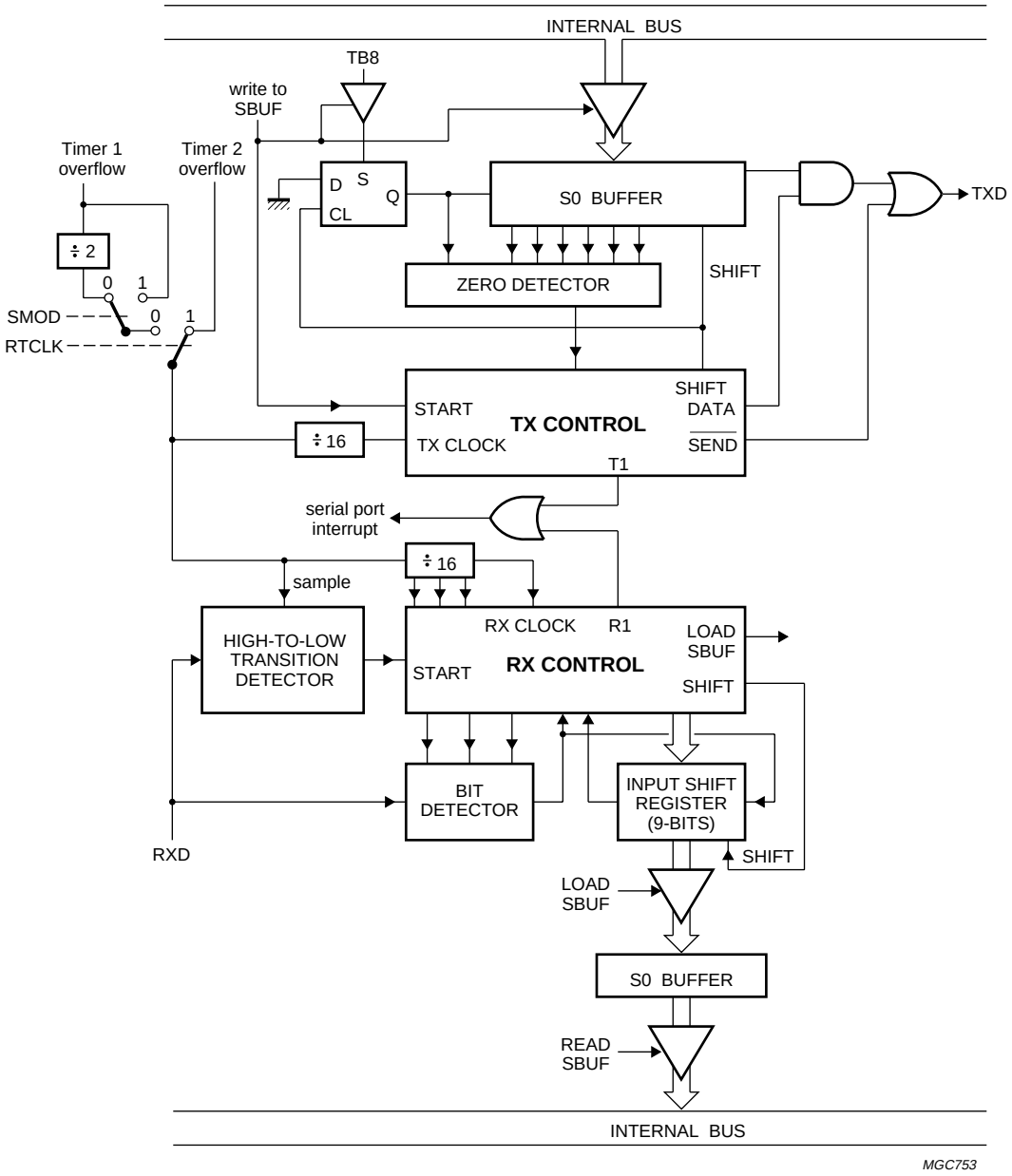


Fig.28 Serial port Mode 1.

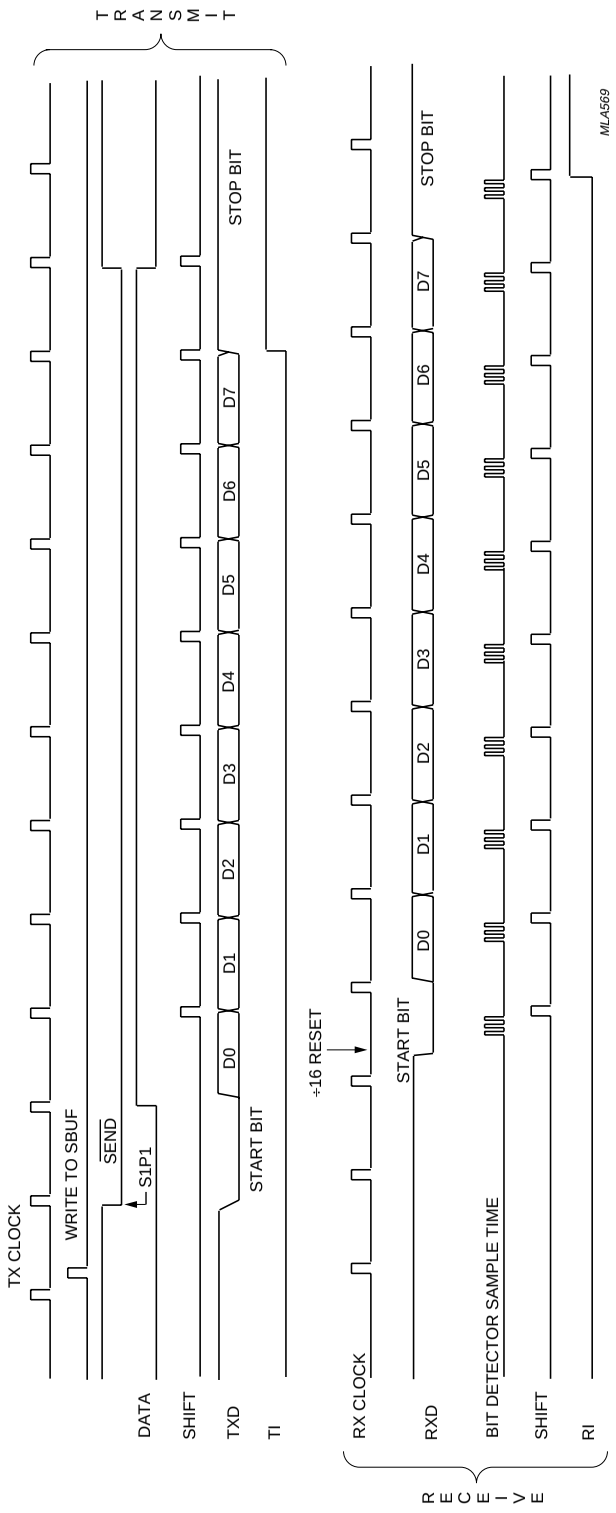


Fig.29 Serial port Mode 1 timing.

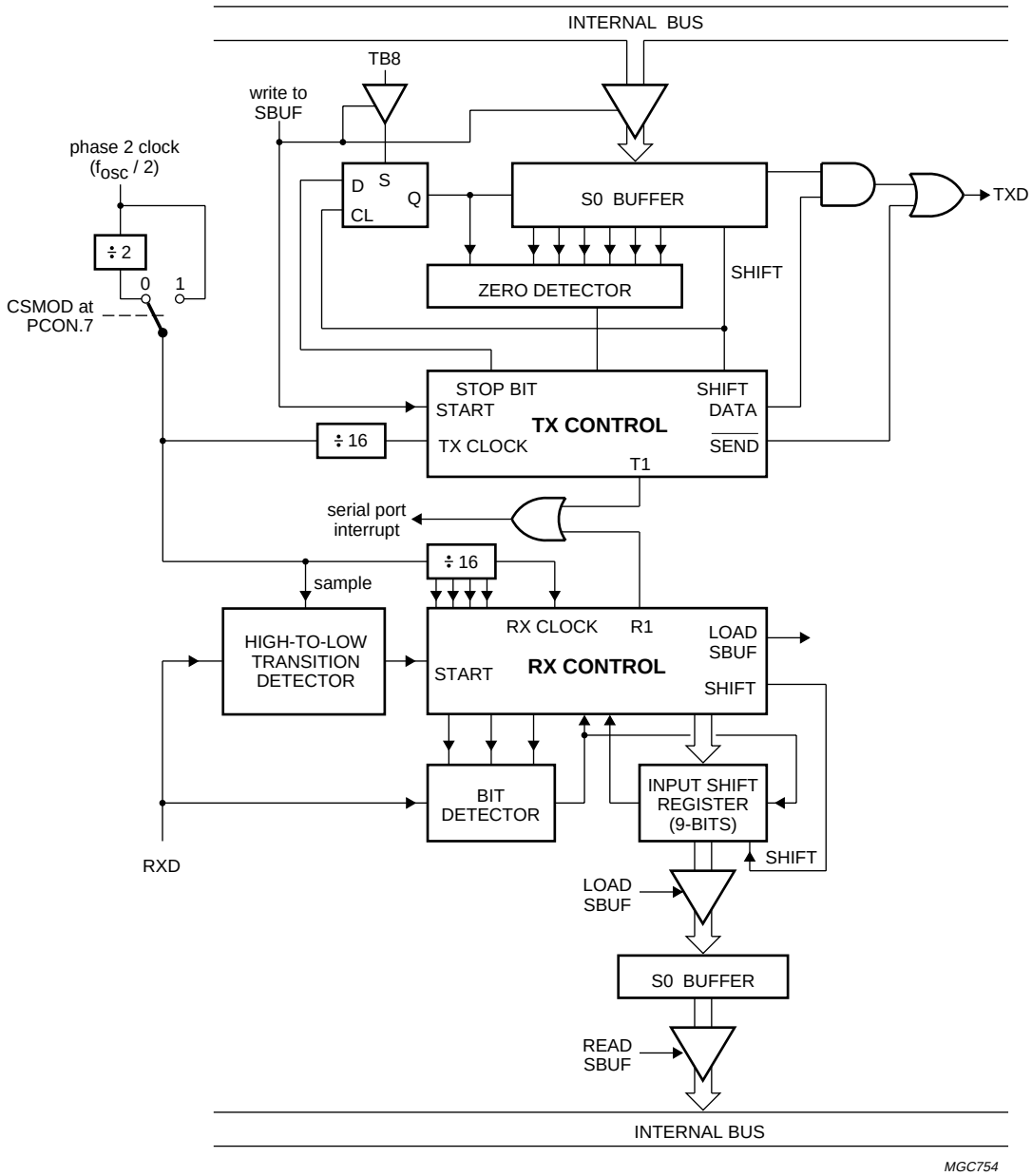


Fig.30 Serial port Mode 2.

