Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

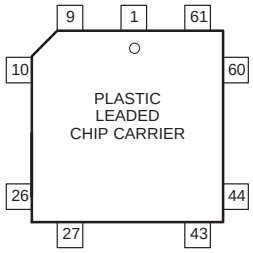
Details	
Product Status	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	16MHz
Connectivity	EBI/EMI, I²C, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	40
Program Memory Size	16KB (16K x 8)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	68-LCC (J-Lead)
Supplier Device Package	68-PLCC (24.18x24.18)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/p87c554sfaa-512

80C51 8-bit microcontroller – 12 clock operation
16K/512 OTP/RAM, 8 channel 10-bit A/D, I^2C, PWM, capture/compare, high I/O

P87C554

PINNING INFORMATION

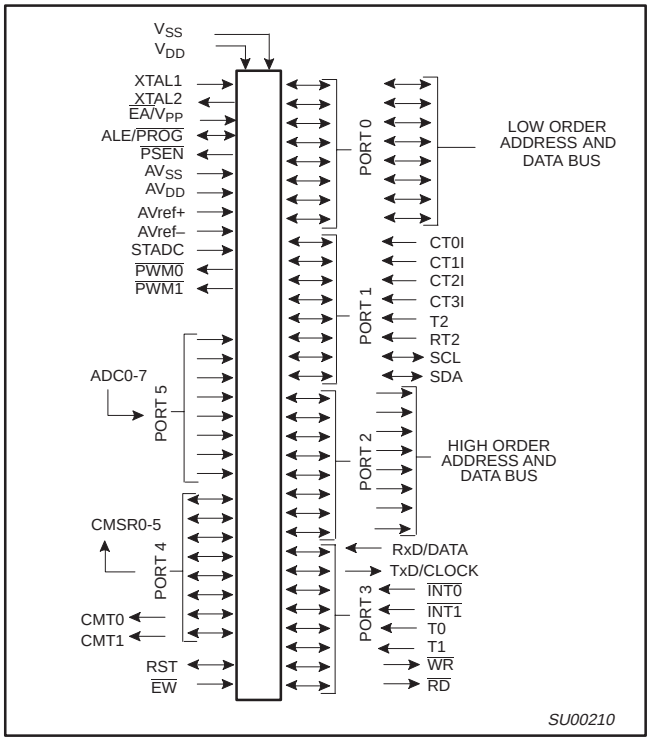
Plastic Leaded Chip Carrier pin functions



Pin	Function	Pin	Function	Pin	Function
1	P5.0/ADC0	24	P3.0/RxD	47	$\overline{PSEN}$
2	V_{DD}	25	P3.1/TxD	48	ALE/$\overline{PROG}$
3	STADC	26	P3.2/$\overline{INT0}$	49	$\overline{EA}$/V_{PP}
4	$\overline{PWM0}$	27	P3.3/$\overline{INT1}$	50	P0.7/AD7
5	$\overline{PWM1}$	28	P3.4/T0	51	P0.6/AD6
6	$\overline{EW}$	29	P3.5/T1	52	P0.5/AD5
7	P4.0/CMSR0	30	P3.6/$\overline{WR}$	53	P0.4/AD4
8	P4.1/CMSR1	31	P3.7/$\overline{RD}$	54	P0.3/AD3
9	P4.2/CMSR2	32	NC	55	P0.2/AD2
10	P4.3/CMSR3	33	NC	56	P0.1/AD1
11	P4.4/CMSR4	34	XTAL2	57	P0.0/AD0
12	P4.5/CMSR5	35	XTAL1	58	AVref–
13	P4.6/CMT0	36	V_{SS}	59	AVref+
14	P4.7/CMT1	37	V_{SS}	60	AV_{SS}
15	RST	38	NC	61	AV_{DD}
16	P1.0/CT0I	39	P2.0/A08	62	P5.7/ADC7
17	P1.1/CT1I	40	P2.1/A09	63	P5.6/ADC6
18	P1.2/CT2I	41	P2.2/A10	64	P5.5/ADC5
19	P1.3/CT3I	42	P2.3/A11	65	P5.4/ADC4
20	P1.4/T2	43	P2.4/A12	66	P5.3/ADC3
21	P1.5/RT2	44	P2.5/A13	67	P5.2/ADC2
22	P1.6/SCL	45	P2.6/A14	68	P5.1/ADC1
23	P1.7/SDA	46	P2.7/A15		

SU00208

LOGIC SYMBOL



SU00210

Table 2. External Pin Status During Idle and Power-Down Modes

MODE	PROGRAM MEMORY	ALE	$\overline{PSEN}$	PORT 0	PORT 1	PORT 2	PORT 3	PORT 4	PWM0/ PWM1
Idle	Internal	1	1	Data	Data	Data	Data	Data	High
Idle	External	1	1	Float	Data	Address	Data	Data	High
Power-down	Internal	0	0	Data	Data	Data	Data	Data	High
Power-down	External	0	0	Float	Data	Data	Data	Data	High

With an external interrupt, INT0 and INT1 must be enabled and configured as level-sensitive. Holding the pin low restarts the oscillator but bringing the pin back high completes the exit. Once the interrupt is serviced, the next instruction to be executed after RETI will be the one following the instruction that put the device into Power Down.

POWER OFF FLAG

The Power Off Flag (POF) is set by on-chip circuitry when the V_{CC} level on the P87C554 rises from 0 to 5 V. The POF bit can be set or cleared by software allowing a user to determine if the reset is the result of a power-on or a warm start after powerdown. The V_{CC} level must remain above 3 V for the POF to remain unaffected by the V_{CC} level.

Design Consideration

- When the idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

ONCE™ Mode

The ONCE ("On-Circuit Emulation") Mode facilitates testing and debugging of systems without the device having to be removed from the circuit. The ONCE Mode is invoked by:

1. Pull ALE low while the device is in reset and $\overline{PSEN}$ is high;
2. Hold ALE low as RST is deactivated.

While the device is in ONCE Mode, the Port 0 pins go into a float state, and the other port pins and ALE and $\overline{PSEN}$ are weakly pulled high. The oscillator circuit remains active. While the device is in this mode, an emulator or test CPU can be used to drive the circuit. Normal operation is restored when a normal reset is applied.

Reduced EMI Mode

The ALE-Off bit, AO (AUXR.0) can be set to disable the ALE output. It will automatically become active when required for external memory accesses and resume to the OFF state after completing the external memory access.

	7	6	5	4	3	2	1	0
PCON (87H)	SMOD1	SMOD0	POF	WLE	GF1	GF0	PD	IDL
	(MSB)							(LSB)

BIT	SYMBOL	FUNCTION
PCON.7	SMOD1	Double Baud rate bit. When set to logic 1, the baud rate is doubled when the serial port SIO0 is being used in modes 1, 2, or 3.
PCON.6	SMOD0	Selects SM0/FE for SCON.7 bit.
PCON.5	POF	Power Off Flag
PCON.4	WLE	Watchdog Load Enable. This flag must be set by software prior to loading timer T3 (watchdog timer). It is cleared when timer T3 is loaded.
PCON.3	GF1	General-purpose flag bit.
PCON.2	GF0	General-purpose flag bit.
PCON.1	PD	Power-down bit. Setting this bit activates the power-down mode. It can only be set if input $\overline{EW}$ is high.
PCON.0	IDL	Idle mode bit. Setting this bit activates the Idle mode.

If logic 1s are written to PD and IDL at the same time, PD takes precedence. The reset value of PCON is (00X00000).

SU00954

Figure 3. Power Control Register (PCON)

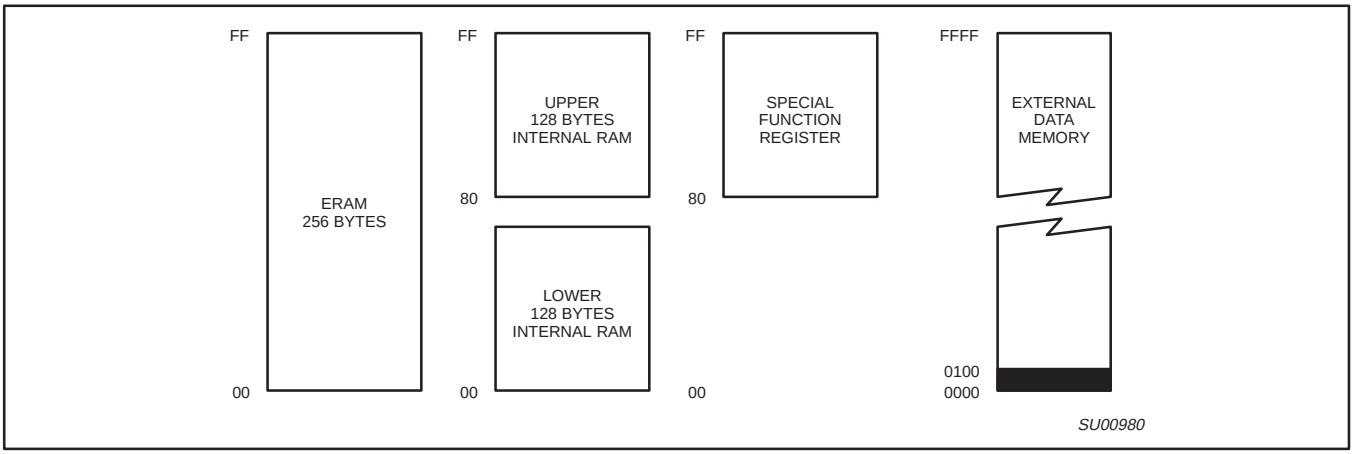


Figure 5. Internal and External Data Memory Address Space with EXTRAM = 0

Dual DPTR

The dual DPTR structure (see Figure 6) is a way by which the chip will specify the address of an external data memory location. There are two 16-bit DPTR registers that address the external memory, and a single bit called DPS = AUXR1/bit0 that allows the program code to switch between them.

