Welcome to E-XFL.COM

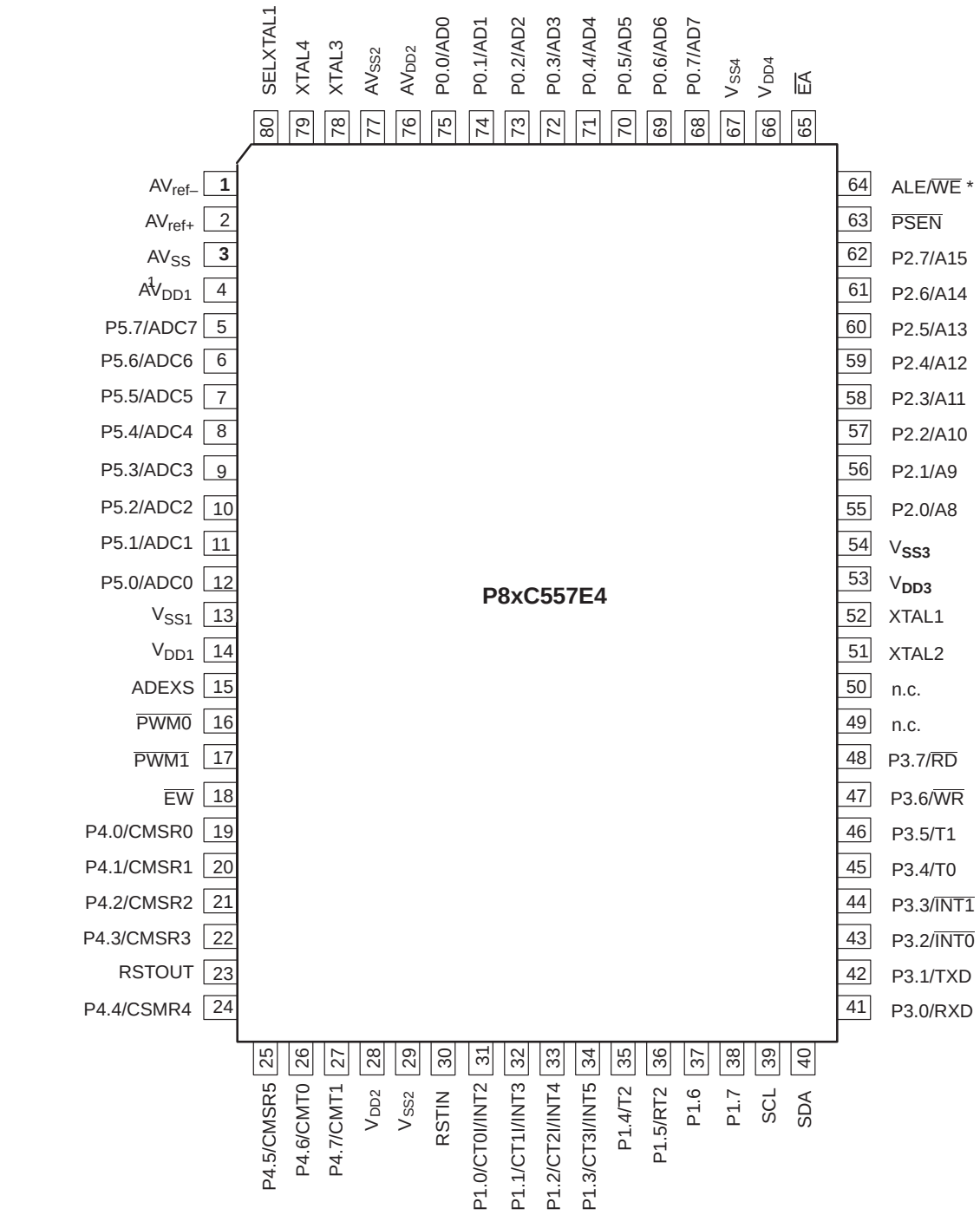
What is "Embedded - Microcontrollers"?

"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	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-BQFP
Supplier Device Package	80-PQFP (14x20)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/p80c557e4efb-01-51

4. PINNING



n.c. = not connected

* = only P89C557E4 with alternate function $\overline{WE}$

Figure 3. Pinning diagram for QFP80 (SOT318)