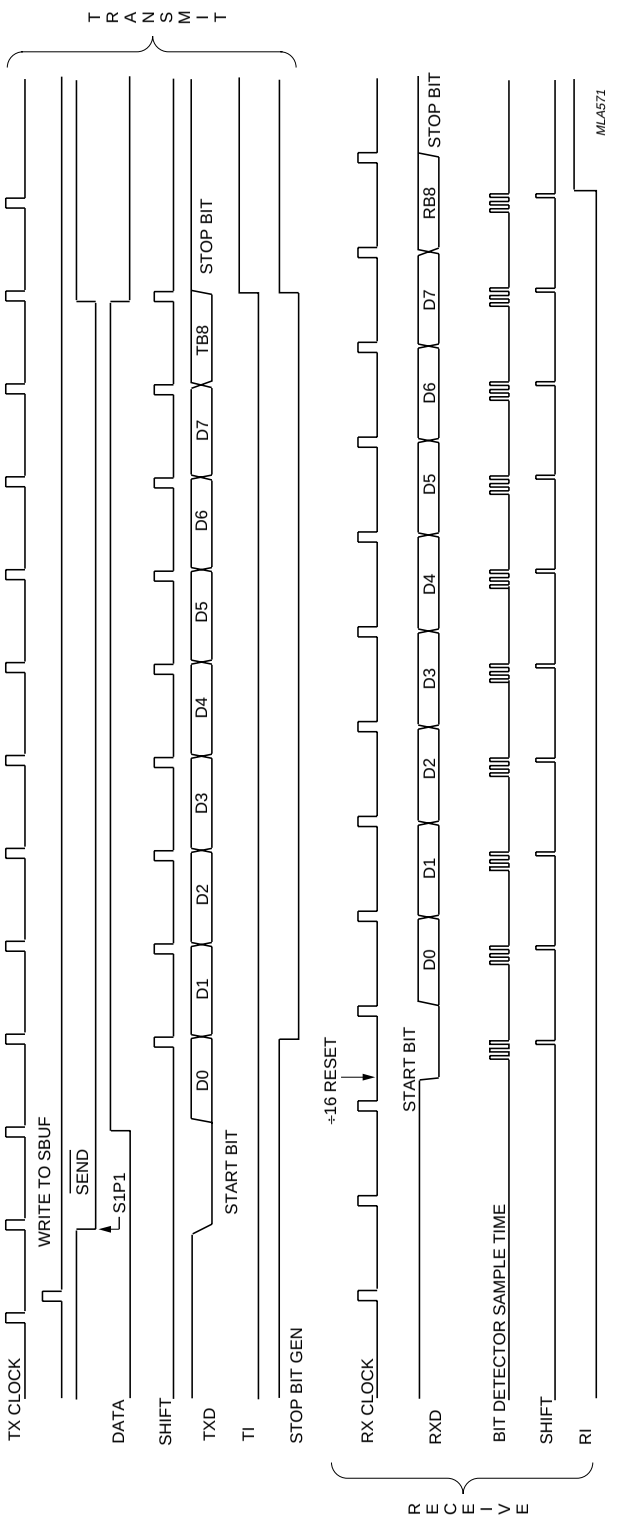


Fig.31 Serial port Mode 2 timing.

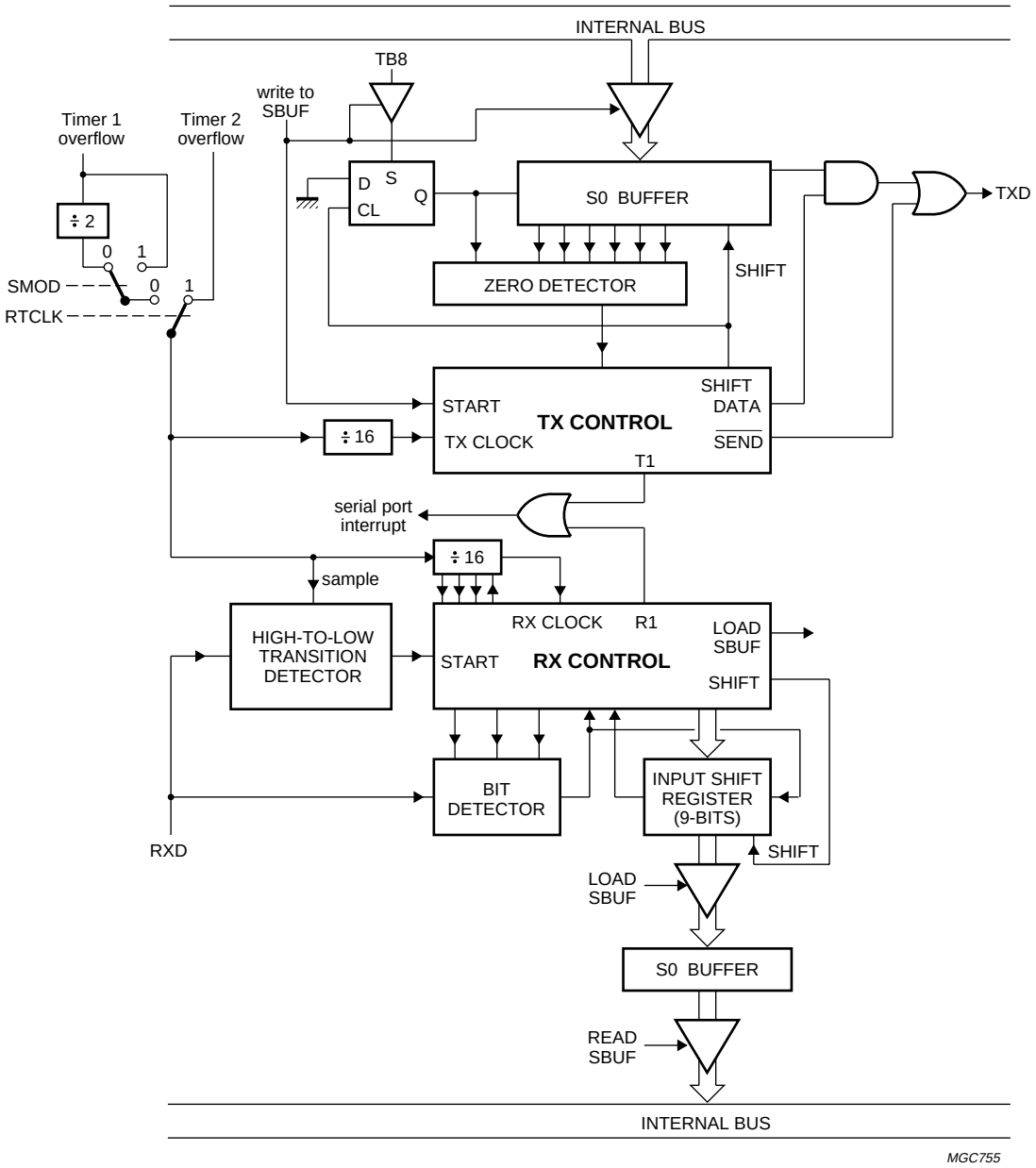


Fig.32 Serial port Mode 3.

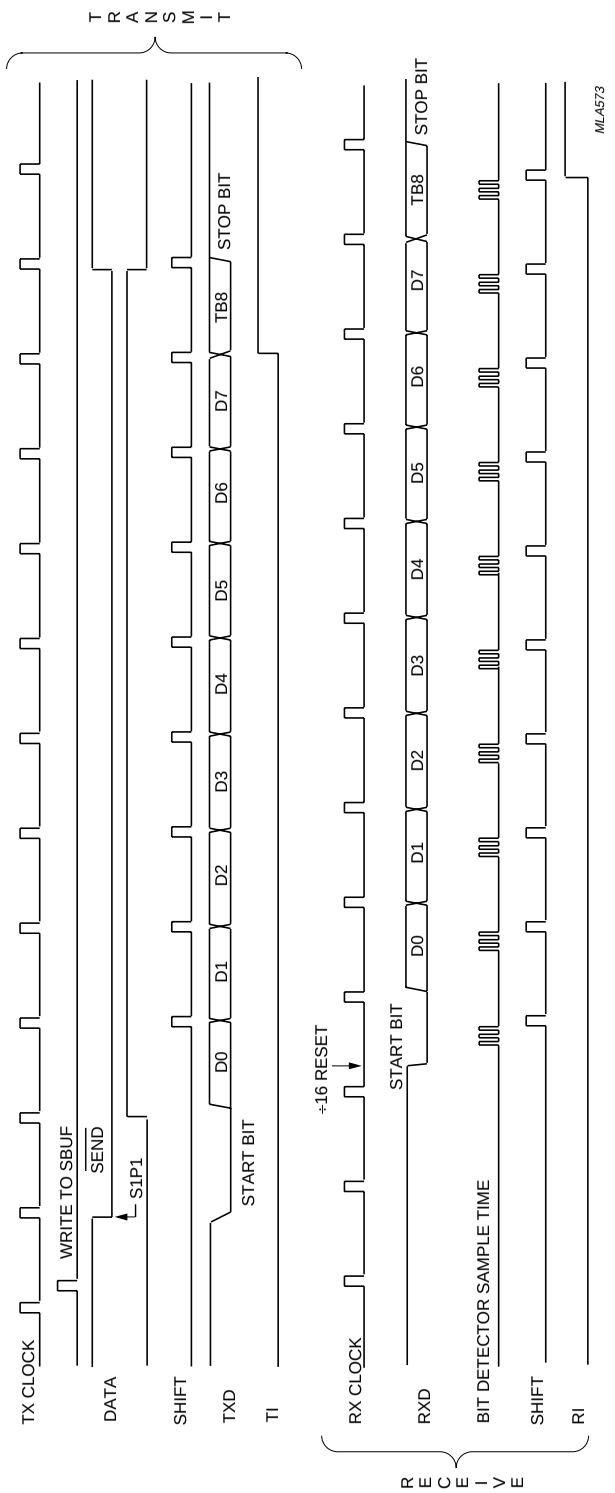


Fig.33 Serial port Mode 3 timing.

4.14 Interrupt system

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronously to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. The TELX family acknowledges interrupt requests from twenty sources:

- INT0 to INT9
- Timer 0, Timer 1 and Timer 2
- I^2C-bus serial I/O
- UART transmitter and receiver
- MSK modem transmitter and receiver
- Low Voltage Detector
- 32 kHz Real-Time Clock.

Each interrupt vectors to a separate location in program memory for its service routine. Each source can be individually enabled or disabled by its corresponding bit in the Interrupt Enable Registers (IEN0 to IEN2). The priority level is selected via the Interrupt Priority Registers (IP0 to IP2). All enabled sources can be globally disabled or enabled. The interrupt system is shown in Fig.34.

4.14.1 External interrupts INT2 to INT9

Port 1 lines serve an alternative purpose as eight additional interrupts: INT2 to INT9. When enabled, each of these lines may wake-up the device from the Power-down mode.

Using the Interrupt Polarity Register (IX1) and the Interrupt Sensitivity Register (ISE1), each pin may be initialized to be either active HIGH, active LOW (i.e. level sensitive), or triggered on a rising or falling edge. A Port 1 level sensitive interrupt will be recognized when a level (HIGH or LOW depending on the Interrupt Polarity Register) on P1.n (n = 0 to 7) is held active for at least one machine cycle. The interrupt request is not serviced until the next machine cycle. The external interrupt configuration is shown in Fig.35.