The DPS bit status should be saved by software when switching between DPTR0 and DPTR1.

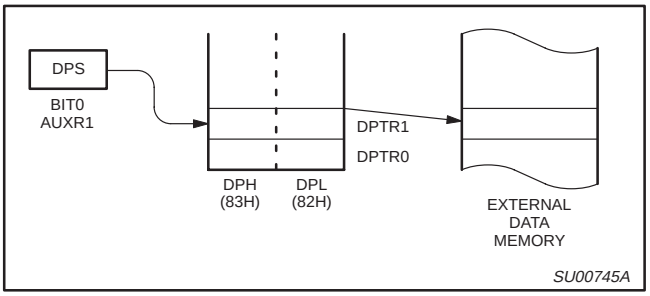


Figure 6.

Note that bit 2 is not writable and is always read as a zero. This allows the DPS bit to be quickly toggled simply by executing an INC AUXR1 instruction without affecting the other bits.

DPTR Instructions

The instructions that refer to DPTR refer to the data pointer that is currently selected using the AUXR1/bit 0 register. The six instructions that use the DPTR are as follows:

INC DPTR	Increments the data pointer by 1
MOV DPTR, #data16	Loads the DPTR with a 16-bit constant
MOV A, @ A+DPTR	Move code byte relative to DPTR to ACC
MOVX A, @ DPTR	Move external RAM (16-bit address) to ACC
MOVX @ DPTR , A	Move ACC to external RAM (16-bit address)
JMP @ A + DPTR	Jump indirect relative to DPTR

The data pointer can be accessed on a byte-by-byte basis by specifying the low or high byte in an instruction which accesses the SFRs. See application note AN458 for more details.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

Slave 0	SADDR	=	1100 0000
	SADEN	=	1111 1001
	Given	=	1100 0XX0
Slave 1	SADDR	=	1110 0000
	SADEN	=	1111 1010
	Given	=	1110 0X0X
Slave 2	SADDR	=	1110 0000
	SADEN	=	1111 1100
	Given	=	1110 00XX

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary to make bit 2 = 1 to exclude slave 2.

The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are trended as don't-cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FF hexadecimal.

Upon reset SADDR (SFR address 0A9H) and SADEN (SFR address 0B9H) are leaded with 0s. This produces a given address of all "don't cares" as well as a Broadcast address of all "don't cares". This effectively disables the Automatic Addressing mode and allows the microcontroller to use standard 80C51 type UART drivers which do not make use of this feature.

Timer T2

Timer T2 is a 16-bit timer consisting of two registers TMH2 (HIGH byte) and TML2 (LOW byte). The 16-bit timer/counter can be switched off or clocked via a prescaler from one of two sources: $f_{OSC}/12$ or an external signal. When Timer T2 is configured as a counter, the prescaler is clocked by an external signal on T2 (P1.4). A rising edge on T2 increments the prescaler, and the maximum repetition rate is one count per machine cycle (1 MHz with a 12 MHz oscillator).

The maximum repetition rate for Timer T2 is twice the maximum repetition rate for Timer 0 and Timer 1. T2 (P1.4) is sampled at S2P1 and again at S5P1 (i.e., twice per machine cycle). A rising edge is detected when T2 is LOW during one sample and HIGH during the next sample. To ensure that a rising edge is detected, the input signal must be LOW for at least 1/2 cycle and then HIGH for at least 1/2 cycle. If a rising edge is detected before the end of S2P1, the timer will be incremented during the following cycle; otherwise it will be incremented one cycle later. The prescaler has a programmable division factor of 1, 2, 4, or 8 and is cleared if its division factor or input source is changed, or if the timer/counter is reset.

Timer T2 may be read "on the fly" but possesses no extra read latches, and software precautions may have to be taken to avoid misinterpretation in the event of an overflow from least to most significant byte while Timer T2 is being read. Timer T2 is not loadable and is reset by the RST signal or by a rising edge on the input signal RT2, if enabled. RT2 is enabled by setting bit T2ER (TM2CON.5).

When the least significant byte of the timer overflows or when a 16-bit overflow occurs, an interrupt request may be generated.

Either or both of these overflows can be programmed to request an interrupt. In both cases, the interrupt vector will be the same. When the lower byte (TML2) overflows, flag T2B0 (TM2CON) is set and flag T20V (TM2IR) is set when TMH2 overflows. These flags are set one cycle after an overflow occurs. Note that when T20V is set, T2B0 will also be set. To enable the byte overflow interrupt, bits ET2 (IEN1.7, enable overflow interrupt, see Figure 11) and T2IS0 (TM2CON.6, byte overflow interrupt select) must be set. Bit TWB0 (TM2CON.4) is the Timer T2 byte overflow flag.

To enable the 16-bit overflow interrupt, bits ET2 (IE1.7, enable overflow interrupt) and T2IS1 (TM2CON.7, 16-bit overflow interrupt select) must be set. Bit T2OV (TM2IR.7) is the Timer T2 16-bit overflow flag. All interrupt flags must be reset by software. To enable both byte and 16-bit overflow, T2IS0 and T2IS1 must be set and two interrupt service routines are required. A test on the overflow flags indicates which routine must be executed. For each routine, only the corresponding overflow flag must be cleared.

Timer T2 may be reset by a rising edge on RT2 (P1.5) if the Timer T2 external reset enable bit (T2ER) in T2CON is set. This reset also clears the prescaler. In the idle mode, the timer/counter and prescaler are reset and halted. Timer T2 is controlled by the TM2CON special function register (see Figure 12).

Timer T2 Extension: When a 12 MHz oscillator is used, a 16-bit overflow on Timer T2 occurs every 65.5, 131, 262, or 524 ms, depending on the prescaler division ratio; i.e., the maximum cycle time is approximately 0.5 seconds. In applications where cycle times are greater than 0.5 seconds, it is necessary to extend Timer T2. This is achieved by selecting fosc/12 as the clock source (set T2MS0, reset T2MS1), setting the prescaler division ration to 1/8 (set T2P0, set T2P1), disabling the byte overflow interrupt (reset T2IS0) and enabling the 16-bit overflow interrupt (set T2IS1). The following software routine is written for a three-byte extension which gives a maximum cycle time of approximately 2400 hours.

```
OVINT: PUSH   ACC      ;save accumulator
       PUSH   PSW      ;save status
       INC    TIMEX1   ;increment first byte (low order)
                       ;of extended timer
       MOV    A,TIMEX1
       JNZ    INTEX    ;jump to INTEX if ;there is no overflow

       INC    TIMEX2   ;increment second byte
       MOV    A,TIMEX2
       JNZ    INTEX    ;jump to INTEX if there is no overflow
       INC    TIMEX3   ;increment third byte (high order)

INTEX: CLR    T2OV     ;reset interrupt flag
       POP    PSW      ;restore status
       POP    ACC      ;restore accumulator
       RETI            ;return from interrupt
```

Timer T2, Capture and Compare Logic: Timer T2 is connected to four 16-bit capture registers and three 16-bit compare registers. A capture register may be used to capture the contents of Timer T2 when a transition occurs on its corresponding input pin. A compare register may be used to set, reset, or toggle port 4 output pins at certain pre-programmable time intervals.

The combination of Timer T2 and the capture and compare logic is very powerful in applications involving rotating machinery, automotive injection systems, etc. Timer T2 and the capture and compare logic are shown in Figure 13.

Reset Value = xx00 0000B

	7	6	5	4	3	2	1	0
ADCON (C5H)	ADC.1	ADC.0	ADEX	ADCI	ADCS	AADR2	AADR1	AADR0
	(MSB)							(LSB)

Bit	Symbol	Function
ADCON.7	ADC.1	Bit 1 of ADC result
ADCON.6	ADC.0	Bit 0 of ADC result
ADCON.5	ADEX	Enable external start of conversion by STADC 0 = Conversion can be started by software only (by setting ADCS) 1 = Conversion can be started by software or externally (by a rising edge on STADC)
ADCON.4	ADCI	ADC interrupt flag: this flag is set when an A/D conversion result is ready to be read. An interrupt is invoked if it is enabled. The flag may be cleared by the interrupt service routine. While this flag is set, the ADC cannot start a new conversion. ADCI cannot be set by software.
ADCON.3	ADCS	ADC start and status: setting this bit starts an A/D conversion. It may be set by software or by the external signal STADC. The ADC logic ensures that this signal is HIGH while the ADC is busy. On completion of the conversion, ADCS is reset immediately after the interrupt flag has been set. ADCS cannot be reset by software. A new conversion may not be started while either ADCS or ADCI is high.