4.1 PIN DESCRIPTION

SYMBOL	PIN	DESCRIPTION
AV_{ref-} AV_{ref+}	1 2	**Low end** of analog to digital conversion reference resistor **High end** of analog to digital conversion reference resistor.
AV_{SS1} AV_{DD1}	3 4	**Analog** ground for ADC **Analog** power supply (+5 V) for ADC
AV_{SS2} AV_{DD2}	77 76	**Analog** ground; for PLL oscillator **Analog** power supply; (+5 V) for PLL oscillator
P5.7 – P5.0	5 – 12	**Port 5** 8-bit input port Port pin — Alternative function P5.0–P5.7 — Eight input channels to ADC (ADC0–ADC7)
V_{DD1}, V_{DD2}, V_{DD3}, V_{DD4}	14, 28, 53, 66	**Digital power supply:** +5 V power supply pins during normal operation and power reduction modes. All pins must be connected.
V_{SS1}, V_{SS2} V_{SS3}, V_{SS4}	13, 29, 54, 67	**Digital ground:** circuit ground potential. All pins must be connected.
ADEXS	15	**Start ADC operation:** Input starting analog to digital conversion triggered by a programmable edge (ADC operation can also be started by software). This pin must not float
$\overline{PWM0}$	16	**Pulse width modulation** output 0
$\overline{PWM1}$	17	**Pulse width modulation** output 1
$\overline{EW}$	18	**Enable watchdog timer:** Enable for T3 watchdog timer and disable Power-down Mode.This pin must not float.
P4.0 – P4.7	19 – 22 24 – 27	**Port 4** 8-bit quasi-bidirectional I/O port Port pin — Alternative function P4.0 — CMSR0 } P4.1 — CMSR1 } P4.2 — CMSR2 } compare and set/reset P4.3 — CMSR3 } outputs on a match with timer T2 P4.4 — CMSR4 } P4.5 — CMSR5 } P4.6 — CMT0 } compare and toggle outputs P4.7 — CMT1 } on a match with timer T2
RSTIN	30	**Reset:** Input to reset the P8xC557E4.
RSTOUT	23	**Reset:** Output of the P8xC557E4 for resetting peripheral devices during initialization and Watchdog Timer overflow.
P1.0 – P1.7	31 – 38	**Port 1** 8-bit quasi-bidirectional I/O port Port pin — Alternative function P1.0 — CT0I/INT2} P1.1 — CT1I/INT3} : Capture timer inputs for P1.2 — CT2I/INT4} timer T2 or external interrupt inputs P1.3 — CT3I/INT5} P1.4 — T2 : T2 event input, rising edge triggered P1.5 — RT2 : T2 timer reset input, rising edge triggered P1.6 P1.7
SCL	39	**I^2C-bus serial clock** I/O port
SDA	40	**I^2C-bus serial data** I/O port If SCL and SDA are not used, they must be connected to V_{SS}.

6.6.2 Configuration and Operation

Every A/D conversion is an autoscan conversion. The two user selectable general operation modes are continuous scan and one-time scan mode.

The desired analog input port channel/s for conversion is/are selected by programming A/D input port scan-select bits in SFR ADPSS. An analog input channel is included in the autoscan loop if the corresponding bit in ADPSS is 1, a channel is skipped if the corresponding bit in ADPSS is 0.

An autoscan is always started according to the lowest bit position of ADPSS that contains a 1.

An autoscan conversion is started by setting the flag ADSST in register ADCON either by software or by an external start signal at input pin ADEXS, if enabled. Either no edge (external start totally disabled), a rising edge or/and a falling edge of ADEXS is selectable for external conversion start by the bits ADSRE and ADSFE in register ADCON.

After completion of an A/D conversion the 10-bit result is stored in the corresponding 10-bit buffer register. Then the next analog input is selected according to the next higher set bit position in ADPSS, converted and stored, and so on. When the result of the last conversion of this autoscan loop is stored, flag ADCON.4/ADINT, the ADC interrupt flag, is set. It is not cleared by interrupt hardware – it must be cleared by software.

In continuous scan mode (ADCON.2/ADCSA=1) the ADC start and status flag ADCON.3/ADSST retains the set state and the autoscan loop restarts from the beginning. In one-time scan mode (ADCSA=0) conversions stop after the last selected analog input was converted, ADINT is set and ADSST is cleared automatically.

ADSST cannot be set (neither externally nor by software) as long as ADINT=1, i.e. as long as ADINT is set, a new conversion start – by setting flag ADSST – is inhibited; actually it is only delayed until ADINT is cleared.

(If a '1' is written to ADSST while ADINT=1, this new value is internally latched and preserved, not setting ADSST until ADCON.4/ADINT=0. In this state, a read of SFR ADCON will display ADCON.3/ADSST=0, because always the effective ADC status is read.)

Note that under software control the analog inputs can also be converted in arbitrary order, when one-time scan mode is selected and in SFR ADPSS only one bit is set at a time. In this case ADINT is set and ADSST is cleared after every conversion.

6.6.3 Resolution and Characteristics

The ADC system has its own analog supply pins AV_{DD} and AV_{SS}. It is referenced by two special reference voltage input pins sourcing the resistance ladder of the DAC: AV_{ref+} and AV_{ref-}. The voltage between AV_{REF+} and AV_{REF-} defines the full-scale range. Due to the 10-bit resolution the full scale range is divided into 1024 unit steps. The unit step voltage is 1 LSB, which is typically 5 mV (AV_{ref+} = 5.12 V, AV_{ref-} = 0 V = AV_{SS}).

The DAC's resistance ladder has 1023 equally spaced taps, separated by a unit resistance 'R'. The first tap is located 0.5 x R above AV_{ref-}, the last tap is located 1.5 x R below AV_{ref+}. This results in a total ladder resistance of 1024 x R. This structure ensures that the DAC is monotonic and results in a symmetrical quantization error. For input voltages between AV_{ref-} and (AV_{ref-} + 1/2 LSB) the 10-bit conversion result code will be 00 0000 0000 B = 000H = 0D. For input voltages between (AV_{ref+} – 3/2 LSB) and AV_{ref+} the 10-bit conversion result code will be 11 1111 1111 B = 3FFH = 1023D.