IRQ1 is the Interrupt Request Flag Register. If the interrupt is enabled, each flag will be set on an interrupt request but must be cleared by software, i.e. via the interrupt software or when the interrupt is disabled by its corresponding bit in IE1. A global interrupt disable will disable the servicing of the interrupts however it does not reset an active interrupt request neither does it stop the detection of an interrupt condition.

4.14.2 Interrupt priority

Each interrupt source can be set to either a high priority or to a low priority. If both priorities are requested simultaneously, the processor will branch to the high priority vector. A low priority interrupt can only be interrupted by a high priority interrupt. A high priority interrupt routine can not be interrupted.

Table 64 shows the interrupt vectors in order of priority. X0 having the highest priority; RTC the lowest. The vector indicates the ROM location where the appropriate interrupt service routine starts.

Table 64 Interrupt vectors

SOURCE	SYMBOL	VECTOR (HEX)
External 0	X0	0003
I^2C-bus port	S1	002B
External 5	X5	0053
MSK modem receiver	MRI	008B
Timer 0	T0	000B
Timer 2	T2	0033
External 6	X6	005B
MSK modem transmitter	MTI	0083
External 1	X1	0013
External 2	X2	003B
External 7	X7	0063
UART transmitter	SOT	007B
Timer 1	T1	001B
External 3	X3	0043
External 8	X8	006B
Low Voltage Detector	LVD	0093
UART receiver	SOR	0023
External 4	X4	004B
External 9	X9	0073
Real-Time Clock	RTC	009B

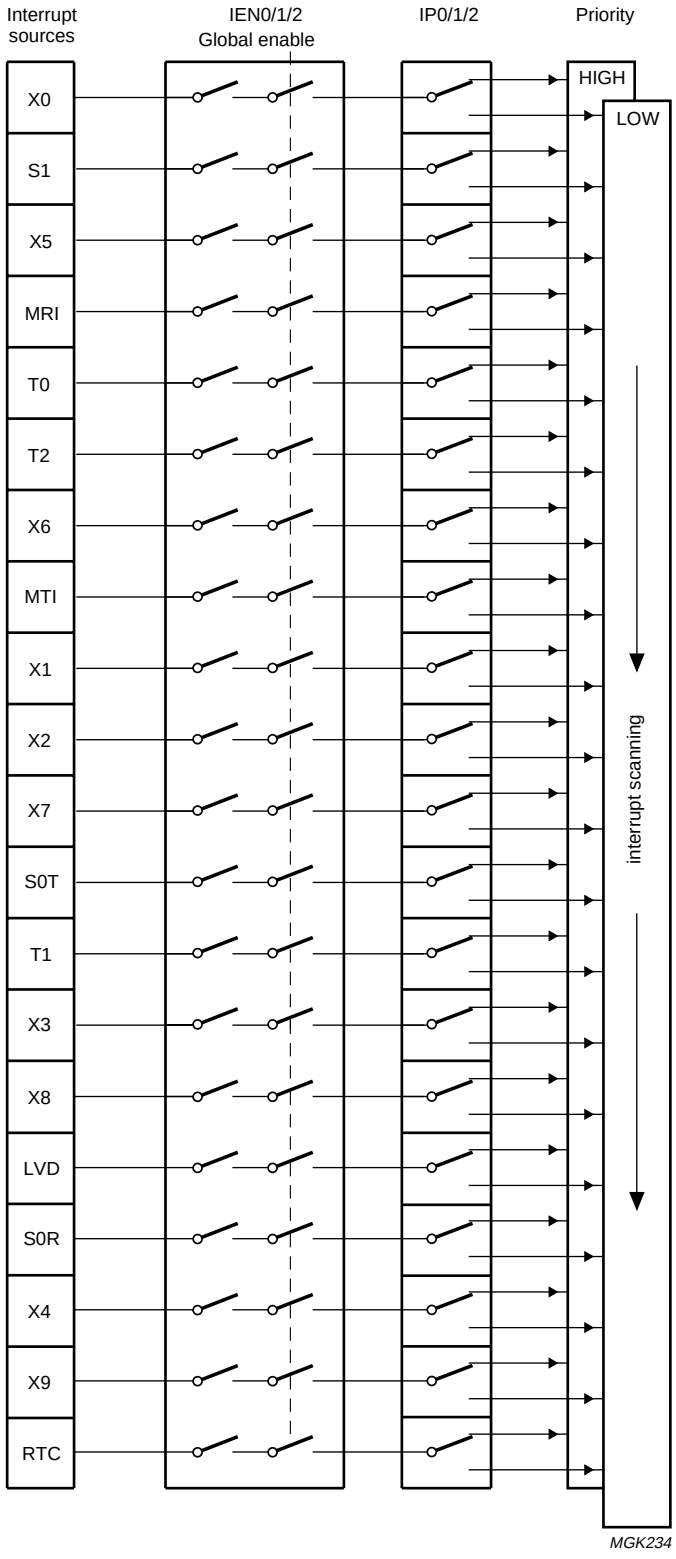


Fig.34 Interrupt system.

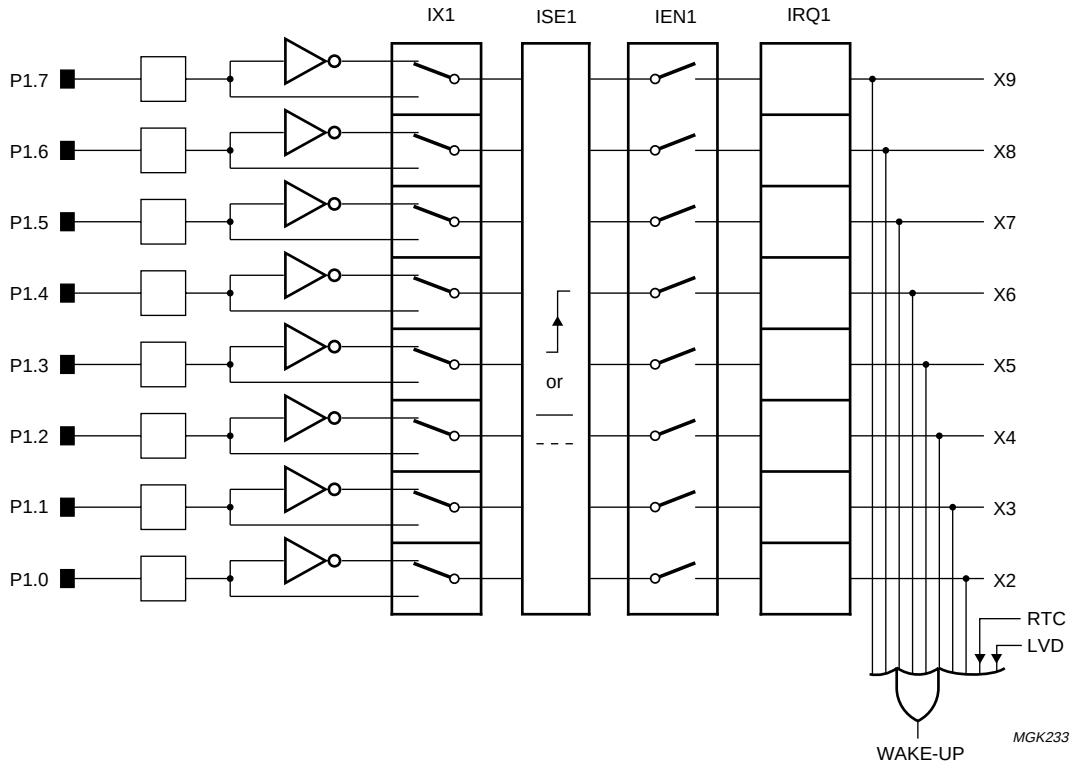


Fig.35 External interrupt configuration.

4.14.3 Interrupt Enable Register (IEN0)

Table 65 Interrupt Enable Register (SFR address A8H)

7	6	5	4	3	2	1	0
EA	ET2	ES1	ES0R	ET1	EX1	ET0	EX0

Table 66 Description of IEN0 bits

BIT(1)	SYMBOL	DESCRIPTION
IEN0.7	EA	General enable/disable control; if EA = 0, no interrupt is enabled; if EA = 1, any individually enabled interrupt will be accepted.
IEN0.6	ET2	enable T2 interrupt
IEN0.5	ES1	enable I^2C-bus interrupt
IEN0.4	ES0R	enable UART receiver interrupt
IEN0.3	ET1	enable Timer 1 interrupt (T1)
IEN0.2	EX1	enable external interrupt 1
IEN0.1	ET0	enable Timer 0 interrupt (T0)
IEN0.0	EX0	enable external interrupt 0

Note

1. Where: logic 0 = interrupt disabled; logic 1 = interrupt enabled.

4.14.4 Interrupt Enable Register (IEN1)

Table 67 Interrupt Enable Register (SFR address E8H)

7	6	5	4	3	2	1	0
EX9	EX8	EX7	EX6	EX5	EX4	EX3	EX2

Table 68 Description of IEN1 bits

BIT(1)	SYMBOL	DESCRIPTION
IEN1.7	EX9	enable external interrupt 9
IEN1.6	EX8	enable external interrupt 8
IEN1.5	EX7	enable external interrupt 7
IEN1.4	EX6	enable external interrupt 6
IEN1.3	EX5	enable external interrupt 5
IEN1.2	EX4	enable external interrupt 4
IEN1.1	EX3	enable external interrupt 3
IEN1.0	EX2	enable external interrupt 2

Note

1. Where: logic 0 = interrupt disabled; logic 1 = interrupt enabled.

4.14.5 Interrupt Enable Register (IEN2)

Table 69 Interrupt Enable Register (SFR address F1H)

7	6	5	4	3	2	1	0
–	–	–	ERTC	ELVD	ES0T	EMTI	EMRI

Table 70 Description of IEN2 bits

BIT(1)	SYMBOL	DESCRIPTION
IEN2.7	–	these 3 bits are reserved
IEN2.6	–	
IEN2.5	–	
IEN2.4	ERTC	enable RTC interrupt
IEN2.3	ELVD	enable Low Voltage Detector interrupt
IEN2.2	ESOT	enable UART transmitter interrupt
IEN2.1	EMTI	enable MSK modem transmitter interrupt
IEN2.0	EMRI	enable MSK modem receiver interrupt

Note

1. Where: logic 0 = interrupt disabled; logic 1 = interrupt enabled.

4.14.6 Interrupt Priority Register (IP0)

Table 71 Interrupt Priority Register (SFR address B8H)

7	6	5	4	3	2	1	0
–	PT2	PS1	PS0	PT1	PX1	PT0	PX0

Table 72 Description of IP0 bits

BIT(1)	SYMBOL	DESCRIPTION
IP0.7	–	Reserved
IP0.6	PT2	Timer 2 interrupt priority level
IP0.5	PS1	I^2C-bus interrupt priority level
IP0.4	PS0	UART SIO interrupt priority level
IP0.3	PT1	Timer 1 interrupt priority level
IP0.2	PX1	External interrupt 1 priority level
IP0.1	PT0	Timer 0 interrupt priority level
IP0.0	PX0	External interrupt 0 priority level