ADCI	ADCS	ADC Status
0	0	ADC not busy; a conversion can be started
0	1	ADC busy; start of a new conversion is blocked
1	0	Conversion completed; start of a new conversion requires ADCI=0
1	1	Conversion completed; start of a new conversion requires ADCI=0

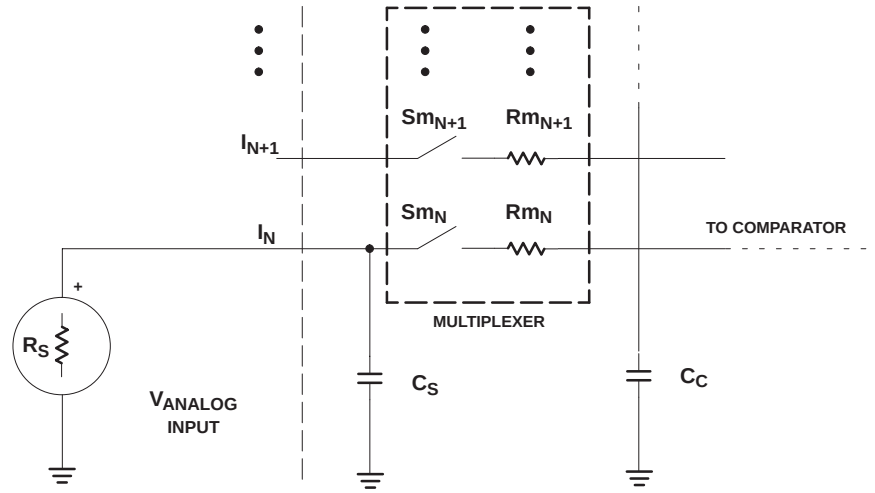
If ADCI is cleared by software while ADCS is set at the same time, a new A/D conversion with the same channel number may be started.
But it is recommended to reset ADCI ***before*** ADCS is set.

ADCON.2	AADR2	Analogue input select: this binary coded address selects one of the eight analogue port bits of P5 to be input to the converter. It can only be changed when ADCI and ADCS are both LOW.
ADCON.1	AADR1	
ADCON.0	AADR0	

AADR2	AADR1	AADR0	Selected Analog Channel
0	0	0	ADC0 (P5.0)
0	0	1	ADC1 (P5.1)
0	1	0	ADC2 (P5.2)
0	1	1	ADC3 (P5.3)
1	0	0	ADC4 (P5.4)
1	0	1	ADC5 (P5.5)
1	1	0	ADC6 (P5.6)
1	1	1	ADC7 (P5.7)

SU00960

Figure 23. ADC Control Register (ADCON)

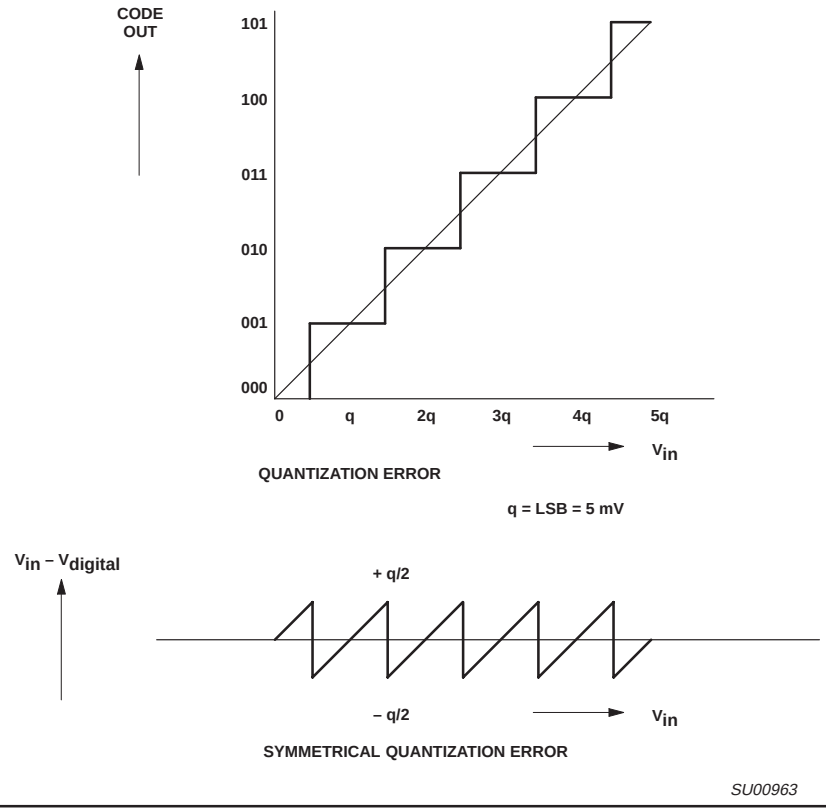


Rm = 0.5 - 3 kΩ
$C_S + C_C$ = 15 pF maximum
R_S = Recommended < 9.6 kΩ for 1 LSB @ 12 MHz

NOTE:
Because the analog to digital converter has a sampled-data comparator, the input looks capacitive to a source. When a conversion is initiated, switch Sm closes for $8t_{CY}$ (8 µs @ 12 MHz crystal frequency) during which time capacitance $C_S + C_C$ is charged. It should be noted that the sampling causes the analog input to present a varying load to an analog source.