The result code corresponding to an analog input voltage (AV_{in}) can be calculated from the formula:

$$\text{ResultCode} = 1024 \times \frac{AV_{IN} - AV_{ref-}}{AV_{ref+} - AV_{ref-}}$$

The analog input voltage should be stable when it is sampled for conversion. At any times the input voltage slew rate must be less than 10 V/ms (5 V conversion range) in order to prevent an undefined result.

This maximum input voltage slew rate can be ensured by an RC low pass filter with R = 2k2 and C = 100 nF. The capacitor between analog input pin and analog ground pin shall be placed close to the pins in order to have maximum effect in minimizing input noise coupling.

6.7 Timer/Counters

The P8xC557E4 contains three 16-bit timer/event counters: Timer 0, Timer 1 and Timer T2 and one 8-bit timer, T3. Timer 0 and Timer 1 may be programmed to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests

6.7.1 Timer 0 and Timer 1

Timers 0 and 1 each have a control bit in SFR TMOD that selects the timer or counter function of the corresponding timer.

In the timer function, the register is incremented every machine cycle. Thus, one can think of it as counting machine cycles. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

In the counter function, the register is incremented in response to a 1-to-0 transition at the corresponding external input pin, T0 or T1. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples show a HIGH in one cycle and a LOW in the next cycle, the counter is incremented. Thus, it takes two machine cycles (24 oscillator periods) to recognize a 1-to-0 transition. There are no restrictions on the duty cycle of the external input signal, but to insure that a given level is sampled at least once before it changes, it should be held for at least one full machine cycle.

Timer 0 and Timer 1 can be programmed independently to operate in one of 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: one 8-bit time-interval or event counter and one 8-bit time-interval counter
  –Timer 1: stopped

	7	6	5	4	3	2	1	0
TCON (88H)	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Figure 20. Timer/Counter mode control (TCON) register.

Table 14. Description of TCON bits

SYMBOL	BIT	FUNCTION
TF1	TCON.7	Timer 1 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine.
TR1	TCON.6	Timer 1 run control bit. Set/cleared by software to turn Timer/Counter on/off.
TF0	TCON.5	Timer 0 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine.
TR0	TCON.4	Timer 0 run control bit. Set/cleared by software to turn Timer/Counter on/off.
IE1	TCON.3	Interrupt 1 edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
IT1	TCON.2	Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.
IE0	TCON.1	Interrupt 0 edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
IT0	TCON.0	Interrupt 0 type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.

	7	6	5	4	3	2	1	0
TM2CON (EAH)	T2IS1	T2IS0	T2ER	T2BO	T2P1	T2P0	T2MS1	T2MS0

Figure 22. T2 control register (TM2CON).

Table 15. Description of TM2CON bits

SYMBOL	BIT	FUNCTION
T2IS1	TM2CON.7	Timer T2 16-bit overflow interrupt select
T2IS0	TM2CON.6	Timer T2 byte overflow interrupt select
T2ER	TM2CON.5	Timer T2 external reset enable. When this bit is set, Timer T2 may be reset by a rising edge on RT2 (P1.5).
T2BO	TM2CON.4	Timer T2 byte overflow interrupt flag
T2P1	TM2CON.3	Timer T2 prescaler select
T2P0	TM2CON.2	
T2MS1	TM2CON.1	Timer T2 mode select
T2MS0	TM2CON.0	

Table 16. Timer 2 prescaler select

T2P1	T2P0	TIMER T2 CLOCK
0	0	Clock source
0	1	Clock source/2
1	0	Clock source/4
1	1	Clock source/8

Table 17. Timer 2 mode select

T2MS1	T2MS0	MODE SELECTED
0	0	Timer T2 halted (off)
0	1	T2 clock source = $f_{CLK}/12$
1	0	Test mode; do not use
1	1	T2 clock source = pin T2