Note

1. Where: logic 0 = low priority; logic 1 = high priority.

4.14.7 Interrupt Priority Register (IP1)

Table 73 Interrupt Priority Register (SFR address F8H)

7	6	5	4	3	2	1	0
PX9	PX8	PX7	PX6	PX5	PX4	PX3	PX2

Table 74 Description of IP1 bits

BIT[1]	SYMBOL	DESCRIPTION
IP1.7	PX9	external interrupt 9 priority level
IP1.6	PX8	external interrupt 8 priority level
IP1.5	PX7	external interrupt 7 priority level
IP1.4	PX6	external interrupt 6 priority level
IP1.3	PX5	external interrupt 5 priority level
IP1.2	PX4	external interrupt 4 priority level
IP1.1	PX3	external interrupt 3 priority level
IP1.0	PX2	external interrupt 2 priority level

Note

1. Where: logic 0 = low priority; logic 1 = high priority.

4.14.8 Interrupt Priority Register (IP2)

Table 75 Interrupt Priority Register (SFR address F9H)

7	6	5	4	3	2	1	0
–	–	–	PRTC	PLVD	PS0T	PMTI	PMRI

Table 76 Description of IP2 bits

BIT[1]	SYMBOL	DESCRIPTION
IP2.7	–	These 3 bits are reserved
IP2.6	–	
IP2.5	–	
IP2.4	PRTC	RTC interrupt priority level
IP2.3	PLVD	Low Voltage Detector interrupt priority level
IP2.2	PS0T	UART transmitter interrupt priority level
IP2.1	PMTI	MSK modem transmitter interrupt priority level
IP2.0	PMRI	MSK modem receiver interrupt priority level

Note

1. Where: logic 0 = low priority; logic 1 = high priority.

4.14.9 Interrupt Polarity Register (IX1)

Writing either a logic 1 or logic 0 to any Interrupt Polarity Register bit sets the polarity level of the corresponding external interrupt to an active HIGH (rising edge) or active LOW (falling edge) respectively.

Table 77 Interrupt Polarity Register (SFR address E9H)

7	6	5	4	3	2	1	0
IX9	IX8	IX7	IX6	IX5	IX4	IX3	IX2

Table 78 Description of IX1 bits

BIT	SYMBOL	DESCRIPTION
IX1.7	IX9	external interrupt 9 polarity level
IX1.6	IX8	external interrupt 8 polarity level
IX1.5	IX7	external interrupt 7 polarity level
IX1.4	IX6	external interrupt 6 polarity level
IX1.3	IX5	external interrupt 5 polarity level
IX1.2	IX4	external interrupt 4 polarity level
IX1.1	IX3	external interrupt 3 polarity level
IX1.0	IX2	external interrupt 2 polarity level

4.14.10 Interrupt Sensitivity Register (ISE1)

Writing either a logic 1 or logic 0 to an Interrupt Sensitivity Register bit sets the type of the corresponding external interrupt to edge sensitive or level sensitive respectively.

Table 79 Interrupt Sensitivity Register (SFR address E1H)

7	6	5	4	3	2	1	0
ISE9	ISE8	ISE7	ISE6	ISE5	ISE4	ISE3	ISE2

Table 80 Description of ISE1 bits

BIT	SYMBOL	DESCRIPTION
ISE1.7	ISE9	external interrupt 9 sensitivity
ISE1.6	ISE8	external interrupt 8 sensitivity
ISE1.5	ISE7	external interrupt 7 sensitivity
ISE1.4	ISE6	external interrupt 6 sensitivity
ISE1.3	ISE5	external interrupt 5 sensitivity
ISE1.2	ISE4	external interrupt 4 sensitivity
ISE1.1	ISE3	external interrupt 3 sensitivity
ISE1.0	ISE2	external interrupt 2 sensitivity

4.14.11 Interrupt Request Flag Register (IRQ1)

Table 81 Interrupt Request Flag Register (SFR address C0H)

7	6	5	4	3	2	1	0
IQ9	IQ8	IQ7	IQ6	IQ5	IQ4	IQ3	IQ2

Table 82 Description of IRQ1 bits

BIT	SYMBOL	DESCRIPTION
IRQ1.7	IQ9	external interrupt 9 request flag
IRQ1.6	IQ8	external interrupt 8 request flag
IRQ1.5	IQ7	external interrupt 7 request flag
IRQ1.4	IQ6	external interrupt 6 request flag
IRQ1.3	IQ5	external interrupt 5 request flag
IRQ1.2	IQ4	external interrupt 4 request flag
IRQ1.1	IQ3	external interrupt 3 request flag
IRQ1.0	IQ2	external interrupt 2 request flag

4.14.12 Interrupt Related registers

The following registers are used in conjunction with the interrupt system.

Table 83 Interrupt Related registers

REGISTER	FUNCTION	SFR ADDRESS
IX1	Interrupt Polarity Register	E9H
ISE1	Interrupt Sensitivity Register	E1H
IRQ1	Interrupt Request Flag Register	C0H
IEN0	Interrupt Enable Register	A8H
IEN1	Interrupt Enable Register (INT2 to INT9)	E8H
IEN2	Interrupt Enable Register	E8H
IP0	Interrupt Priority Register	B8H
IP1	Interrupt Priority Register (INT2 to INT9)	F8H
IP2	Interrupt Priority Register	B8H

4.15 Idle and Power-down operation

Idle mode operation permits the interrupt, serial ports (UART and I^2C-bus), serial interfaces, RTC and timer blocks to continue to function while the clock to the CPU is halted.

The following functions remain active during the Idle mode. These functions may generate an interrupt or reset; thus ending the Idle mode.

- Timer 0, Timer 1 and Timer 2
- UART and I^2C-bus interface
- MSK modem
- External interrupts
- 32 kHz Real-Time Clock.

The Power-down operation stops the oscillator and reduces power consumption to a few micro-amps. This mode can only be activated by setting the PD bit in the PCON register or via the Low Voltage Detector. The Idle and Power-down clock configuration is shown in Fig.36.

4.15.1 IDLE MODE

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before the Idle mode is activated. Once in the Idle mode, the CPU status is preserved along with the Stack Pointer, Program Counter, Program Status Word and Accumulator. The RAM and all other registers maintain their data during Idle mode.

There are two ways to terminate the Idle mode:

1. Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware thus terminating the Idle mode. The interrupt is serviced, and following the RETI instruction, the next instruction to be executed will be the one following the instruction that put the device in the Idle mode. The flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or during the Idle mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When the Idle mode is terminated by an interrupt, the service routine can examine the status of the flag bits.
2. The second way of terminating the Idle mode is with an external hardware reset, or an internal reset caused by an overflow of the Watchdog Timer. Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (12 oscillator periods) to complete the reset operation. Reset redefines all SFRs but does not affect the on-chip RAM.

4.15.2 POWER-DOWN MODE

The instruction that sets PCON.1 is the last executed prior to going into the Power-down mode. Once in the Power-down mode, the oscillator is stopped. The contents of the on-chip RAM and the SFRs are preserved. The port pins output the value held in their respective SFRs. ALE is held LOW and $\overline{PSEN}$ is held HIGH.

The EEPROM should be switched off via register EECON before the Power-down mode is entered. Make sure not to enter the Power-down mode before a write or erase cycle is finished. For details on EEPROM operations, see Section 4.9.

The Power-down mode can also be entered and exited automatically by using the on-chip Low Voltage Detection circuit. This is described in Sections 4.18.2 and 4.18.3.

To reach lowest possible power consumption it is strongly recommended to write 00H in both the DTMF frequency registers HGF and LGF.

In the Power-down mode, V_{DD} may be reduced to minimize circuit power consumption. The supply voltage must not be reduced until the Power-down mode is entered, and must be restored before the hardware reset is applied which will free the oscillator. Reset should not be released until the oscillator has restarted and stabilized.

4.15.3 WAKE-UP FROM POWER-DOWN MODE

Setting the PD flag in the PCON register forces the controller into the Power-down mode. Setting this flag enables the controller to be woken-up from the Power-down mode with either the external interrupts INT2 to INT9, a reset operation, the LVD or via the RTC.

4.15.3.1 Wake-up using INT2 to INT9

If any of the interrupts INT2 to INT9 are enabled, the device can be woken-up from the Power-down mode with the external interrupts. To ensure that the oscillator is stable before the controller restarts, the internal clock will remain inactive for either 32 or 1024 oscillator periods. The length of the delay is programmable via the DELS bit in the PCON register. The delay is generated by an on-chip delay counter. After reset, 1024 oscillator periods delay is the default setting.

4.15.3.2 Wake-up using $\overline{RST}$

If using the $\overline{RST}$ pin for Wake-up, refer to Section 4.17.1.

4.15.3.3 Wake-up using LVD

The Power-down mode can be entered and exited automatically by using the on-chip Low Voltage Detection circuit. This is described in detail in the Section 4.18.2.

4.15.3.4 Wake-up using RTC

The on-chip 32 kHz Real-Time Clock (RTC) can be used to wake-up the microcontroller periodically without reset. This is described in detail in Section 4.4.

4.15.4 STATUS OF EXTERNAL PINS

The status of the external pins during Idle and Power-down mode is shown in Table 84. If the Power-down mode is activated whilst accessing external program memory, the port data that is held in the Special Function Register P2 is restored to Port 2. If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull-up transistor p1 (see Fig.5).

Table 84 Status of external pins during Idle and Power-down modes

MODE	MEMORY	ALE	$\overline{PSEN}$	PORT 0	PORT 1	PORT 2	PORT 3
Idle	internal	1	1	port data	port data	port data	port data
Idle	external	1	1	weak pull-up	port data	address	port data
Power-down	internal	0	1	port data	port data	port data	port data
Power-down	external	0	1	weak pull-up	port data	port data	port data

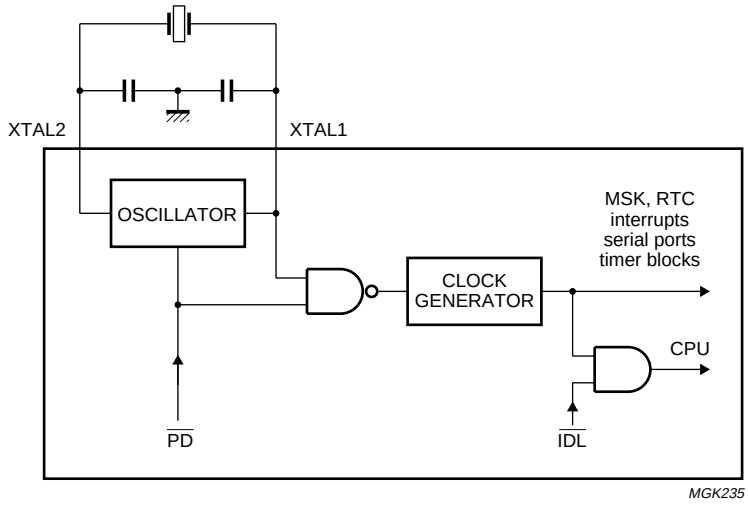


Fig.36 Internal Idle and Power-down clock configuration.