SU00962

Figure 25. A/D Input: Equivalent Circuit



SU00963

Figure 26. Effective Conversion Characteristic

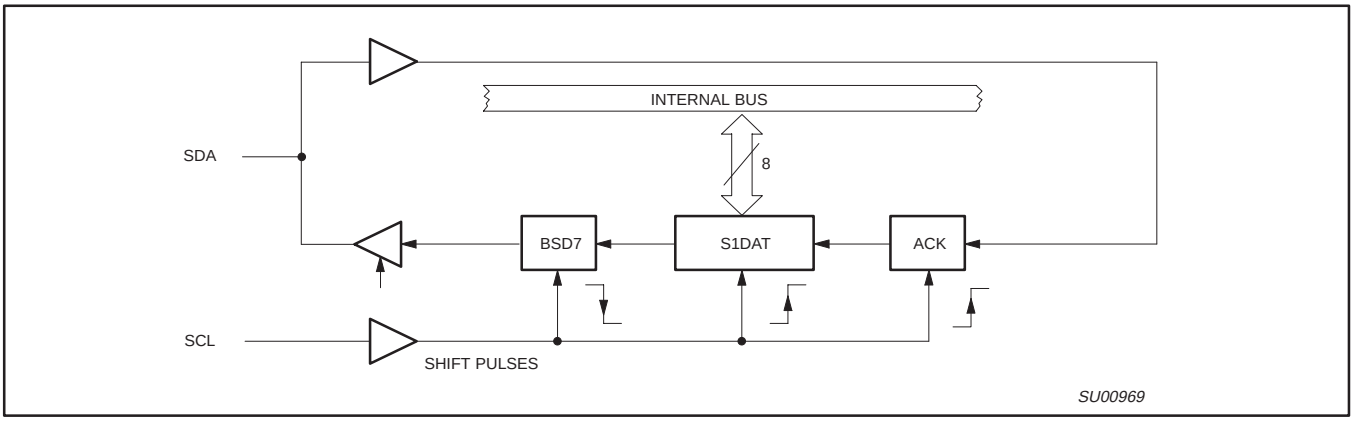


Figure 38. Serial Input/Output Configuration

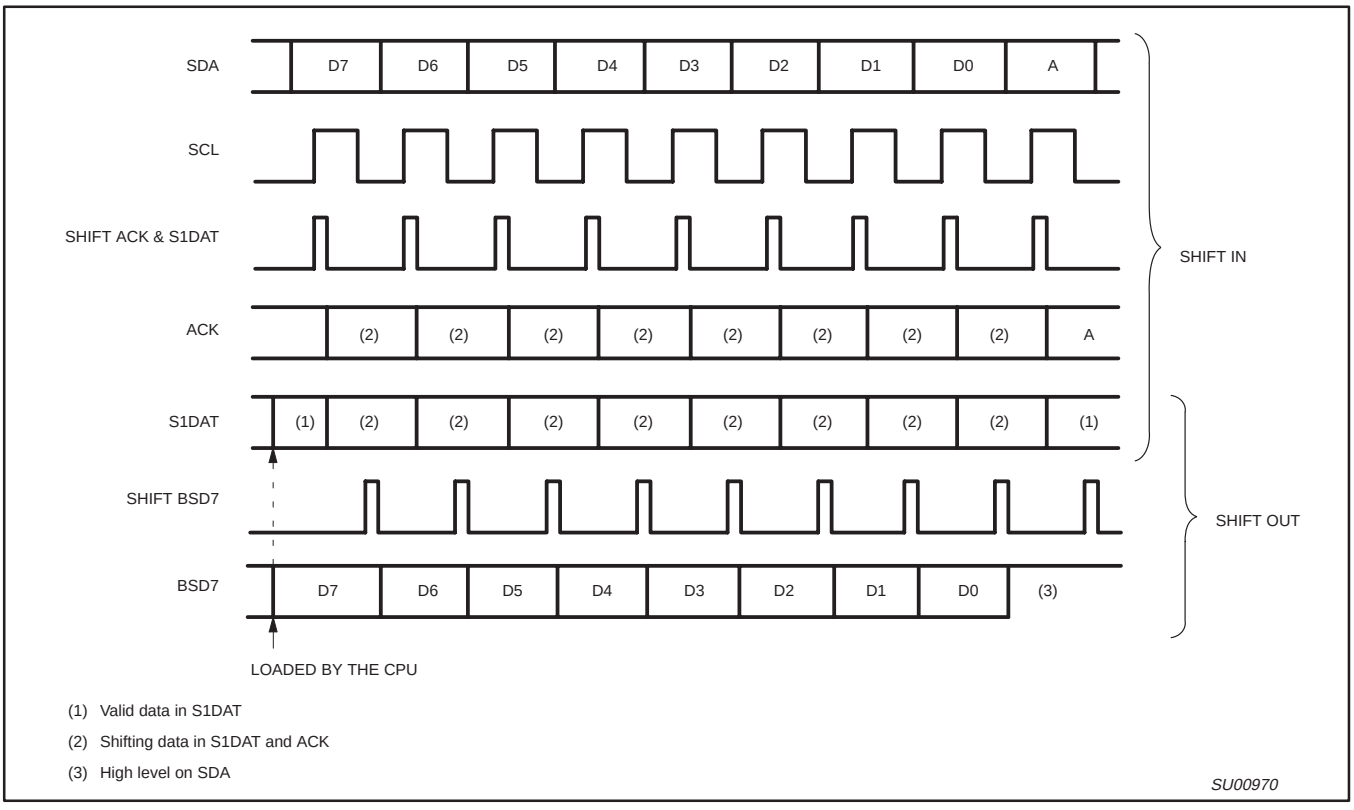


Figure 39. Shift-in and Shift-out Timing

In the following text, it is assumed that ENS1 = “1”.

STA, THE START FLAG
STA = “1”: When the STA bit is set to enter a master mode, the SIO1 hardware checks the status of the I2C bus and generates a START condition if the bus is free. If the bus is not free, then SIO1 waits for a STOP condition (which will free the bus) and generates a START condition after a delay of a half clock period of the internal serial clock generator.

If STA is set while SIO1 is already in a master mode and one or more bytes are transmitted or received, SIO1 transmits a repeated START condition. STA may be set at any time. STA may also be set when SIO1 is an addressed slave.

STA = “0”: When the STA bit is reset, no START condition or repeated START condition will be generated.

STO, THE STOP FLAG
STO = “1”: When the STO bit is set while SIO1 is in a master mode, a STOP condition is transmitted to the I2C bus. When the STOP condition is detected on the bus, the SIO1 hardware clears the STO flag. In a slave mode, the STO flag may be set to recover from an error condition. In this case, no STOP condition is transmitted to the I2C bus. However, the SIO1 hardware behaves as if a STOP condition has been received and switches to the defined “not addressed” slave receiver mode. The STO flag is automatically cleared by hardware.

More Information on SIO1 Operating Modes: The four operating modes are:

– Master Transmitter
– Master Receiver
– Slave Receiver
– Slave Transmitter

Data transfers in each mode of operation are shown in Figures 40–43. These figures contain the following abbreviations:

Abbreviation	Explanation
S	Start condition
SLA	7-bit slave address
R	Read bit (high level at SDA)
W	Write bit (low level at SDA)
A	Acknowledge bit (low level at SDA)
$\overline{A}$	Not acknowledge bit (high level at SDA)
Data	8-bit data byte
P	Stop condition

In Figures 40-43, circles are used to indicate when the serial interrupt flag is set. The numbers in the circles show the status code held in the S1STA register. At these points, a service routine must be executed to continue or complete the serial transfer. These service routines are not critical since the serial transfer is suspended until the serial interrupt flag is cleared by software.

When a serial interrupt routine is entered, the status code in S1STA is used to branch to the appropriate service routine. For each status code, the required software action and details of the following serial transfer are given in Tables 6-10.

Master Transmitter Mode: In the master transmitter mode, a number of data bytes are transmitted to a slave receiver (see Figure 40). Before the master transmitter mode can be entered, S1CON must be initialized as follows:

	7	6	5	4	3	2	1	0
S1CON (D8H)	CR2	ENS1	STA	STO	SI	AA	CR1	CR0
	bit rate	1	0	0	0	X	bit rate	

CR0, CR1, and CR2 define the serial bit rate. ENS1 must be set to logic 1 to enable SIO1. If the AA bit is reset, SIO1 will not acknowledge its own slave address or the general call address in the event of another device becoming master of the bus. In other words, if AA is reset, SIO0 cannot enter a slave mode. STA, STO, and SI must be reset.

The master transmitter mode may now be entered by setting the STA bit using the SETB instruction. The SIO1 logic will now test the I²C bus and generate a start condition as soon as the bus becomes free. When a START condition is transmitted, the serial interrupt flag (SI) is set, and the status code in the status register (S1STA) will be 08H. This status code must be used to vector to an interrupt service routine that loads S1DAT with the slave address and the data direction bit (SLA+W). The SI bit in S1CON must then be reset before the serial transfer can continue.

When the slave address and the direction bit have been transmitted and an acknowledgment bit has been received, the serial interrupt flag (SI) is set again, and a number of status codes in S1STA are possible. There are 18H, 20H, or 38H for the master mode and also 68H, 78H, or B0H if the slave mode was enabled (AA = logic 1). The appropriate action to be taken for each of these status codes is detailed in Table 6. After a repeated start condition (state 10H). SIO1 may switch to the master receiver mode by loading S1DAT with SLA+R).

Master Receiver Mode: In the master receiver mode, a number of data bytes are received from a slave transmitter (see Figure 41). The transfer is initialized as in the master transmitter mode. When the start condition has been transmitted, the interrupt service routine must load S1DAT with the 7-bit slave address and the data direction bit (SLA+R). The SI bit in S1CON must then be cleared before the serial transfer can continue.

When the slave address and the data direction bit have been transmitted and an acknowledgment bit has been received, the serial interrupt flag (SI) is set again, and a number of status codes in S1STA are possible. These are 40H, 48H, or 38H for the master mode and also 68H, 78H, or B0H if the slave mode was enabled (AA = logic 1). The appropriate action to be taken for each of these status codes is detailed in Table 7. ENS1, CR1, and CR0 are not affected by the serial transfer and are not referred to in Table 7. After a repeated start condition (state 10H), SIO1 may switch to the master transmitter mode by loading S1DAT with SLA+W.

Slave Receiver Mode: In the slave receiver mode, a number of data bytes are received from a master transmitter (see Figure 42). To initiate the slave receiver mode, S1ADR and S1CON must be loaded as follows:

	7	6	5	4	3	2	1	0
S1ADR (DBH)	X	X	X	X	X	X	X	GC
	own slave address							

The upper 7 bits are the address to which SIO1 will respond when addressed by a master. If the LSB (GC) is set, SIO1 will respond to the general call address (00H); otherwise it ignores the general call address.

	7	6	5	4	3	2	1	0
S1CON (D8H)	CR2	ENS1	STA	STO	SI	AA	CR1	CR0
	X	1	0	0	0	1	X	X

CR0, CR1, and CR2 do not affect SIO1 in the slave mode. ENS1 must be set to logic 1 to enable SIO1. The AA bit must be set to enable SIO1 to acknowledge its own slave address or the general call address. STA, STO, and SI must be reset.

When S1ADR and S1CON have been initialized, SIO1 waits until it is addressed by its own slave address followed by the data direction bit which must be “0” (W) for SIO1 to operate in the slave receiver mode. After its own slave address and the W bit have been received, the serial interrupt flag (I) is set and a valid status code can be read from S1STA. This status code is used to vector to an interrupt service routine, and the appropriate action to be taken for each of these status codes is detailed in Table 8. The slave receiver mode may also be entered if arbitration is lost while SIO1 is in the master mode (see status 68H and 78H).

If the AA bit is reset during a transfer, SIO1 will return a not acknowledge (logic 1) to SDA after the next received data byte. While AA is reset, SIO1 does not respond to its own slave address or a general call address. However, the I²C bus is still monitored and address recognition may be resumed at any time by setting AA. This means that the AA bit may be used to temporarily isolate SIO1 from the I²C bus.