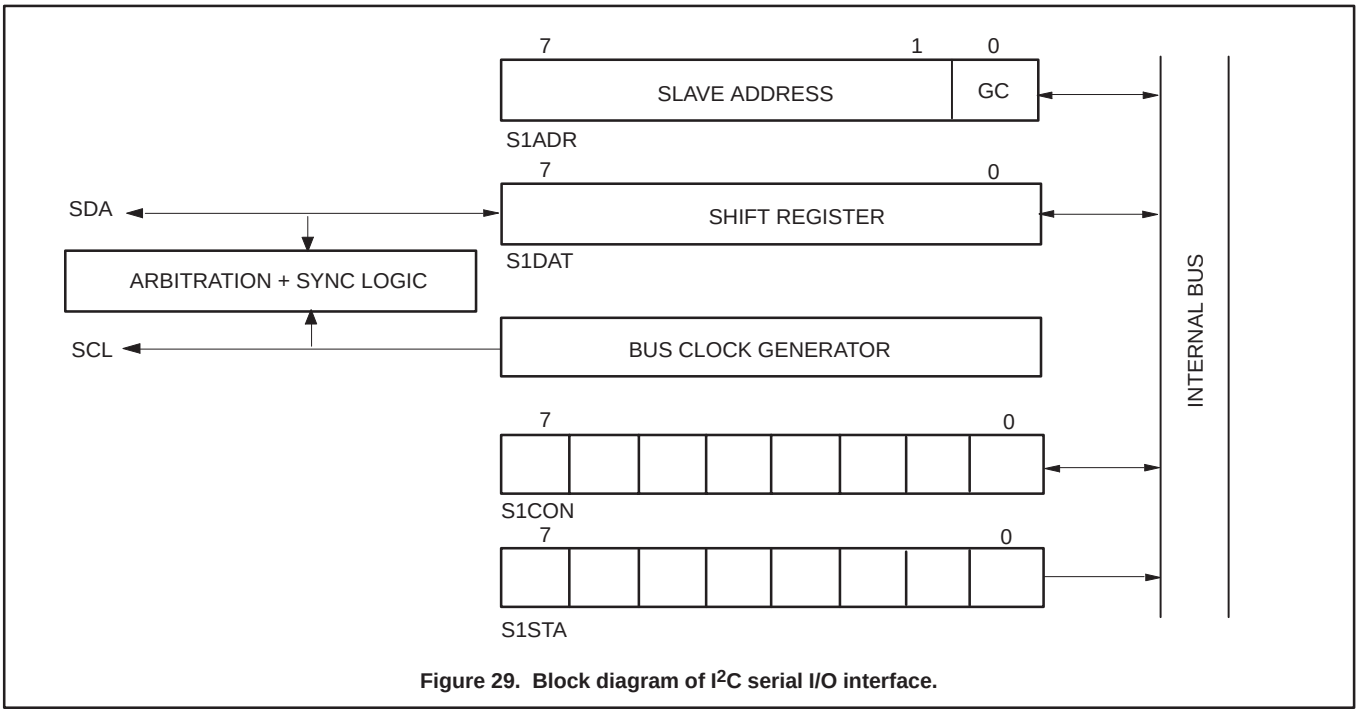


Figure 29. Block diagram of I^2C serial I/O interface.

The polling cycle is repeated with every machine cycle, and the values polled are the values present at S5P2 of the previous machine cycle. Note that if an interrupt flag is active but is not being responded to because of one of the above conditions, and if the flag is inactive when the blocking condition is removed, then the blocked interrupt will not be serviced. Thus, the fact that the interrupt flag was once active but not serviced is not remembered. Every polling cycle is new.

The processor acknowledges an interrupt request by executing a hardware-generated LCALL to the appropriate service routine. In some cases it also clears the flag which generated the interrupt, and in others it does not. It clears the Timer 0, Timer 1, and external interrupt flags. An external interrupt flag (IE0 or IE1) is cleared only if it was transition-activated. All other interrupt flags are not cleared by hardware and must be cleared by the software. The LCALL pushes the contents of the program counter on to the stack (but it does not save the PSW) and reloads the PC with an address that depends on the source of the interrupt being vectored to as shown in Table 38.

Execution proceeds from the vector address until the RETI instruction is encountered. The RETI instruction clears the “priority level active” flip-flop that was set when this interrupt was acknowledged. It then pops the top two bytes from the stack and reloads the program counter. Execution of the interrupted program continues from where it was interrupted.

	7	6	5	4	3	2	1	0
IEN0 (A8H)	EA	EAD	ES1	ES0	ET1	EX1	ET0	EX0

Figure 34. Interrupt enable register (IEN0).

Table 33. Description of IEN0 bits

SYMBOL	BIT	FUNCTION
EA	IEN0.7	Global enable/disable control 0 = No interrupt is enabled 1 = Any individually enabled interrupt will be accepted
EAD	IEN0.6	Enable ADC interrupt
ES1	IEN0.5	Enable SIO1 (I^2C) interrupt
ES0	IEN0.4	Enable SIO0 (UART) interrupt
ET1	IEN0.3	Enable Timer 1 interrupt
EX1	IEN0.2	Enable External interrupt 1 / Seconds interrupt
ET0	IEN0.1	Enable Timer 0 interrupt
EX0	IEN0.0	Enable External interrupt 0

	7	6	5	4	3	2	1	0
IEN1 (E8H)	ET2	ECM2	ECM1	ECM0	ECT3	ECT2	ECT1	ECT0

Figure 36. Interrupt enable register (IEN1).

Table 34. Description of IEN1 bits

SYMBOL	BIT	FUNCTION
ET2	IEN1.7	Enable T2 overflow interrupt(s)
ECM2	IEN1.6	Enable T2 comparator 2 interrupt
ECM1	IEN1.5	Enable T2 comparator 1 interrupt
ECM0	IEN1.4	Enable T2 comparator 0 interrupt
ECT3	IEN1.3	Enable T2 capture register 3 interrupt
ECT2	IEN1.2	Enable T2 capture register 2 interrupt
ECT1	IEN1.1	Enable T2 capture register 1 interrupt
ECT0	IEN1.0	Enable T2 capture register 0 interrupt

If the enable bit is 0, then the interrupt is disabled, if the enable bit is 1, then the interrupt is enabled.

	7	6	5	4	3	2	1	0
IP0 (B8H)	–	PAD	PS1	PS0	PT1	PX1	PT0	PX0

Figure 37. Interrupt priority register (IP0).