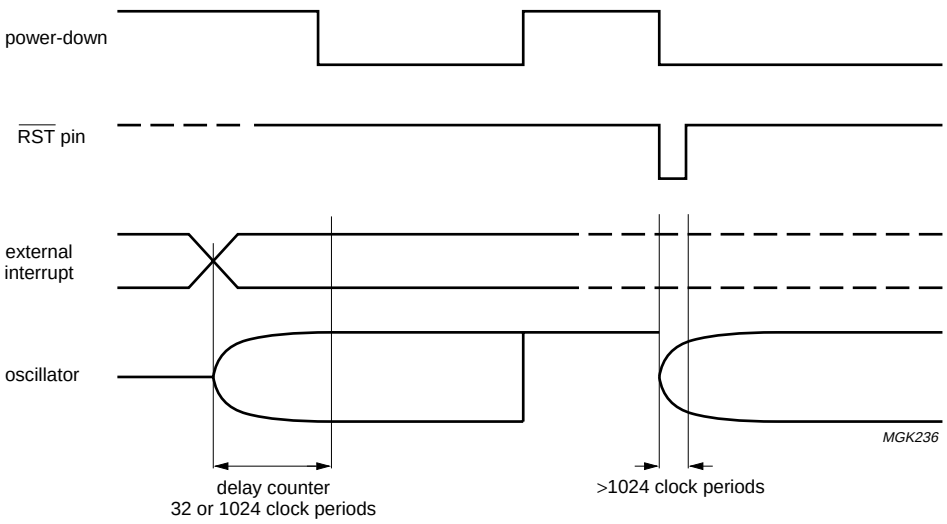


Fig.37 Wake-up operation.

4.15.5 Power Control Register (PCON)

The reduced power modes are activated by software using this Special Function Register. PCON is not bit addressable.

Table 85 Power Control Register (SFR address 87H)

7	6	5	4	3	2	1	0
SMOD	ARD	RFI	DELS	GF1	GF0	PD	IDL

Table 86 Description of PCON bits

BIT	SYMBOL	DESCRIPTION
PCON.7	SMOD	**Double Baud rate.** When set to a logic 1 the baud rate is doubled when the serial port SIO0 is being used in Modes 1, 2 or 3.
PCON.6	ARD	**AUX-RAM disable.** When ARD = 1, the internal AUX-RAM is disabled and all MOVX instructions access the external data memory - as it is with the standard PCB80C51.
PCON.5	RFI	**Reduced Radio Frequency Interference.** When set to a logic 1 the toggling of ALE pin is prohibited; this bit is cleared on reset.
PCON.4	DELS	**Delay Short.** To ensure that the oscillator is stable before the controller restarts after wake-up from power-down, the internal clock will remain inactive for either 32 or 1024 oscillator periods. DELS = 0 means a delay of 1024 clock periods (default at reset), DELS = 1 means a delay of 32 clock periods.
PCON.3	GF1	**General Purpose Flag bit.**
PCON.2	GF0	**General Purpose Flag bit.**
PCON.1	PD	**Power-down.** Setting this bit activates the Power-down mode; see note 1.
PCON.0	IDL	**Idle mode.** Setting this bit activates the Idle mode; see note 1.

Note

1. If logic 1s are written to PD and IDL at the same time, PD takes precedence. The reset value of PCON is (0XX00000).

4.16 Oscillator circuitry

The on-chip amplitude controlled oscillator circuitry is a single-stage inverting amplifier biased by an internal feedback resistor R_{fb}. The oscillator circuit is shown in Fig.38. When using a quartz crystal to drive the oscillator, no external components are needed. When using an external ceramic resonator to drive the oscillator, external components may be required depending upon the ceramic resonator specifications (refer to specific product specification). Two different configurations are shown in Fig.39(a) and Fig.39(b).

To drive the device with an external clock source, apply the external clock signal to XTAL1, and leave XTAL2 to float, as shown in Fig.39(c).

If the amplitude of the input signal is less than V_{DD} to V_{SS} or a sine wave is applied, capacitive decoupling is needed as shown in Fig.39(d).

In the Power-down mode the oscillator is stopped and XTAL1 and XTAL2 are internally pulled LOW. The current of the whole oscillator is switched off (signals ENABLECUR and ENABLECLK are inactive).

The system clock can be made available on a port pin by setting the ECLK bit in the ALTP register (see Section 4.10.4). This is useful in applications where the system clock of the microcontroller is used to clock other ICs. In this case the port latch of the port pin should be set to a logic 1 in order to avoid conflict between the system clock output and the port output.

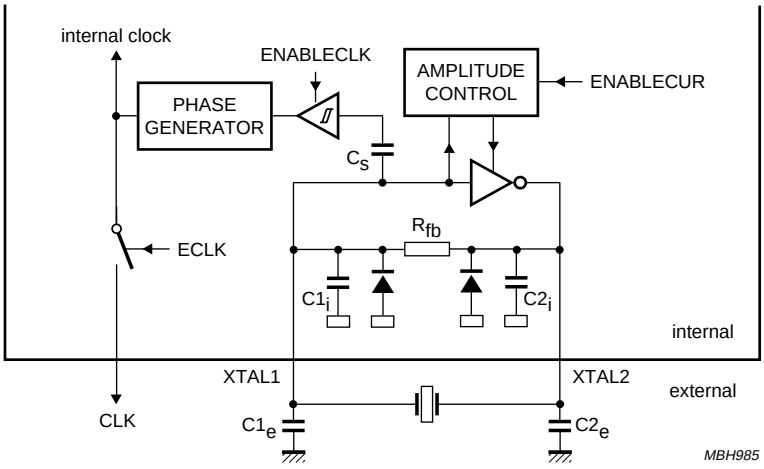


Fig.38 Oscillator.

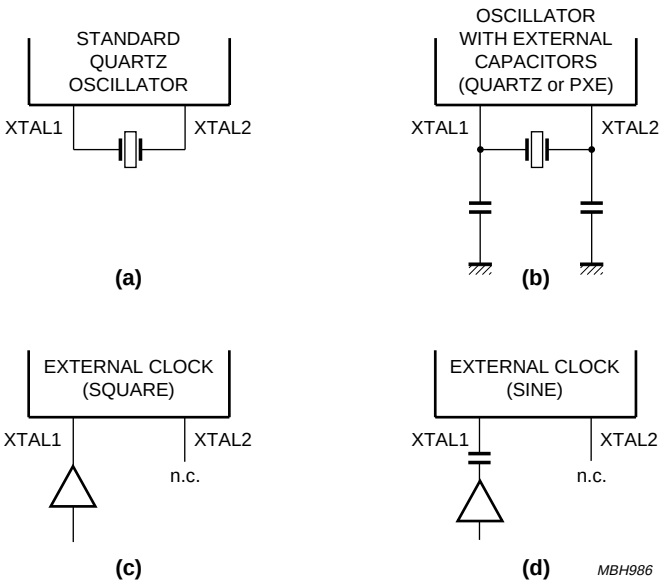


Fig.39 Alternative oscillator configurations.

4.17 Reset

To initialize the TELX microcontroller a reset is performed by one of three methods:

- Applying an external signal to the $\overline{RST}$ pin
- Via internal Power-on reset circuitry
- Via the on-chip Watchdog Timer.

The state of the port pins after a reset is given in the respective product specification.

The $\overline{RST}$ pin can function as an input or output pin. As an output pin it can be used to reset other IC's.

The internal RAM and the EEPROM are not affected by reset. When V_{DD} is turned on, the RAM contents are indeterminate.

4.17.1 External reset using the $\overline{RST}$ pin

The external reset input for the TELX microcontroller is the $\overline{RST}$ pin, it is asynchronous to the internal clock. A Schmitt trigger is used at the input for noise rejection. Immediately after the $\overline{RST}$ goes LOW the CPU responds by executing an internal reset, the SFRs and port pins adopt their reset state, ALE and $\overline{PSEN}$ are held HIGH. As long as $\overline{RST}$ pin remains LOW, the slot generator is halted at timeslot 1 and the reset state is maintained. When $\overline{RST}$ goes HIGH, the slot generator is started and program execution starts after 2 machine cycles.

4.17.2 External Power-on reset using the $\overline{RST}$ pin

An automatic reset can be obtained by connecting the $\overline{RST}$ pin to V_{SS} via a capacitor. At power-on, the voltage on the $\overline{RST}$ pin is equal to V_{SS} and increases from V_{SS} as the capacitor charges through the internal resistor (R_{RST}) to V_{DD}. V_{RST} must remain below the higher threshold of the Schmitt trigger long enough for the oscillator to become stable. The time required is approximately 1024 oscillator periods. The reset configuration is shown in Fig.40.

4.17.3 Internal Power-on/Power-off reset (POR)

The device contains an on-chip Power-on-reset circuit which activates a reset as long as V_{DD} is below a predefined level (V_{PORH}). If V_{DD} exceeds V_{PORH}, the oscillator will start-up. However, to ensure that the oscillator is stable before the controller starts, the clock signals are gated away from the CPU for 1024 oscillator periods.After this delay the slot generator is started and program execution starts after 2 machine cycles.

The Power-on-reset circuit also ensures, that the microcontroller will be switched-off as soon as a second predefined level (V_{PORL}) is reached as V_{DD} decreases.

The on-chip POR circuit can also be switched off by connecting the PORACTIVE pin to V_{SS}. This reduces the Power-down current even further and can be chosen if external reset circuitry is used.

If the POR signal is active, the $\overline{RST}$ pin will be pulled LOW.

The state of internal registers after a reset are given in the product specifications.

4.17.4 Trip points of POR (Power-On/Off-Reset)

At power-up or at varying supply voltage, the POR circuit will ensure that the microcontroller is reset correctly at predefined levels. The POR trip points are defined as follows:

- POR trip level HIGH (V_{PORH}). When this level is reached at rising V_{DD}, the internal reset signal is deactivated (oscillator released and CPU released after a delay of 1024 or 32 clock periods).
- POR trip level LOW (V_{PORL}). When this level is reached at falling V_{DD}, the internal reset signal is activated (oscillator stopped).

The minimum V_{DD} for the microcontroller depends on the clock frequency used.

Eight different voltages for trip level HIGH (V_{PORH}) can be chosen. The hysteresis $V_{PORH} - V_{PORL}$ can be chosen either to be fixed at a level relative to V_{PORH}, typically 100 mV (so V_{PORL} will be $V_{PORH} - 100$ mV), or at a very low value typically at 1.3 V. Any combination of a V_{PORH} option and a hysteresis option can be chosen. The chosen option for the trip levels and the type of hysteresis can be checked by reading the Reset Status Register (RSTAT); see Section 4.17.5.

The hysteresis option with V_{PORL} = 1.3 V is foreseen for the case when no reset is wanted when V_{DD} is decreasing below minimum V_{DD} (operating) but still being above minimum V_{DD} (RAM retention).