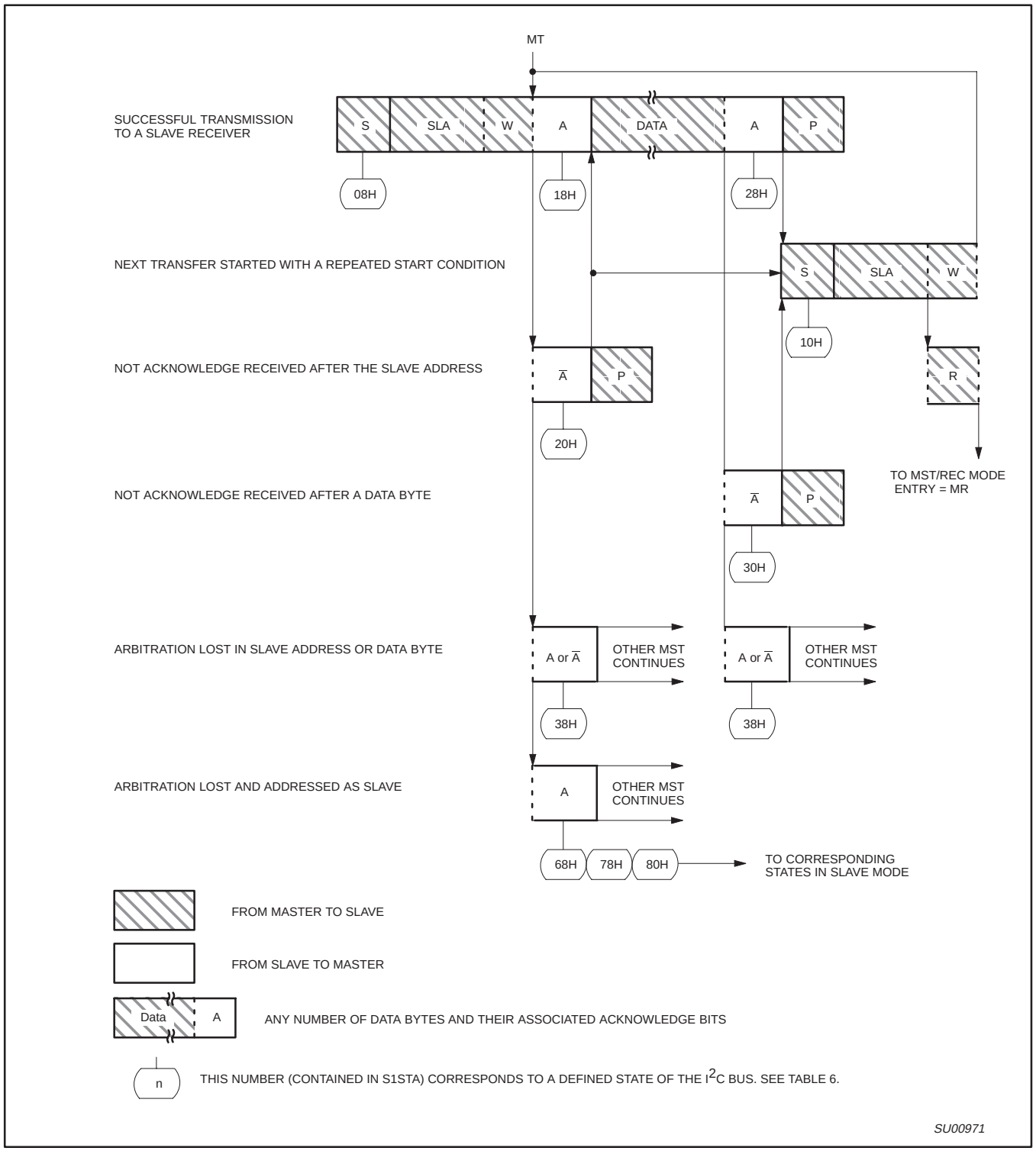


Figure 40. Format and States in the Master Transmitter Mode

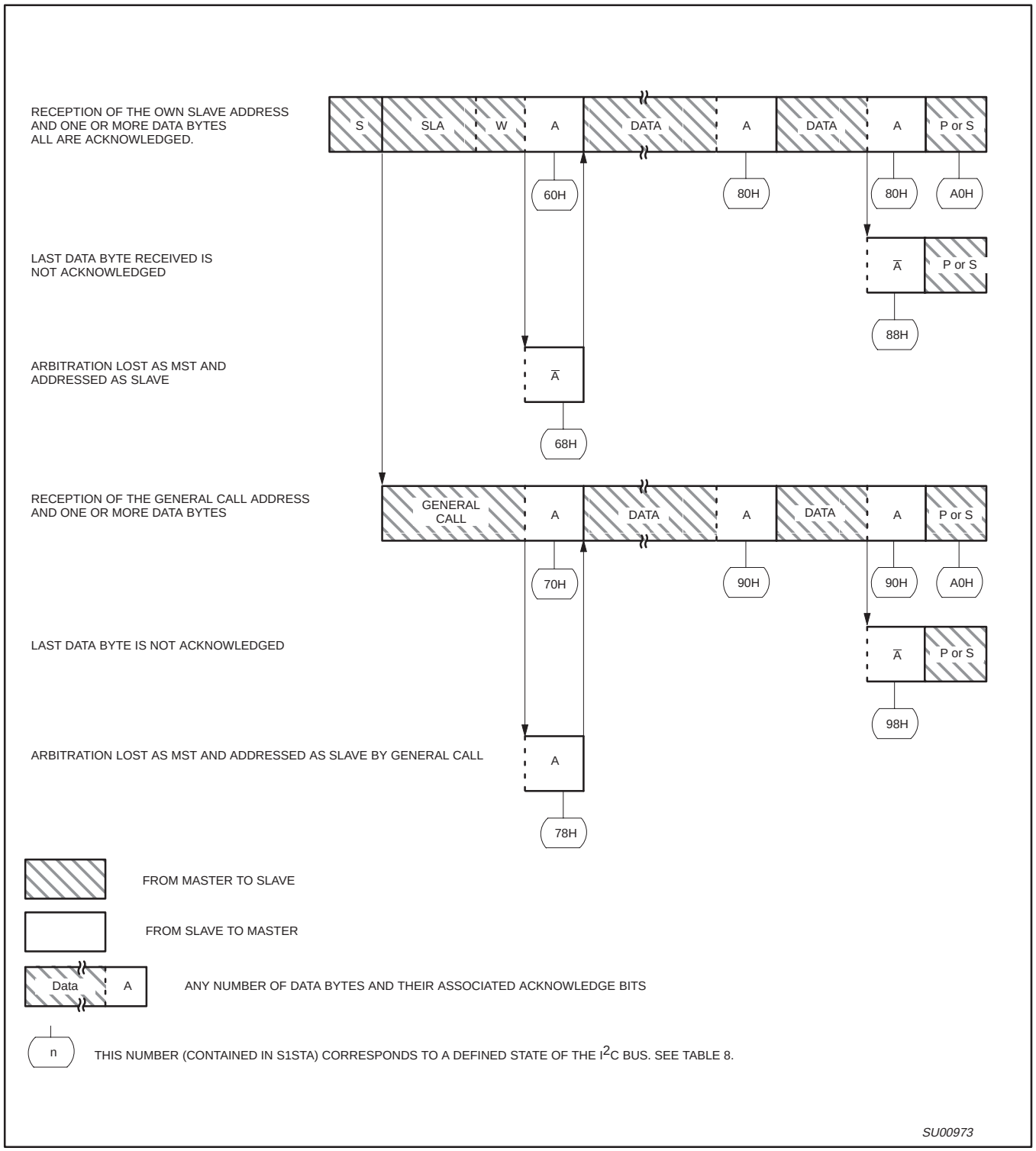


Figure 42. Format and States in the Slave Receiver Mode

Table 8. Slave Receiver Mode

STATUS CODE (S1STA)	STATUS OF THE I²C BUS AND SIO1 HARDWARE	APPLICATION SOFTWARE RESPONSE					NEXT ACTION TAKEN BY SIO1 HARDWARE
		TO/FROM S1DAT	TO S1CON				
			STA	STO	SI	AA	
60H	Own SLA+W has been received; ACK has been returned	No S1DAT action or	X	0	0	0	Data byte will be received and NOT ACK will be returned
		no S1DAT action	X	0	0	1	Data byte will be received and ACK will be returned
68H	Arbitration lost in SLA+R/$\overline{W}$ as master; Own SLA+W has been received, ACK returned	No S1DAT action or	X	0	0	0	Data byte will be received and NOT ACK will be returned
		no S1DAT action	X	0	0	1	Data byte will be received and ACK will be returned
70H	General call address (00H) has been received; ACK has been returned	No S1DAT action or	X	0	0	0	Data byte will be received and NOT ACK will be returned
		no S1DAT action	X	0	0	1	Data byte will be received and ACK will be returned
78H	Arbitration lost in SLA+R/$\overline{W}$ as master; General call address has been received, ACK has been returned	No S1DAT action or	X	0	0	0	Data byte will be received and NOT ACK will be returned
		no S1DAT action	X	0	0	1	Data byte will be received and ACK will be returned
80H	Previously addressed with own SLV address; DATA has been received; ACK has been returned	Read data byte or	X	0	0	0	Data byte will be received and NOT ACK will be returned
		read data byte	X	0	0	1	Data byte will be received and ACK will be returned
88H	Previously addressed with own SLA; DATA byte has been received; NOT ACK has been returned	Read data byte or	0	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or General call address
		read data byte or	0	0	0	1	Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1
		read data byte or	1	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or General call address. A START condition will be transmitted when the bus becomes free
		read data byte	1	0	0	1	Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1. A START condition will be transmitted when the bus becomes free.
90H	Previously addressed with General Call; DATA byte has been received; ACK has been returned	Read data byte or	X	0	0	0	Data byte will be received and NOT ACK will be returned
		read data byte	X	0	0	1	Data byte will be received and ACK will be returned