Table 35. Description of IP0 bits

SYMBOL	BIT	FUNCTION
–	IP0.7	Reserved for future use
PAD	IP0.6	ADC interrupt priority level
PS1	IP0.5	SIO1 (I^2C) interrupt priority level
PS0	IP0.4	SIO0 (UART) interrupt priority level
PT1	IP0.3	Timer 1 interrupt priority level
PX1	IP0.2	External interrupt 1/Seconds interrupt priority level
PT0	IP0.1	Timer 0 interrupt priority level
PX0	IP0.0	External interrupt 0 priority level

7. INSTRUCTION SET

The P8xC557E4 uses the powerful instruction set of the PCB80C51. It consists of 49 single-byte, 45 two-byte and 17 three-byte instructions. Using a 16 MHz quartz, 64 of the instructions are executed in 0.75 μs, 45 in 1,5 μs and the multiply, divide instructions in 3 μs.

A summary of the instruction set is given in Table 43.

The P8xC557E4 has additional Special Function Registers to control the on-chip peripherals.

7.1 Addressing Modes

Most instructions have a “destination, source” field that specifies the data type, addressing modes and operands involved. For all these instructions, except for MOVs, the destination operand is also the source operand (e.g., ADD A,R7).

There are five kinds of addressing modes:

- Register Addressing
  - R0 – R7 (4 banks)
  - A,B,C (bit), AB (2 bytes), DPTR (double byte)
- Direct Addressing
  - lower 128 bytes of internal Main RAM (including the 4 R0–R7 register banks)
  - Special Function Registers
  - 128 bits in a subset of the internal Main RAM
  - 128 bits in a subset of the Special Function Registers
- Register-Indirect Addressing
  - internal Main RAM (@R0, @R1, @SP [PUSH/POP])
  - internal Auxiliary RAM (@R0, @R1, @DPTR)
  - external Data Memory (@R0, @R1, @DPTR)
- Immediate Addressing
  - Program Memory (in-code 8 bit or 16 bit constant)
- Base-Register-plus Index-Register-Indirect Addressing
  - Program Memory look-up table (@DPTR+A, @PC+A)

The first three addressing modes are usable for destination operands.

7.1.1 80C51 Family Instruction Set

Table 42. Instruction that affect Flag settings[1]

INSTRUCTION	FLAG		
	C	OV	AC
ADD	X	X	X
ADDC	X	X	X
SUBB	X	X	X
MUL	0	X	
DIV	0	X	
DA	X	X	
RRC	X		
RLC	X		
SETB C	1		
CLR C	0		
CPL C	X		
ANL C, bit	X		
ANL C,/bit	X		
ANL C, bit	X		
ORL C, bit	X		
MOV C, bit	X		
CJNE	X		

NOTES:
1. Note that operations on SFR byte address 208 or bit addresses 209-215 (i.e., the PSW or bits in the PSW) will also affect flag settings.

Notes on instruction set and addressing modes:

Rn	Register R7-R0 of the currently selected Register Bank.
direct	8-bit internal data location’s address. This could be an Internal Data RAM location (0-127) or a SFR [i.e., I/O port, control register, status register, etc. (128-255)].
@Ri	8-bit RAM location addressed indirectly through register R1 or R0 of the actual register bank.
#data	8-bit constant included in the instruction.
#data 16	16-bit constant included in the instruction
addr 16	16-bit destination address. Used by LCALL and LJMP. A branch can be anywhere within the 64 Kbytes Program Memory address space.
addr 11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 Kbytes 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.
bit	Direct Addressed bit in Internal Data RAM or Special Function Register.

Hexadecimal opcode cross-reference to Table 43:

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

Table 43. 80C51 Instruction Set Summary (Continued)

MNEMONIC		DESCRIPTION	BYTE	CYCLES	OPCODE (HEX.)
LOGICAL OPERATIONS (Continued)					
XRL	A,#data	Exclusive-OR immediate data to Accumulator	2	1	64
XRL	direct,A	Exclusive-OR Accumulator to direct byte	2	1	62
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3	2	63
CLR	A	Clear Accumulator	1	1	E4
CPL	A	Complement Accumulator	1	1	F4
RL	A	Rotate Accumulator left	1	1	23
RLC	A	Rotate Accumulator left through the carry	1	1	33
RR	A	Rotate Accumulator right	1	1	03
RRC	A	Rotate Accumulator right through the carry	1	1	13
SWAP	A	Swap nibbles within the Accumulator	1	1	C4
DATA TRANSFER					
MOV	A,Rn	Move register to Accumulator	1	1	E*
MOV	A,direct	Move direct byte to Accumulator	2	1	E5
MOV	A,@Ri	Move indirect RAM to Accumulator	1	1	E6, E7
MOV	A,#data	Move immediate data to Accumulator	2	1	74
MOV	Rn,A	Move Accumulator to register	1	1	F*
MOV	Rn,direct	Move direct byte to register	2	2	A*
MOV	RN,#data	Move immediate data to register	2	1	7*
MOV	direct,A	Move Accumulator to direct byte	2	1	F5
MOV	direct,Rn	Move register to direct byte	2	2	8*
MOV	direct,direct	Move direct byte to direct	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 Accumulator 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	2	1	76, 77
MOV	DPTR,#data16	Load Data Pointer with a 16-bit constant	3	2	90
MOVC	A,@A+DPTR	Move Code byte relative to DPTR to ACC	1	2	93
MOVC	A,@A+PC	Move Code byte relative to PC to ACC	1	2	83
MOVX	A,@Ri	Move AUX-RAM (8-bit addr) to ACC	1	2	E2, E3
MOVX	A,@DPTR	Move AUX-RAM (16-bit addr) to A_{CC}	1	2	E0
MOVX	@Ri,A	Move ACC to AUX-RAM (8-bit addr)	1	2	F2, F3
MOVX	@DPTR,A	Move ACC to AUX-RAM (16-bit addr)	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,Rn	Exchange register with Accumulator	1	1	C*
XCH	A,direct	Exchange direct byte with Accumulator	2	1	C5
XCH	A,@Ri	Exchange indirect RAM with Accumulator	1	1	C6, C7
XCHD	A,@Ri	Exchange low-order digit indirect RAM with ACC	1	1	D6, D7