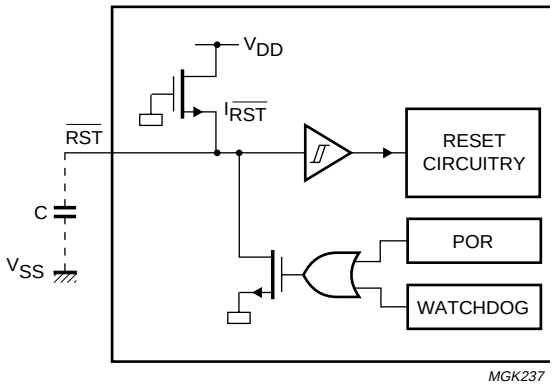


Fig.40 Reset configuration.

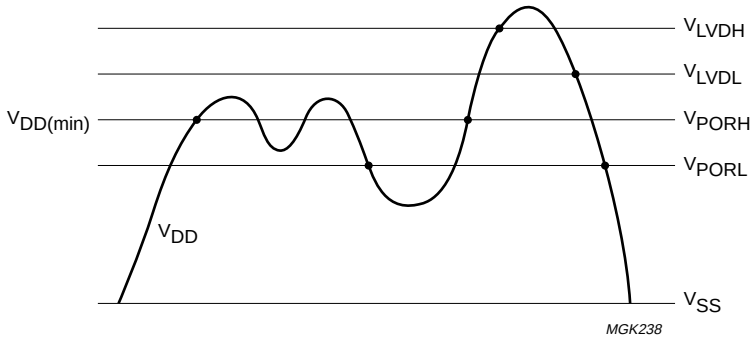


Fig.41 Power-on reset switching level.

4.17.5 Reset Status Register (RSTAT)

This 8-bit register gives the chosen reset option for the POR circuit and for the on-chip oscillator.

Table 87 Reset Status register RSTAT (SFR address E6H)

7	6	5	4	3	2	1	0
–	–	–	O3	O2	O1	O0	GMH

Table 88 Description of RSTAT bits

BIT	SYMBOL	DESCRIPTION
RSTAT.7	–	These 3-bits are not used.
RSTAT.6	–	
RSTAT.5	–	
RSTAT.4	O3	This bit defines the hysteresis of the POR.
RSTAT.3	O2	These 3-bits define the high trip point of the POR; see Table 89.
RSTAT.2	O1	
RSTAT.1	O0	
RSTAT.0	GMH	This bit reflects the chosen transconductance of the on-chip oscillator; its value should not be changed, since this could affect the behaviour of the oscillator

Table 89 POR trip points

PARAMETER	OPTION	RSTAT CONTENTS	MIN.	TYP.	MAX.	UNIT
Accuracy of V_{PORH}			–10		10	%
Trip level HIGH (V_{PORH})	Option 1	XXXX 000X	1.75	1.94	2.13	V
	Option 2	XXXX 001X	1.84	2.04	2.24	V
	Option 3	XXXX 010X	1.94	2.15	2.37	V
	Option 4	XXXX 011X	2.03	2.26	2.49	V
	Option 5	XXXX 100X	2.13	2.37	2.61	V
	Option 6	XXXX 101X	2.42	2.69	2.96	V
	Option 7	XXXX 110X	2.52	2.80	3.08	V
	Option 8	XXXX 111X	2.61	2.90	3.19	V
Trip level LOW (V_{PORL})	Option A	XXX0 XXXX	V_{PORH} – 0.15	V_{PORH} – 0.10	V_{PORH} – 0.05	V
	Option B	XXX1 XXXX	–	1.30	–	V

4.17.6 Internal reset via the watchdog

The Watchdog which is available on the $\overline{RST}$ pin is described in Section 4.8.3.

4.18 Low Voltage Detection

The Low Voltage Detection (LVD) is a feature which can be used to determine if a certain voltage level of V_{DD} has been reached, e.g. low voltage warning for EEPROM or DTMF operation, or for normal operation. The LVD is programmed by software via the LVD Control Register (LVDCON).

An active output from the Low Voltage Detection block will set the LVDI bit in the LVD Control Register, this can be detected by software, and an internal interrupt will be generated providing the ELVD bit in the IEN2 register is set. The LVDI bit must be reset by software. The LVD interrupt can be activated on a rising or falling edge of V_{DD}. The selection is made using the LVDX bit in the LVDCON register.

The state of the LVD signal is indicated by the LVDS bit in LVDCON. LVDS can only be read.

The LVD can also be used to enter and exit the Power-down mode automatically without first setting the PD bit in the PCON register. This feature is further described in Section 4.18.2.

Programming of the LVD trip points is done by software for both the OTP and ROM versions, via the LVDCON register. The trip levels can be changed any time during program execution. Ten different options for the high trip level (V_{LVDH}) plus an option ‘Off’ for power saving are offered. The hysteresis between the high and low trip points ($V_{LVDH} - V_{LVDL}$) is typically 100 mV. The various options are listed in Table 92.

The variation of the different trip points for POR and LVD are related as both blocks use the same on-chip bandgap reference. The levels of the POR and LVD trip points should be programmed to be in the following order: $V_{LVDH} > V_{LVDL} > V_{PORH} > V_{PORL}$ (refer to Fig.41).

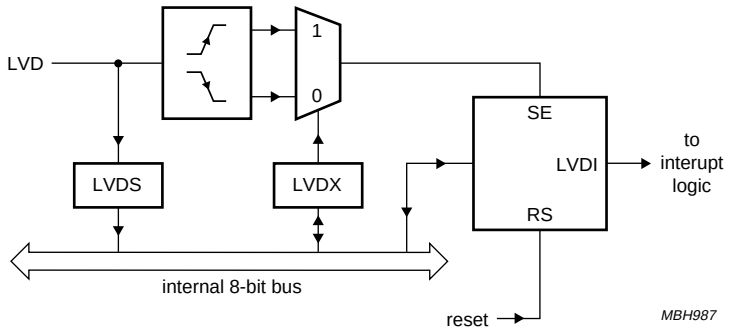


Fig.42 LVD interrupt logic.

4.18.1 LVD CONTROL REGISTER (LVDCON)

Table 90 LVD Control Register (SFR address F2H)

7	6	5	4	3	2	1	0
LVDPEN	LVDI	LVDX	LVDS	SEL3	SEL2	SEL1	SEL0

Table 91 Description of LVDCON bits

BIT	SYMBOL	DESCRIPTION
LVDCON.7	LVDPEN	**LVD Power-down enable.**
LVDCON.6	LVDI	**LVD Interrupt Flag.** Set by hardware, reset by software.
LVDCON.5	LVDX	**LVD Polarity Flag.** If LVDX = 0, then the LVD interrupt is activated on the falling edge of V_{DD}. If LVDX = 1, then the LVD interrupt is activated on the rising edge of V_{DD}. See Fig.42.
LVDCON.4	LVDS	**LVD Status.** read only; see Fig.42
LVDCON.3	SEL3	**Select option.** These 4-bits select the LVD trip level option; see Table 92.
LVDCON.2	SEL2	
LVDCON.1	SEL1	
LVDCON.0	SEL0	

Table 92 Selection and levels of the LVD trip points

PARAMETER	SEL3	SEL2	SEL1	SEL0	OPTION	MIN.	TYP.	MAX.	UNIT
Accuracy of V_{LVDH}	–	–	–	–	–	–10	–	+10	%
Trip level HIGH (V_{LVDH})	0	0	0	0	Option 0	Off	Off	Off	
	0	0	0	1	Option 1	1.75	1.94	2.13	V
	0	0	1	0	Option 2	1.84	2.04	2.24	V
	0	0	1	1	Option 3	1.94	2.15	2.37	V
	0	1	0	0	Option 4	2.03	2.26	2.49	V
	0	1	0	1	Option 5	2.13	2.37	2.61	V
	0	1	1	0	Option 6	2.22	2.47	2.72	V
	0	1	1	1	Option 7	2.32	2.58	2.84	V
	1	0	0	0	Option 8	2.42	2.69	2.96	V
	1	0	0	1	Option 9	2.52	2.80	3.08	V
	1	0	1	0	Option 10	2.61	2.90	3.19	V
Trip level LOW (V_{LVDL})	–	–	–	–	–	V_{LVDH} – 0.15	V_{LVDH} – 0.10	V_{LVDH} – 0.05	V

4.18.2 Entering and exiting Power-down mode automatically using LVD.

The TELX family offers the feature of entering the Power-down mode automatically, without any external voltage detector, see Fig.43. If the LVDPEN bit in the LVDCON register is set, the TELX microcontroller will enter the Power-down mode automatically as soon as the low LVD trip level (V_{LVDL}) is reached when the supply voltage decreases. This is useful when the Power-down mode must be entered quickly before the $V_{DD(min)}$ level is reached.

The trip point for V_{LVDL} is set by software as described in Section 4.18. Exit from Power-down mode is done either via a reset, or when the high LVD trip level (V_{LVDH}) is reached when the supply voltage rises.

4.18.3 Entering and exiting Power-down mode explicitly, using an LVD interrupt.

The Power-down mode can also be entered via an interrupt generated by the LVD and a corresponding interrupt software routine. In the interrupt routine, actions such as saving data in EEPROM or displaying a warning could be taken before entering the Power-down mode by setting LVDPEN. If V_{DD} rises above V_{LVDH}, the microcontroller exits the Power-down mode and resumes program execution inside the interrupt routine, see Fig.44.

Note that the next instruction after setting the LVDPEN bit will be executed before the microcontroller enters the Power-down mode. If this is not wanted, a NOP instruction should be inserted directly after the instruction setting the LVDPEN bit.

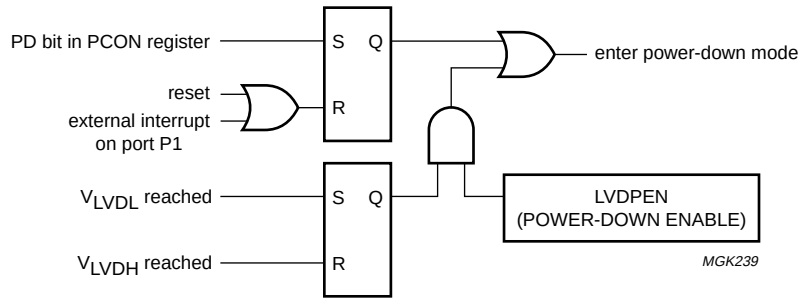


Fig.43 Power-down mode entry.