98H	Previously addressed with General Call; DATA byte has been received; NOT ACK has been returned	Read data byte or	0	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or General call address
		read data byte or	0	0	0	1	Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1
		read data byte or	1	0	0	0	Switched to not addressed SLV mode; no recognition of own SLA or General call address. A START condition will be transmitted when the bus becomes free
		read data byte	1	0	0	1	Switched to not addressed SLV mode; Own SLA will be recognized; General call address will be recognized if S1ADR.0 = logic 1. A START condition will be transmitted when the bus becomes free.

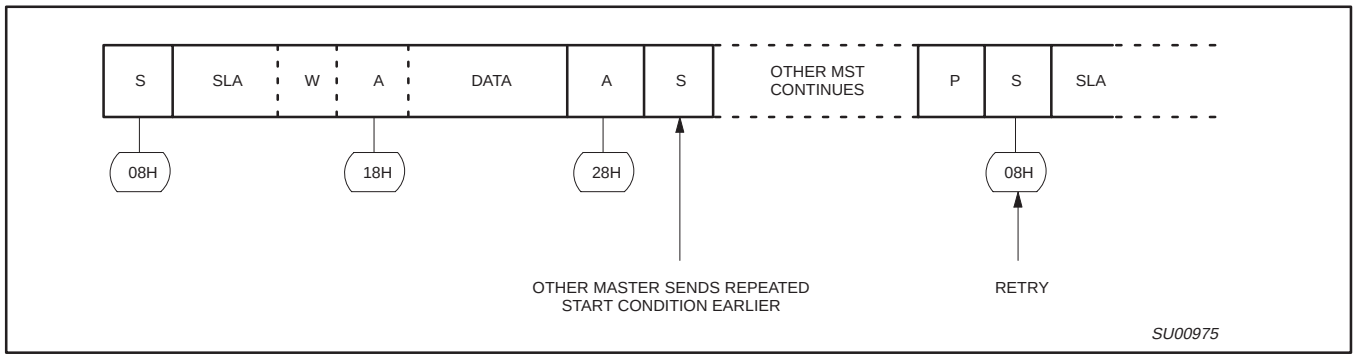


Figure 44. Simultaneous Repeated START Conditions from 2 Masters

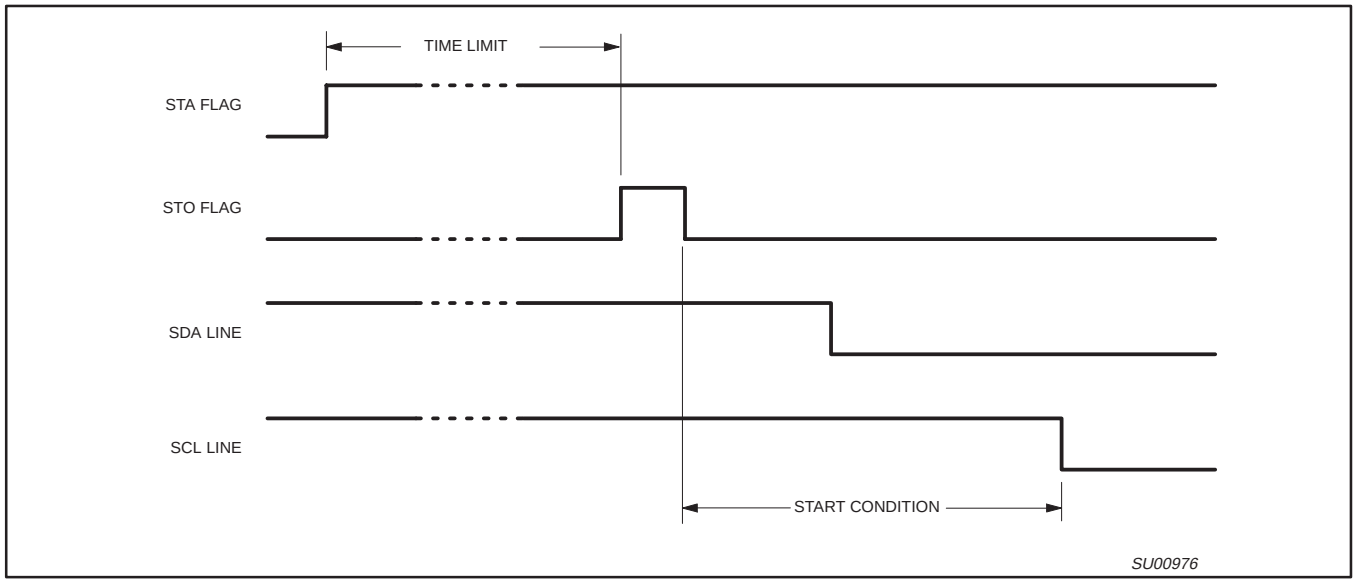


Figure 45. Forced Access to a Busy I2C Bus

I2C BUS OBSTRUCTED BY A LOW LEVEL ON SCL OR SDA

An I2C bus hang-up occurs if SDA or SCL is pulled LOW by an uncontrolled source. If the SCL line is obstructed (pulled LOW) by a device on the bus, no further serial transfer is possible, and the SIO1 hardware cannot resolve this type of problem. When this occurs, the problem must be resolved by the device that is pulling the SCL bus line LOW.

If the SDA line is obstructed by another device on the bus (e.g., a slave device out of bit synchronization), the problem can be solved by transmitting additional clock pulses on the SCL line (see Figure 46). The SIO1 hardware transmits additional clock pulses when the STA flag is set, but no START condition can be generated because the SDA line is pulled LOW while the I2C bus is considered free. The SIO1 hardware attempts to generate a START condition after every two additional clock pulses on the SCL line. When the SDA line is eventually released, a normal START condition is transmitted, state 08H is entered, and the serial transfer continues.

If a forced bus access occurs or a repeated START condition is transmitted while SDA is obstructed (pulled LOW), the SIO1 hardware performs the same action as described above. In each case, state 08H is entered after a successful START condition is transmitted and normal serial transfer continues. Note that the CPU is not involved in solving these bus hang-up problems.

BUS ERROR

A bus error occurs when a START or STOP condition is present at an illegal position in the format frame. Examples of illegal positions are during the serial transfer of an address byte, a data or an acknowledge bit.