Table 44. Instruction map P8xC557E4

second hexadecimal character of opcode

first hexadecimal character of opcode	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	AJMP addr11	LJMP addr16	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 addr11	LCALL addr16	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 addr11	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 addr11	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 addr11	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 addr11	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 addr11	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 addr11	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 addr11	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,#data16	ACALL addr11	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 addr11	MOV C, bit	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 addr11	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 addr11	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 addr11	SETB bit	SETB C	DA A	DNJZ dir, rel	XCHD A, @ Ri		DJNZ Rr, rel							
							0	1	0	1	2	3	4	5	6	7
E	MOVX A, @DPTR	AJMP addr11	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 addr11	MOVX A, @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

***) MOV A, ACC is not a valid instruction**

8. FLASH EEPROM

8.1 General

- 32 Kbytes electrically erasable internal program memory with Block-and Page-Erase option (”Flash Memory”).
- Internal fixed boot ROM.
- Up to 32 Kbytes external program memory in combination with the internal FEEPROM ($\overline{EA}$=1).
- Up to 64 Kbytes external program memory if the internal program memory is switched off ($\overline{EA}$=0).

The FEEPROM can be read and written byte-wise. Full Erase, Block Erase, and Page erase will erase 32 Kbytes, 256 bytes and 32 bytes respectively. In-circuit programming and out-of-circuit programming is possible. On-chip erase and write timing generation and on chip high voltage generation contribute to a user friendly interface.

8.2 Features

- Read:
  byte-wise
- Write:
  byte-wise within 2.5 ms.
  (previously erased by a page, block or full erase).
- Erase:
  Page Erase (32 bytes) within 5 ms.
  Block Erase (256 bytes) within 5 ms.
  Full Erase (32 Kbytes) within 5 ms.
  Erased bytes contain FFH.
- Endurance:
  100 erase and write cycles each byte at T_{amb} = 22°C
- Retention:
  10 years
- Out-of-circuit programming:
  Parallel programming with 87C51 compatible hardware Interface to programmer.
- In-circuit programming:
  Serial programming via RS232 interface under boot ROM program control. Auto baud rate selection.
  Intel Hex Object file Format.
  The user program can call routines in the boot ROM for erase, write and verify of the FEEPROM.
- High programming voltage generation: on chip
- Zero point on-chip oscillator and timer to generate the write and erase time durations.
- Programmable security for the code in the FEEPROM to prevent software piracy. The Security Byte is located in the highest address (7FFFH) of the FEEPROM.
- Supply voltage monitoring circuit on-chip to prevent loss of information in the FEEPROM during power-on and power-off.

8.3 Memory Map

Figure 48 shows the memory map of the user program memory and the boot ROM. They are located in the same program address space. Two bits UBS1 and UBS0 of the FEEPROM control special function register FMCON select between the two memory blocks.

User program memory selection
If UBS1 and UBS0 are both 0, then the user program memory is mapped into the 64 K program memory space and the boot ROM cannot be selected. This is the situation after a reset when $\overline{PSEN}$ and ALE have not been pulled down during reset. Program execution starts at 0000H in the internal FEEPROM or in the external program memory dependent on the level of $\overline{EA}$ during reset.

Boot ROM selection
After a reset program execution starts in the boot ROM when during reset $\overline{PSEN}$ and $\overline{EA}$ are pulled down while ALE stay high. The boot ROM size is 1 Kbyte. Besides the serial in-circuit programming routine the boot ROM contains the routines for erase, write and verify of the FEEPROM, which can be called by the user program (LCALL to the address space between 63 K and 64 K).

Switching between user program memory and boot ROM
Switching between user program memory (internal or external) and boot ROM is possible if UBS1 and UBS0 are 0,1. Then in the program memory address space between 0 and 63k the user program memory is selected and in the memory space between 63 K and 64 K the boot ROM is selected.

To switch from user program memory to boot ROM first UBS0 must be set (UBS1 stay 0) and a jump or call instruction to a location >63 K must be executed.