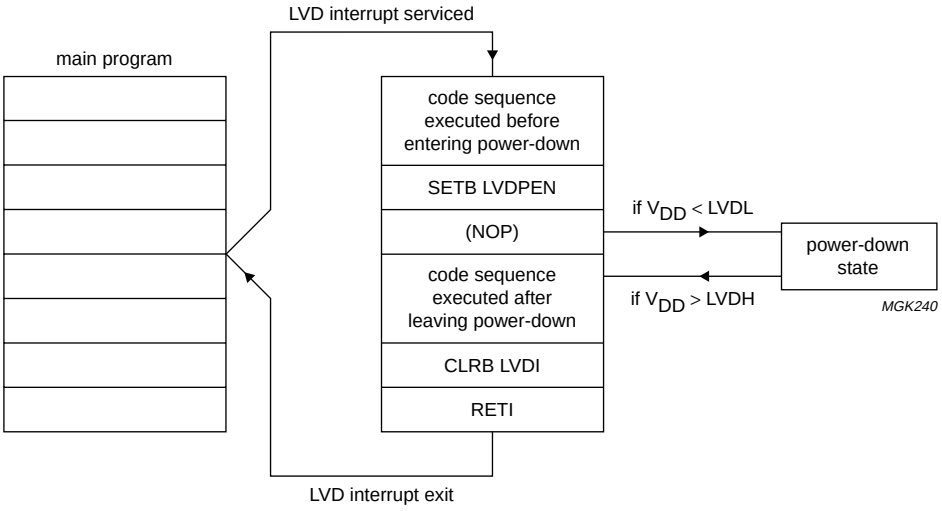


Fig.44 Power-down entry via interrupt.

5 INSTRUCTION SET

The TELX Family uses a powerful instruction set which permits the expansion of on-chip CPU peripherals and optimizes byte efficiency and execution speed. Assigned opcodes add new high-power operation and permit new addressing modes. The instruction set consists of 49 single-byte, 46 two-byte and 16 three-byte instructions. When using a 3.58 MHz oscillator, 64 instructions execute in 1.68 μs and 45 instructions execute in 3.35 μs. Multiply and divide instructions execute in 6.70 μs.

Table 93 Instruction Set

MNEMONIC		DESCRIPTION	BYTES	CYCLES	OPCODE (HEX)
Arithmetic operations					
ADD	A, Rr	add register to A	1	1	2*
ADD	A, direct	add direct byte to A	2	1	25
ADD	A, @Ri	add indirect RAM to A	1	1	26, 27
ADD	A, #data	add immediate data to A	2	1	24
ADDC	A, Rr	add register to A with carry flag	1	1	3*
ADDC	A, direct	add direct byte to A with carry flag	2	1	35
ADDC	A, @Ri	add indirect RAM to A with carry flag	1	1	36, 37
ADDC	A, #data	add immediate data to A with carry flag	2	1	34
SUBB	A, Rr	subtract register from A with borrow	1	1	9*
SUBB	A, direct	subtract direct byte from A with borrow	2	1	95
SUBB	A, @Ri	subtract indirect RAM from A with borrow	1	1	96, 97
SUBB	A, #data	subtract immediate data from A with borrow	2	1	94
INC	A	increment A	1	1	04
INC	Rr	increment register	1	1	0*
INC	direct	increment direct byte	2	1	05
INC	@Ri	increment indirect RAM	1	1	06, 07
DEC	A	decrement A	1	1	14
DEC	Rr	decrement register	1	1	1*
DEC	direct	decrement direct byte	2	1	15
DEC	@Ri	decrement indirect RAM	1	1	16, 17
INC	DPTR	increment data pointer	1	2	A3
MUL	AB	multiply A and B	1	4	A4
DIV	AB	divide A by B	1	4	84
DA	A	decimal adjust A	1	1	D4

MNEMONIC		DESCRIPTION	BYTES	CYCLES	OPCODE (HEX)
Logic operations					
ANL	A, Rr	AND register to A	1	1	5*
ANL	A, direct	AND direct byte to A	2	1	55
ANL	A, @Ri	AND indirect RAM to A	1	1	56, 57
ANL	A, #data	AND immediate data to A	2	1	54
ANL	direct, A	AND A to direct byte	2	1	52
ANL	direct, #data	AND immediate data to direct byte	3	2	53
ORL	A, Rr	OR register to A	1	1	4*
ORL	A, direct	OR direct byte to A	2	1	45
ORL	A, @Ri	OR indirect RAM to A	1	1	46, 47
ORL	A, #data	OR immediate data to A	2	1	44
ORL	direct, A	OR A to direct byte	2	1	42
ORL	direct, #data	OR immediate data to direct byte	3	2	43
XRL	A, Rr	exclusive-OR register to A	1	1	6*
XRL	A, direct	exclusive-OR direct byte to A	2	1	65
XRL	A, @Ri	exclusive-OR indirect RAM to A	1	1	66, 67
XRL	A, #data	exclusive-OR immediate data to A	2	1	64
XRL	direct, A	exclusive-OR A to direct byte	2	1	62
XRL	direct, #data	exclusive-OR immediate data to direct byte	3	2	63
CLR	A	clear A	1	1	E4
CPL	A	complement A	1	1	F4
RL	A	rotate A left	1	1	23
RLC	A	rotate A left through the carry flag	1	1	33
RR	A	rotate A right	1	1	03
RRC	A	rotate A right through the carry flag	1	1	13
SWAP	A	swap nibbles within A	1	1	C4

MNEMONIC		DESCRIPTION	BYTES	CYCLES	OPCODE (HEX)
Data transfer					
MOV	A, Rr	move register to A	1	1	E*
MOV	A, direct**	move direct byte to A	2	1	E5
MOV	A, @Ri	move indirect RAM to A	1	1	E6, E7
MOV	A, #data	move immediate data to A	2	1	74
MOV	Rr, A	move A to register	1	1	F*
MOV	Rr, direct	move direct byte to register	2	2	A*
MOV	Rr, #data	move immediate data to register	2	1	7*
MOV	direct, A	move A to direct byte	2	1	F5
MOV	direct, Rr	move register to direct byte	2	2	8*
MOV	direct, direct	move direct byte to direct byte	3	2	85
MOV	direct, @Ri	move indirect RAM to direct byte	2	2	86, 87
MOV	direct, #data	move immediate data to direct byte	3	2	75
MOV	@RI, A	move A to indirect RAM	1	1	F6, F7
MOV	@Ri, direct	move direct byte to indirect RAM	2	2	A6, A7
MOV	@Ri, #data	move immediate data to indirect RAM	3	1	76, 77
MOV	DPTR, #data 16	load data pointer with a 16-bit constant	3	2	90
MOVC	A, @A + DPTR	move code byte relative to DPTR to A	1	2	93
MOVC	A, @A + PC	move code byte relative to PC to A	1	2	83
MOVX	A, @Ri	move external RAM (8-bit address) to A	1	2	E2, E3
MOVX	A, @DPTR	move external RAM (16-bit address) to A	1	2	E0
MOVX	@Ri, A	move A to external RAM (8-bit address)	1	2	F2, F3
MOVX	@DPTR, A	move A to external RAM (16-bit address)	1	2	F0
PUSH	direct	push direct byte onto stack	2	2	C0
POP	direct	pop direct byte from stack	2	2	D0
XCH	A, Rr	exchange register with A	1	1	C*
XCH	A, direct	exchange direct byte with A	2	1	C5
XCH	A, @Ri	exchange indirect RAM with A	1	1	C6, C7
XCHD	A, @Ri	exchange LOW-order nibble indirect RAM with A	1	1	D6, D7

MNEMONIC		DESCRIPTION	BYTES	CYCLES	OPCODE (HEX)
Boolean variable manipulation					
CLR	C	clear carry flag	1	1	C3
CLR	bit	clear direct bit	2	1	C2
SETB	C	set carry flag	1	1	D3
SETB	bit	set direct bit	2	1	D2
CPL	C	complement carry flag	1	1	B3
CPL	bit	complement direct bit	2	1	B2
ANL	C, bit	AND direct bit to carry flag	2	2	82
ANL	C, /bit	AND complement of direct bit to carry flag	2	2	B0
ORL	C, bit	OR direct bit to carry flag	2	2	72
ORL	C, /bit	OR complement of direct bit to carry flag	2	2	A0
MOV	C, bit	move direct bit to carry flag	2	1	A2
MOV	bit, C	move carry flag to direct bit	2	2	92
Program and machine control					
ACALL	addr11	absolute subroutine call	2	2	•1 addr
LCALL	addr16	long subroutine call	3	2	12
RET		return from subroutine	1	2	22
RETI		return from interrupt	1	2	32
AJMP	addr11	absolute jump	2	2	♦1 addr
LJMP	addr16	long jump	3	2	02
SJMP	rel	short jump (relative address)	2	2	80
JMP	@A + DPTR	jump indirect relative to the DPTR	1	2	73
JZ	rel	jump if A is zero	2	2	60
JNZ	rel	jump if A is not zero	2	2	70
JC	rel	jump if carry flag is set	2	2	40
JNC	rel	jump if carry flag is not set	2	2	50
JB	bit, rel	jump if direct bit is set	3	2	20
JNB	bit, rel	jump if direct bit is not set	3	2	30
JBC	bit, rel	jump if direct bit is set and clear bit	3	2	10
CJNE	A, direct, rel	compare direct to A and jump if not equal	3	2	B5
CJNE	A, #data, rel	compare immediate to A and jump if not equal	3	2	B4
CJNE	Rr, #data, rel	compare immediate to register and jump if not equal	3	2	B*
CJNE	@Ri, #data, rel	compare immediate to indirect and jump if not equal	3	2	B6, B7
DJNZ	Rr, rel	decrement register and jump if not zero	2	2	D*
DJNZ	direct, rel	decrement direct and jump if not zero	3	2	D5
NOP		no operation	1	1	00

Table 94 Notation for data addressing modes

SYMBOL	DESCRIPTION
Rr	working registers R0 to R7
direct	128 internal RAM locations and any special function register (SFR)
@Ri	indirect internal RAM location addressed by register R0 or R1
#data	8-bit constant included in instruction
#data 16	16-bit constant included as bytes 2 and 3 of instruction
bit	direct addressed bit in internal RAM or SFR
addr16	16-bit destination address; used by LCALL and LJMP; the branch will be anywhere within the 64 kbyte program memory address space
addr11	11-bit destination address; used by ACALL and AJMP; the branch will be within the same 2 kbyte page of program memory as the first byte of the following instruction
rel	signed (two's complement) 8-bit offset byte; used by SJMP and all conditional jumps. Range is –128 to +127 bytes relative to first byte of the following instruction

Table 95 Hexadecimal opcode cross-reference

SYMBOL	DESCRIPTION
*	8, 9, A, B, C, D, E, F.
•	11, 31, 51, 71, 91, B1, D1, F1.
♦	01, 21, 41, 61, 81, A1, C1, E1.