The SIO1 hardware only reacts to a bus error when it is involved in a serial transfer either as a master or an addressed slave. When a bus error is detected, SIO1 immediately switches to the not addressed slave mode, releases the SDA and SCL lines, sets the interrupt flag, and loads the status register with 00H. This status code may be used to vector to a service routine which either attempts the aborted serial transfer again or simply recovers from the error condition as shown in Table 10.

```
                        !*************************************************************************************************
                        ! SI01 EQUATE LIST
                        !*************************************************************************************************
                        !*************************************************************************************************
                        ! LOCATIONS OF THE SI01 SPECIAL FUNCTION REGISTERS
                        !*************************************************************************************************
00D8                    S1CON     –0xd8
00D9                    S1STA     –0xd9
00DA                    S1DAT     –0xda
00DB                    S1ADR     –0xdb

00A8                    IEN0      –0xa8
00B8                    IP0       –02b8

                        !*************************************************************************************************
                        ! BIT LOCATIONS
                        !*************************************************************************************************

00DD                    STA       –0xdd                    ! STA bit in S1CON
00BD                    SI01HP    –0xbd                    ! IP0, SI01 Priority bit

                        !*************************************************************************************************
                        ! IMMEDIATE DATA TO WRITE INTO REGISTER S1CON
                        !*************************************************************************************************
00D5                    ENS1_NOTSTA_STO_NOTSI_AA_CR0       –0xd5    ! Generates STOP
                                                                    ! (CR0 = 100kHz)
00C5                    ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0    –0xc5    ! Releases BUS and
                                                                    ! ACK
00C1                    ENS1_NOTSTA_NOTSTO_NOTSI_NOTAA_CR0 –0xc1    ! Releases BUS and
                                                                    ! NOT ACK
00E5                    ENS1_STA_NOTSTO_NOTSI_AA_CR0       –0xe5    ! Releases BUS and
                                                                    ! set STA

                        !*************************************************************************************************
                        ! GENERAL IMMEDIATE DATA
                        !*************************************************************************************************
0031                    OWNSLA    –0x31                    ! Own SLA+General Call
                                                           ! must be written into S1ADR
00A0                    ENSI01    –0xa0                    ! EA+ES1, enable SIO1 interrupt
                                                           ! must be written into IEN0
0001                    PAG1      –0x01                    ! select PAG1 as HADD
00C0                    SLAW      –0xc0                    ! SLA+W to be transmitted
00C1                    SLAR      –0xc1                    ! SLA+R to be transmitted
0018                    SELRB3    –0x18                    ! Select Register Bank 3

                        !*************************************************************************************************
                        ! LOCATIONS IN DATA RAM
                        !*************************************************************************************************
0030                    MTD       –0x30                    ! MST/TRX/DATA base address
0038                    MRD       –0x38                    ! MST/REC/DATA base address
0040                    SRD       –0x40                    ! SLV/REC/DATA base address
0048                    STD       –0x48                    ! SLV/TRX/DATA base address

0053                    BACKUP        –0x53                ! Backup from NUMBYTMST
                                                           ! To restore NUMBYTMST in case
                                                           ! of an Arbitration Loss.
0052                    NUMBYTMST     –0x52                ! Number of bytes to transmit
                                                           ! or receive as MST.
0051                    SLA           –0x51                ! Contains SLA+R/W to be
                                                           ! transmitted.
0050                    HADD          –0x50                ! High Address byte for STATE 0
                                                           ! till STATE 25.
```

```
                        !*************************************************************************************************
                        ! INITIALIZATION ROUTINE
                        ! Example to initialize IIC Interface as slave receiver or slave transmitter and
                        ! start a MASTER TRANSMIT or a MASTER RECEIVE function. 4 bytes will be transmitted or received.
                        !*************************************************************************************************
                        .sect   strt
                        .base   0x00
0000    4100                            ajmp    INIT                    ! RESET

                        .sect   initial
                        .base   0x200
0200    75DB31          INIT:           mov     S1ADR,#OWNSLA           ! Load own SLA + enable
                                                                        ! general call recognition
0203    D296                            setb    P1(6)                   ! P1.6 High level.
0205    D297                            setb    P1(7)                   ! P1.7 High level.
0207    755001                          mov     HADD,#PAG1
020A    43A8A0                          orl     IEN0,#ENSI01            ! Enable SI01 interrupt
020D    C2BD                            clr     SI01HP                  ! SI01 interrupt low priority
020F    75D8C5                          mov     S1CON, #ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                        ! Initialize SLV funct.

                        !*************************************************************************************************

                        !-------------------------------------------------
                        ! START MASTER TRANSMIT FUNCTION
                        !-------------------------------------------------

0212    755204                          mov     NUMBYTMST,#0x4          ! Transmit 4 bytes.
0215    7551C0                          mov     SLA,#SLAW               ! SLA+W, Transmit funct.
0218    D2DD                            setb    STA                     ! set STA in S1CON

                        !-------------------------------------------------
                        ! START MASTER RECEIVE FUNCTION
                        !-------------------------------------------------
021A    755204                          mov     NUMBYTMST,#0x4          ! Receive 4 bytes.
021D    7551C1                          mov     SLA,#SLAR               ! SLA+R, Receive funct.
0220    D2DD                            setb    STA                     ! set STA in S1CON


                        !*************************************************************************************************
                        ! SI01 INTERRUPT ROUTINE
                        !*************************************************************************************************
                        .sect   intvec                                  ! SI01 interrupt vector
                        .base   0x00

                        ! S1STA and HADD are pushed onto the stack.
                        ! They serve as return address for the RET instruction.
                        ! The RET instruction sets the Program Counter to address HADD,
                        ! S1STA and jumps to the right subroutine.

002B    C0D0                            push    psw                     ! save  psw
002D    C0D9                            push    S1STA
002F    C050                            push    HADD
0031    22                              ret                             ! JMP to address HADD,S1STA.

                        !-------------------------------------------------
                        ! STATE    : 00, Bus error.
                        ! ACTION   : Enter not addressed SLV mode and release bus. STO reset.
                        !-------------------------------------------------
                        .sect   st0
                        .base   0x100

0100    75D8D5                          mov     S1CON,#ENS1_NOTSTA_STO_NOTSI_AA_CR0 ! clr SI
                                                                                    ! set  STO,AA
0103    D0D0                            pop     psw
0105    32                              reti
```

```
                        !*************************************************************************************************
                        !*************************************************************************************************
                        ! MASTER STATE SERVICE ROUTINES
                        !*************************************************************************************************
                        ! State 08 and State 10 are both for MST/TRX and MST/REC.
                        ! The R/W bit decides whether the next state is within
                        ! MST/TRX mode or within MST/REC mode.
                        !*************************************************************************************************

                        !-------------------------------------------------------------
                        ! STATE     : 08, A, START condition has been transmitted.
                        ! ACTION    : SLA+R/W are transmitted, ACK bit is received.
                        !-------------------------------------------------------------
                        .sect       mts8
                        .base       0x108

0108    8551DA                              mov   S1DAT,SLA                     ! Load  SLA+R/W
010B    75D8C5                              mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                                ! clr  SI
010E    01A0                                ajmp  INITBASE1

                        !-------------------------------------------------------------
                        ! STATE     : 10, A repeated START condition has been
                        !               transmitted.
                        ! ACTION    : SLA+R/W are transmitted, ACK bit is received.
                        !-------------------------------------------------------------
                        .sect       mts10
                        .base       0x110

0110    8551DA                              mov   S1DAT,SLA                     ! Load  SLA+R/W
0113    75D8C5                              mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                                ! clr  SI
010E    01A0                                ajmp  INITBASE1

                        .sect       ibase1
                        .base       0xa0
00A0    75D018  INITBASE1:                  mov   psw,#SELRB3
00A3    7930                                mov   r1,#MTD
00A5    7838                                mov   r0,#MRD
00A7    855253                              mov   BACKUP,NUMBYTMST              ! Save initial value
00AA    D0D0                                pop   psw
00AC    32                                  reti



                        !*************************************************************************************************
                        !*************************************************************************************************
                        ! MASTER TRANSMITTER STATE SERVICE ROUTINES
                        !*************************************************************************************************
                        !*************************************************************************************************

                        !-------------------------------------------------------------
                        ! STATE     : 18, Previous state was STATE 8 or STATE 10, SLA+W have been transmitted,
                        !               ACK has been received.
                        ! ACTION    : First DATA is transmitted, ACK bit  is received.
                        !-------------------------------------------------------------
                        .sect       mts18
                        .base       0x118

0118    75D018                              mov   psw,#SELRB3
011B    87DA                                mov   S1DAT,@r1
011D    01B5                                ajmp  CON
```

```
                        !-------------------------------------------------------
                        ! STATE    : 20, SLA+W have been transmitted, NOT ACK has been received
                        ! ACTION   : Transmit STOP condition.
                        !-------------------------------------------------------
                        .sect      mts20
                        .base      0x120

0120  75D8D5                                  mov   S1CON,#ENS1_NOTSTA_STO_NOTSI_AA_CR0
                                                                    ! set  STO, clr  SI
0123  D0D0                                    pop   psw
0125  32                                      reti

                        !-------------------------------------------------------
                        ! STATE    : 28, DATA of S1DAT have been transmitted, ACK received.
                        ! ACTION   : If Transmitted DATA is last DATA then transmit a STOP condition,
                        !              else transmit next DATA.
                        !-------------------------------------------------------
                        .sect      mts28
                        .base      0x128

0128  D55285                                  djnz  NUMBYTMST,NOTLDAT1      ! JMP if NOT last DATA
012B  75D8D5                                  mov   S1CON,#ENS1_NOTSTA_STO_NOTSI_AA_CR0
                                                                    ! clr  SI, set  AA
012E  01B9                                    ajmp  RETmt

                        .sect      mts28sb
                        .base      0x0b0
00B0  75D018    NOTLDAT1:                     mov   psw,#SELRB3
00B3  87DA                                    mov   S1DAT,@r1
00B5  75D8C5    CON:                          mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                    ! clr  SI, set   AA
00B8  09                                      inc   r1
00B9  D0D0      RETmt     :                   pop   psw
00BB  32                                      reti
                        !-------------------------------------------------------
                        ! STATE    : 30, DATA of S1DAT have been transmitted, NOT ACK received.
                        ! ACTION   : Transmit a STOP condition.
                        !-------------------------------------------------------
                        .sect      mts30
                        .base      0x130

0130  75D8D5                                  mov   S1CON,#ENS1_NOTSTA_STO_NOTSI_AA_CR0
                                                                    ! set  STO, clr  SI
0133  D0D0                                    pop   psw
0135  32                                      reti

                        !-------------------------------------------------------
                        ! STATE    : 38, Arbitration lost in SLA+W or DATA.
                        ! ACTION   : Bus is released, not addressed SLV mode is entered.
                        !              A new START condition is transmitted when the IIC bus is free again.
                        !-------------------------------------------------------
                        .sect      mts38
                        .base      0x138

0138  75D8E5                                  mov   S1CON,#ENS1_STA_NOTSTO_NOTSI_AA_CR0
013B  855352                                  mov   NUMBYTMST,BACKUP
013E  01B9                                    ajmp  RETmt
```

```
                        !– – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – –
                        ! STATE    : 80, Previously addressed with own SLA. DATA received, ACK returned.
                        ! ACTION   : Read DATA.
                        !            IF received DATA was the last
                        !            THEN superfluous DATA will be received  and NOT ACK returned
                                     ELSE next DATA will be received and ACK returned.
                        !– – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – –
                        .sect      srs80
                        .base      0x180

0180    75D018                              mov   psw,#SELRB3
0183    A6DA                                mov   @r0,S1DAT                    ! Read received DATA
0185    01D8                                ajmp  REC2

                        .sect      srs80s
                        .base      0xd8

00D8    D906            REC2:               djnz  r1,NOTLDAT3
00DA    75D8C1          LDAT:               mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_NOTAA_CR0
                                                                                ! clr  SI,AA
00DD    D0D0                                pop   psw
00DF    32                                  reti
00E0    75D8C5          NOTLDAT3:           mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                                ! clr  SI, set  AA
00E3    08                                  inc   r0
00E4    D0D0            RETsr:              pop   psw
00E6    32                                  reti

                        !– – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – –
                        ! STATE    : 88, Previously addressed with own SLA. DATA received NOT ACK returned.
                        ! ACTION   : No save of DATA, Enter NOT addressed SLV mode.
                        !            Recognition of own SLA. General call recognized, if S1ADR. 0–1.
                        !– – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – –
                        .sect      srs88
                        .base      0x188

0188    75D8C5                              mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                                ! clr  SI, set  AA
018B    01E4                                ajmp  RETsr

                        !– – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – –
                        ! STATE    : 90, Previously addressed with general call.
                        !            DATA has been received, ACK has been returned.
                        ! ACTION   : Read DATA.
                                     After General call only one byte will be received with ACK
                        !            the second DATA will be received with NOT ACK.
                        !            DATA will be received and NOT ACK returned.
                        !– – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – –
                        .sect      srs90
                        .base      0x190

0190    75D018                              mov   psw,#SELRB3
0193    A6DA                                mov   @r0,S1DAT                    ! Read received DATA
0195    01DA                                ajmp  LDAT
                        !– – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – –
                        ! STATE    : 98, Previously addressed with general call.
                        !            DATA has been received, NOT ACK has been returned.
                        ! ACTION   : No save of DATA, Enter NOT addressed SLV mode.
                                     Recognition of own SLA. General call recognized, if S1ADR. 0–1.
                        !– – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – – –
                        .sect      srs98
                        .base      0x198

0198    75D8C5                              mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                                ! clr  SI, set  AA
019B    D0D0                                pop   psw
019D    32                                  reti
```

```
                        !-------------------------------------------------------
                        ! STATE    : B8, DATA has been transmitted, ACK received.
                        ! ACTION   : DATA will be transmitted, ACK bit is received.
                        !-------------------------------------------------------
                        .sect      stsb8
                        .base      0x1b8
01B8    75D018                              mov   psw,#SELRB3
01BB    87DA                                mov   S1DAT,@r1
01BD    01F8                                ajmp  SCON

                        .sect      scn
                        .base      0xf8

00F8    75D8C5          SCON:               mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                    ! clr  SI, set  AA
00FB    09                                  inc   r1
00FC    D0D0                                pop   psw
00FE    32                                  reti
                        !-------------------------------------------------------
                        ! STATE    : C0, DATA has been transmitted, NOT ACK received.
                        ! ACTION   : Enter not addressed SLV mode.
                        !-------------------------------------------------------
                        .sect      stsc0
                        .base      0x1c0
01C0    75D8C5                              mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                    ! clr  SI, set  AA
01C3    D0D0                                pop   psw
01C5    32                                  reti

                        !-------------------------------------------------------
                        ! STATE    : C8, Last DATA has been transmitted (AA=0), ACK received.
                        ! ACTION   : Enter not addressed SLV mode.
                        !-------------------------------------------------------
                        .sect      stsc8
                        .base      0x1c8
01C8    75D8C5                              mov   S1CON,#ENS1_NOTSTA_NOTSTO_NOTSI_AA_CR0
                                                                    ! clr  SI, set  AA
01CB    D0D0                                pop   psw
01CD    32                                  reti


                        !*******************************************************************************************************
                        !*******************************************************************************************************
                        ! END OF SI01 INTERRUPT ROUTINE
                        !*******************************************************************************************************
                        !*******************************************************************************************************
```