At the moment of crossing the 63 K address border by a return instruction the switching from boot ROM to user memory (internal or external) is performed. After crossing the 63 K address border UBS1 and UBS0 are cleared and the total 64 K memory space is mapped as user program memory. By clearing UBS1 and UBS0, no special requirements to the user program are necessary to do that after a read or erase or write routine.

A small restriction for memory switching is that no memory switching is allowed from or to the address space between 63 K and 64 K of the user program memory because the UBS bits must stay 0 in this range. This restriction can be avoided if the memory switching is always done by a subroutine in the address range between 0 and 63 K.

Description

The user program code in the FEEPROM is executed as in the standard 80C51 microcontroller. Erase and write cycles in the FEEPROM are always performed under control of the boot program in the boot ROM in the address space between 63 K and 64 K. Address and data parameters are passed via DPTR and accumulator A respectively. During an erase or write cycle in the FEEPROM no other access or program execution in the FEEPROM is possible. All interrupts must be disabled when the user program calls a user routine in the boot ROM.

The boot routine for serial programming takes care of addressing, data transfer, verify, high voltage control, error message and return to the user program memory. It also contains the serial communication routine.

The FEEPROM control register FMCON is a special function register. It contains the control bits for verify, write, erase and boot ROM switching.

	7	6	5	4	3	2	1	0
FMCON (FB)	UBS1	UBS0	HV	– [1)]	FCB3	FCB2	FCB1	FCB0

Figure 49. FEEPROM control register.

NOTE:
1. Reserved for future use; a write operation must write “0” to the location.

Table 45. Description of FMCON bits

UBS1	**UBS0**	**User - Boot selection bits**
0	0	User memory mapped from 0 to 64 K.
0	1	User memory mapped from 0 to 63 K. Boot ROM mapped from 63 K to 64 K.
1	0	User memory mapped from 0 to 63 K, but UBS1 bit cleared by hardware in this user address range. Boot ROM mapped from 63 K to 64 K. User software should not write “1” UBS1.
1	1	Boot ROM mapped from 0 to 64 K. User software should not write “1” UBS1.

HV	**High voltage indication bit.** Read only. Is “1” as long as the high voltage for an erase or write operation is present.

FCB3	**FCB2**	**FCB1**	**FCB0**	**Function Code Bits**
0	0	0	0	Value after Reset.
0	1	0	1	Byte Write or byte read (verify)
1	1	0	0	Page Erase (32 bytes boundaries).
0	0	1	1	Block Erase (256 bytes boundaries).
1	0	1	0	Full Erase (32 Kbytes).

The four FCB bits are write protected if the security feature is activated. Then only instructions in the internal program memory (FEEPROM) are able to write FCB (3–0), boot ROM and external program memory instructions cannot change FCB (3–0) except the full erase code can be loaded.

The duration of a write or erase operation is determined by the FEEPROM timer. This timer includes a zero point RC oscillator and cannot be controlled by software.

For calling a user routine in the boot ROM first all interrupts must be disabled and the DPTR and A have to be loaded with the desired values. After setting UBS0 = 1 and UBS1 = 0 and selecting the function via FCB-bits the respective user routine has to be called.

The table below lists the boot ROM user routines, which can be called by the user program. The content of FMCON, A and DPTR before the call is described by “(IN)” and the contents after the return is described by “(OUT)”. The boot ROM user routines do not change other registers or Data memory.

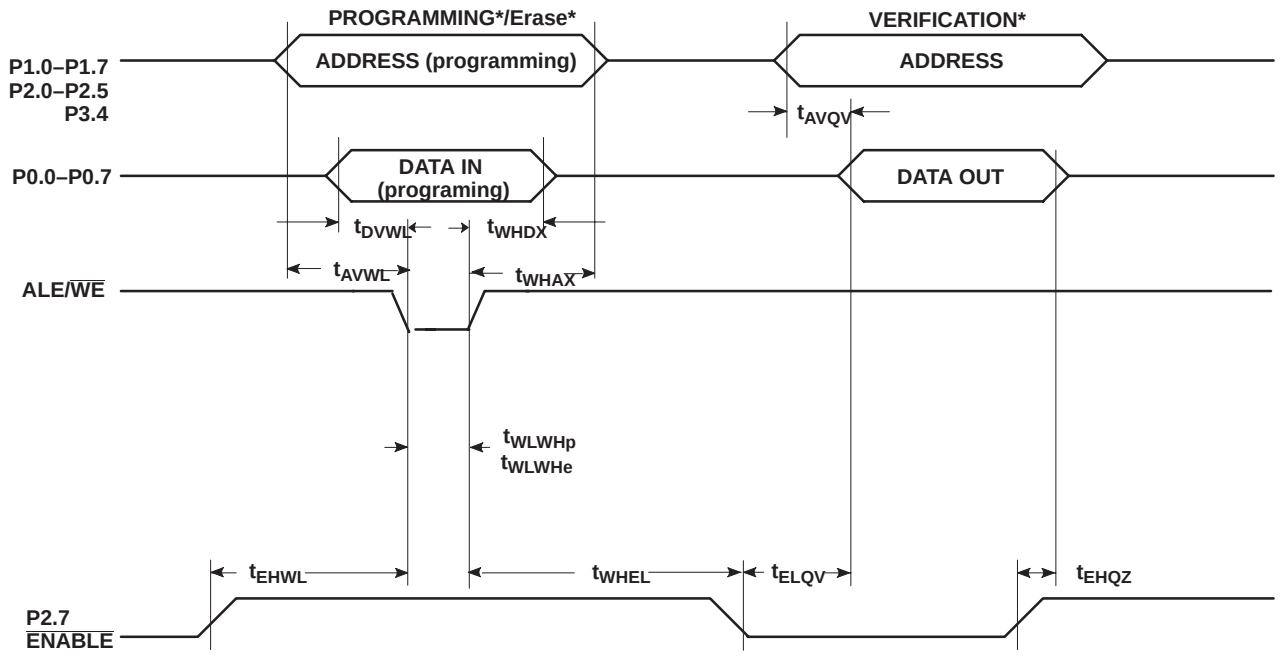
BOOT-ROM ROUTINE	**CALL ADDRESS**	**FMCON (IN)**	**FMCON (OUT)**	**ACC (IN)**	**ACC (OUT)**	**DPTR (IN)**	**DPTR (OUT)**
BYTE_READ	FFBAH	45H	15H	XXH	BYTE	BYTE ADDRESS	BYTE ADDRESS
BYTE_WRITE	FFADH	45H	15H	BYTE	BYTE [(V)]	BYTE ADDRESS	BYTE ADDRESS
PAGE_ERASE	FFAAH	4CH	1CH	XXH	08H	PAGE ADDRESS [1)]	PAGE ADDRESS [2)]
BLOCK_ERASE	FFA5H	43H	13H	XXH	02H	BLOCK ADDRESS [3)]	BLOCK ADDRESS [4)]
FULL_ERASE	FFA0H	4AH	1AH	XXH	0AH	XXXXH	0018H