5.1 Instruction Map

first hexadecimal character of opcode

second hexadecimal character of opcode

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	AJMP addr 11	LJMP addr 16	RR A	INC A	INC dir	INC @ Ri		INC Rr							
							0	1	0	1	2	3	4	5	6	7
1	JBC bit, rel	ACALL addr 11	LCALL addr 16	RRC A	DEC A	DEC dir	DEC @ Ri		DEC Rr							
							0	1	0	1	2	3	4	5	6	7
2	JB bit, rel	AJMP addr 11	RET	RL A	ADD A, # data	ADD A, dir	ADD A, @ Ri		ADD A, Rr							
							0	1	0	1	2	3	4	5	6	7
3	JNB bit, rel	ACALL addr 11	RETI	RLC A	ADDC A, # data	ADDC A, dir	ADDC A, @ Ri		ADDC A, Rr							
							0	1	0	1	2	3	4	5	6	7
4	JC rel	AJMP addr 11	ORL dir, A	ORL dir, # data	ORL A, # data	ORL A, dir	ORL A, @ Ri		ORL A, Rr							
							0	1	0	1	2	3	4	5	6	7
5	JNC rel	ACALL addr 11	ANL dir, A	ANL dir, # data	ANL A, # data	ANL A, dir	ANL A, @ Ri		ANL A, Rr							
							0	1	0	1	2	3	4	5	6	7
6	JZ rel	AJMP addr 11	XRL dir, A	XRL dir, # data	XRL A, # data	XRL A, dir	XRL A, @ Ri		XRL A, Rr							
							0	1	0	1	2	3	4	5	6	7
7	JNZ rel	ACALL addr 11	ORL C, bit	JMP @ A+DPTR	MOV A, # data	MOV dir, # data	MOV @ Ri, # data		MOV Rr, # data							
							0	1	0	1	2	3	4	5	6	7
8	SJMP rel	AJMP addr 11	ANL C, bit	MOVC A, @ A+PC	DIV AB	MOV dir, dir	MOV dir, @ Ri		MOV dir, Rr							
							0	1	0	1	2	3	4	5	6	7
9	MOV DPTR, # data 16	ACALL addr 11	MOV bit, C	MOVC A, @ A+DPTR	SUBB A, # data	SUBB A, dir	SUBB A, @ Ri		SUBB A, Rr							
							0	1	0	1	2	3	4	5	6	7
A	ORL C, / bit	AJMP addr 11	MOV bit, C	INC DPTR	MUL AB		MOV @ Ri, dir		MOV Rr, dir							
							0	1	0	1	2	3	4	5	6	7
B	ANL C, / bit	ACALL addr 11	CPL bit	CPL C	CJNE A, # data, rel	CJNE A, dir, rel	CJNE @ Ri, # data, rel		CJNE Rr, # data, rel							
							0	1	0	1	2	3	4	5	6	7
C	PUSH dir	AJMP addr 11	CLR bit	CLR C	SWAP A	XCH A, dir	XCH A, @ Ri		XCH A, Rr							
							0	1	0	1	2	3	4	5	6	7
D	POP dir	ACALL addr 11	SETB bit	SETB C	DA A	DJNZ dir, rel	XCHD A, @ Ri		DJNZ Rr, rel							
							0	1	0	1	2	3	4	5	6	7
E	MOVX A, @ DPTR	AJMP addr 11	MOVX A, @ Ri		CLR A	MOV A, dir *	MOV A, @ Ri		MOV A, Rr							
			0	1			0	1	0	1	2	3	4	5	6	7
F	MOVX @ DPTR, A	ACALL addr 11	MOVX @ Ri, A		CPL A	MOV dir, A	MOV @ Ri, A		MOV Rr, A							
			0	1			0	1	0	1	2	3	4	5	6	7

MLA632 - 1

* MOV A, ACC is not a valid instruction.

6 DEFINITIONS

Data sheet status	
Objective specification	This data sheet contains target or goal specifications for product development.
Preliminary specification	This data sheet contains preliminary data; supplementary data may be published later.
Product specification	This data sheet contains final product specifications.
Limiting values	
Limiting values given are in accordance with the Absolute Maximum Rating System (IEC 134). Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Characteristics sections of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability.	
Application information	
Where application information is given, it is advisory and does not form part of the specification.	

7 LIFE SUPPORT APPLICATIONS

These products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Philips customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Philips for any damages resulting from such improper use or sale.

8 PURCHASE OF PHILIPS I²C COMPONENTS

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 specification defined by Philips. This specification can be ordered using the code 9398 393 40011.

NOTES

NOTES

NOTES

Philips Semiconductors – a worldwide company

Argentina: see South America

Australia: 34 Waterloo Road, NORTH RYDE, NSW 2113,
Tel. +61 2 9805 4455, Fax. +61 2 9805 4466

Austria: Computerstr. 6, A-1101 WIEN, P.O. Box 213,
Tel. +43 1 60 101, Fax. +43 1 60 101 1210

Belarus: Hotel Minsk Business Center, Bld. 3, r. 1211, Volodarski Str. 6,
220050 MINSK, Tel. +375 172 200 733, Fax. +375 172 200 773

Belgium: see The Netherlands

Brazil: see South America

Bulgaria: Philips Bulgaria Ltd., Energoproject, 15th floor,
51 James Bourchier Blvd., 1407 SOFIA,
Tel. +359 2 689 211, Fax. +359 2 689 102

Canada: PHILIPS SEMICONDUCTORS/COMPONENTS,
Tel. +1 800 234 7381

China/Hong Kong: 501 Hong Kong Industrial Technology Centre,
72 Tat Chee Avenue, Kowloon Tong, HONG KONG,
Tel. +852 2319 7888, Fax. +852 2319 7700

Colombia: see South America

Czech Republic: see Austria

Denmark: Prags Boulevard 80, PB 1919, DK-2300 COPENHAGEN S,
Tel. +45 32 88 2636, Fax. +45 31 57 0044

Finland: Sinikalliontie 3, FIN-02630 ESPOO,
Tel. +358 9 615800, Fax. +358 9 61580920

France: 4 Rue du Port-aux-Vins, BP317, 92156 SURESNES Cedex,
Tel. +33 1 40 99 6161, Fax. +33 1 40 99 6427

Germany: Hammerbrookstraße 69, D-20097 HAMBURG,
Tel. +49 40 23 53 60, Fax. +49 40 23 536 300

Greece: No. 15, 25th March Street, GR 17778 TAVROS/ATHENS,
Tel. +30 1 4894 339/239, Fax. +30 1 4814 240

Hungary: see Austria

India: Philips INDIA Ltd, Shivsagar Estate, A Block, Dr. Annie Besant Rd.
Worli, MUMBAI 400 018, Tel. +91 22 4938 541, Fax. +91 22 4938 722

Indonesia: see Singapore

Ireland: Newstead, Clonskeagh, DUBLIN 14,
Tel. +353 1 7640 000, Fax. +353 1 7640 200

Israel: RAPAC Electronics, 7 Kehilat Saloniki St, PO Box 18053,
TEL AVIV 61180, Tel. +972 3 645 0444, Fax. +972 3 649 1007

Italy: PHILIPS SEMICONDUCTORS, Piazza IV Novembre 3,
20124 MILANO, Tel. +39 2 6752 2531, Fax. +39 2 6752 2557

Japan: Philips Bldg 13-37, Kohnan 2-chome, Minato-ku, TOKYO 108,
Tel. +81 3 3740 5130, Fax. +81 3 3740 5077

Korea: Philips House, 260-199 Itaewon-dong, Yongsan-ku, SEOUL,
Tel. +82 2 709 1412, Fax. +82 2 709 1415

Malaysia: No. 76 Jalan Universiti, 46200 PETALING JAYA, SELANGOR,
Tel. +60 3 750 5214, Fax. +60 3 757 4880

Mexico: 5900 Gateway East, Suite 200, EL PASO, TEXAS 79905,
Tel. +9-5 800 234 7381

Middle East: see Italy

Netherlands: Postbus 90050, 5600 PB EINDHOVEN, Bldg. VB,
Tel. +31 40 27 82785, Fax. +31 40 27 88399

New Zealand: 2 Wagener Place, C.P.O. Box 1041, AUCKLAND,
Tel. +64 9 849 4160, Fax. +64 9 849 7811

Norway: Box 1, Manglerud 0612, OSLO,
Tel. +47 22 74 8000, Fax. +47 22 74 8341

Philippines: Philips Semiconductors Philippines Inc.,
106 Valero St. Salcedo Village, P.O. Box 2108 MCC, MAKATI,
Metro MANILA, Tel. +63 2 816 6380, Fax. +63 2 817 3474

Poland: Ul. Lukiska 10, PL 04-123 WARSZAWA,
Tel. +48 22 612 2831, Fax. +48 22 612 2327

Portugal: see Spain

Romania: see Italy

Russia: Philips Russia, Ul. Usatcheva 35A, 119048 MOSCOW,
Tel. +7 095 755 6918, Fax. +7 095 755 6919

Singapore: Lorong 1, Toa Payoh, SINGAPORE 1231,
Tel. +65 350 2538, Fax. +65 251 6500

Slovakia: see Austria

Slovenia: see Italy

South Africa: S.A. PHILIPS Pty Ltd., 195-215 Main Road Martindale,
2092 JOHANNESBURG, P.O. Box 7430 Johannesburg 2000,
Tel. +27 11 470 5911, Fax. +27 11 470 5494

South America: Rua do Rocio 220, 5th floor, Suite 51,
04552-903 São Paulo, SÃO PAULO - SP, Brazil,
Tel. +55 11 821 2333, Fax. +55 11 829 1849

Spain: Balmes 22, 08007 BARCELONA,
Tel. +34 3 301 6312, Fax. +34 3 301 4107

Sweden: Kottbygatan 7, Akalla, S-16485 STOCKHOLM,
Tel. +46 8 632 2000, Fax. +46 8 632 2745

Switzerland: Allmendstrasse 140, CH-8027 ZÜRICH,
Tel. +41 1 488 2686, Fax. +41 1 481 7730

Taiwan: Philips Semiconductors, 6F, No. 96, Chien Kuo N. Rd., Sec. 1,
TAIPEI, Taiwan Tel. +886 2 2134 2865, Fax. +886 2 2134 2874

Thailand: PHILIPS ELECTRONICS (THAILAND) Ltd.,
209/2 Sanpavuth-Bangna Road Prakanong, BANGKOK 10260,
Tel. +66 2 745 4090, Fax. +66 2 398 0793

Turkey: Talatpasa Cad. No. 5, 80640 GÜLTEPE/ISTANBUL,
Tel. +90 212 279 2770, Fax. +90 212 282 6707

Ukraine: PHILIPS UKRAINE, 4 Patrice Lumumba str., Building B, Floor 7,
252042 KIEV, Tel. +380 44 264 2776, Fax. +380 44 268 0461

United Kingdom: Philips Semiconductors Ltd., 276 Bath Road, Hayes,
MIDDLESEX UB3 5BX, Tel. +44 181 730 5000, Fax. +44 181 754 8421

United States: 811 East Arques Avenue, SUNNYVALE, CA 94088-3409,
Tel. +1 800 234 7381

Uruguay: see South America

Vietnam: see Singapore

Yugoslavia: PHILIPS, Trg N. Pasica 5/v, 11000 BEOGRAD,
Tel. +381 11 625 344, Fax.+381 11 635 777

For all other countries apply to: Philips Semiconductors, Marketing & Sales Communications,
Building BE-p, P.O. Box 218, 5600 MD EINDHOVEN, The Netherlands, Fax. +31 40 27 24825

Internet: http://www.semiconductors.philips.com

SCA54

Printed in The Netherlands 457047/00/01/pp92 Date of release: 1997 Jul 02 Document order number: 9397 750 02551

Let's make things better.

Philips Semiconductors

PHILIPS