DC ELECTRICAL CHARACTERISTICS

V_{SS}, AV_{SS} = 0 V

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS		UNIT
			MIN	MAX	
I_{DD}	Supply current operating	See notes 1 and 2 f_{OSC} = 16 MHz		16	mA
I_{ID}	Idle mode	See notes 1 and 3 f_{OSC} = 16 MHz		4	mA
I_{PD}	Power-down current	See notes 1 and 4; 2 V < V_{PD} < V_{DD} max		50	µA
Inputs					
V_{IL}	Input low voltage, except $\overline{EA}$, P1.6, P1.7		–0.5	$0.2V_{DD}$–0.1	V
V_{IL1}	Input low voltage to $\overline{EA}$		–0.5	$0.2V_{DD}$–0.3	V
V_{IL2}	Input low voltage to P1.6/SCL, P1.7/SDA[5]		–0.5	$0.3V_{DD}$	V
V_{IH}	Input high voltage, except XTAL1, RST		$0.2V_{DD}$+0.9	V_{DD}+0.5	V
V_{IH1}	Input high voltage, XTAL1, RST		$0.7V_{DD}$	V_{DD}+0.5	V
V_{IH2}	Input high voltage, P1.6/SCL, P1.7/SDA[5]		$0.7V_{DD}$	6.0	V
I_{IL}	Logical 0 input current, ports 1, 2, 3, 4, except P1.6, P1.7	V_{IN} = 0.45 V		–50	µA
I_{TL}	Logical 1-to-0 transition current, ports 1, 2, 3, 4, except P1.6, P1.7	See note 6		–650	µA
$\pm I_{IL1}$	Input leakage current, port 0, $\overline{EA}$, STADC, $\overline{EW}$	0.45 V < V_I < V_{DD}		10	µA
$\pm I_{IL2}$	Input leakage current, P1.6/SCL, P1.7/SDA	0 V < V_I < 6 V 0 V < V_{DD} < 5.5 V		10	µA
$\pm I_{IL3}$	Input leakage current, port 5	0.45 V < V_I < V_{DD}		1	µA
$\pm I_{IL4}$	Input leakage current, ports 1, 2, 3, 4 in high impedance mode	0.45 V < V_{in} < V_{DD}		10	µA
Outputs					
V_{OL}	Output low voltage, ports 1, 2, 3, 4, except P1.6, P1.7	I_{OL} = 1.6mA[7]		0.4	V
V_{OL1}	Output low voltage, port 0, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$	I_{OL} = 3.2mA[7]		0.4	V
V_{OL2}	Output low voltage, P1.6/SCL, P1.7/SDA	I_{OL} = 3.0mA[7]		0.4	V
V_{OH}	Output high voltage, ports 1, 2, 3, 4, except P1.6/SCL, P1.7/SDA	V_{CC} = 2.7 V I_{OH} = –20 µA	V_{CC} – 0.7		V
		V_{CC} = 4.5 I_{OH} = –30 µA	V_{CC} – 0.7		V
V_{OH1}	Output high voltage (port 0 in external bus mode, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$)[8]	V_{CC} = 2.7 V I_{OH} = –3.2mA	V_{CC} – 0.7		V
V_{OH2}	Output high voltage (RST)	$-I_{OH}$ = 400 µA $-I_{OH}$ = 120 µA	2.4 $0.8V_{DD}$		V V
R_{RST}	Internal reset pull-down resistor		40	225	kΩ
C_{IO}	Pin capacitance	Test freq = 1 MHz, T_{amb} = 25°C		10	pF
Analog Inputs					
AV_{DD}	Analog supply voltage: 87C554[9]	AV_{DD} = V_{DD}±0.2 V	2.7	5.5	V
AI_{DD}	Analog supply current: operating:	Port 5 = 0 to AV_{DD}		1.2	mA
AI_{ID}	Idle mode: 87C554			50	µA
AI_{PD}	Power-down mode: 87C554	2 V < AV_{PD} < AV_{DD} max		50	µA