X = don’t care or not defined
V = verified byte (read back)
1) = 5 LSB’s of DPTR are don’t care
2) = 5 LSB’s of DPTR are “0”
3) = 8 LSB’s of DPTR are don’t care
4) = 8 LSB’s of DPTR contain 08H.

FEEPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS

T_{amb} = –40 °C to +85 °C, V_{DD} = 5 V ± 10%, V_{SS} = 0 V (see Figure 53)

SYMBOL	PARAMETER	MIN	MAX	UNIT
$1/t_{CLK}$	System clock frequency (standard oscillator)	4	6	MHz
t_{AVWL}	Address setup to $\overline{WE}$ LOW	$48t_{CLK}$	–	
t_{WHAX}	Address hold after $\overline{WE}$ HIGH	$48t_{CLK}$	–	
t_{DVWL}	Data setup to $\overline{WE}$ LOW	$48t_{CLK}$	–	
t_{WHDX}	Data hold after $\overline{WE}$ HIGH	$48t_{CLK}$	–	
t_{EHWL}	P2.7 ($\overline{ENABLE}$) HIGH to $\overline{WE}$ LOW	$48t_{CLK}$	–	
t_{WHEL}	$\overline{WE}$ HIGH to P2.7 ($\overline{ENABLE}$) LOW	$48t_{CLK}$	–	
t_{WLWHp}	$\overline{WE}$ width (programming)	2.25	2.75	ms
t_{WLWHe}	$\overline{WE}$ width (erase)	4.5	5.5	ms
t_{AVQV}	Address to data valid	–	$48t_{CLK}$	
t_{ELQV}	P2.7 ($\overline{ENABLE}$) Low to data valid	–	$48t_{CLK}$	
t_{EHQZ}	Data float after P2.7 ($\overline{ENABLE}$) HIGH	0	$48t_{CLK}$	



* For ERASE conditions see Figure 50.
For PROGRAM conditions see Figure 51.
For VERIFY conditions see Figure 52.

Figure 53. FEEPROM Programming/Erase and Verification Waveforms

DC ELECTRICAL CHARACTERISTICS (Continued)

V_{DD} = 5 V (± 10%), V_{SS} = 0 V, T_{amb} = –40°C to +85°C (P8xC557E4EFx).
DC parameters not included here are the same as in the P8xC557E4EBx, DC electrical characteristics
All voltages with respect to V_{SS} unless otherwise specified.

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS MIN	LIMITS MAX	UNIT
R_{RST}	Internal reset pull-down resistor		50	150	kΩ
C_{IO}	Pin capacitance	Test freq = 1MHz, T_{amb} = 25 °C		10	pF
Inputs					
V_{IL}	Input LOW voltage, except $\overline{EA}$, SCL, SDA		–0.5	$0.2V_{DD}$–0.15	V
V_{IL1}	Input LOW voltage to $\overline{EA}$		–0.5	$0.2V_{DD}$–0.35	V
V_{IH}	Input HIGH voltage, except XTAL1, RSTIN, SCL, SDA, ADEXS		$0.2V_{DD}$+1.0	V_{DD}+0.5	V
V_{IH1}	Input HIGH voltage, XTAL1, RSTIN, ADEXS		$0.7V_{DD}$+0.1	V_{DD}+0.5	V
I_{IL}	Input current LOW level, Ports 1, 2, 3, 4	V_{IN} = 0.45 V		–75	µA
I_{TL}	Transition current HIGH to LOW, Ports 1, 2, 3, 4	See note 6		–750	µA

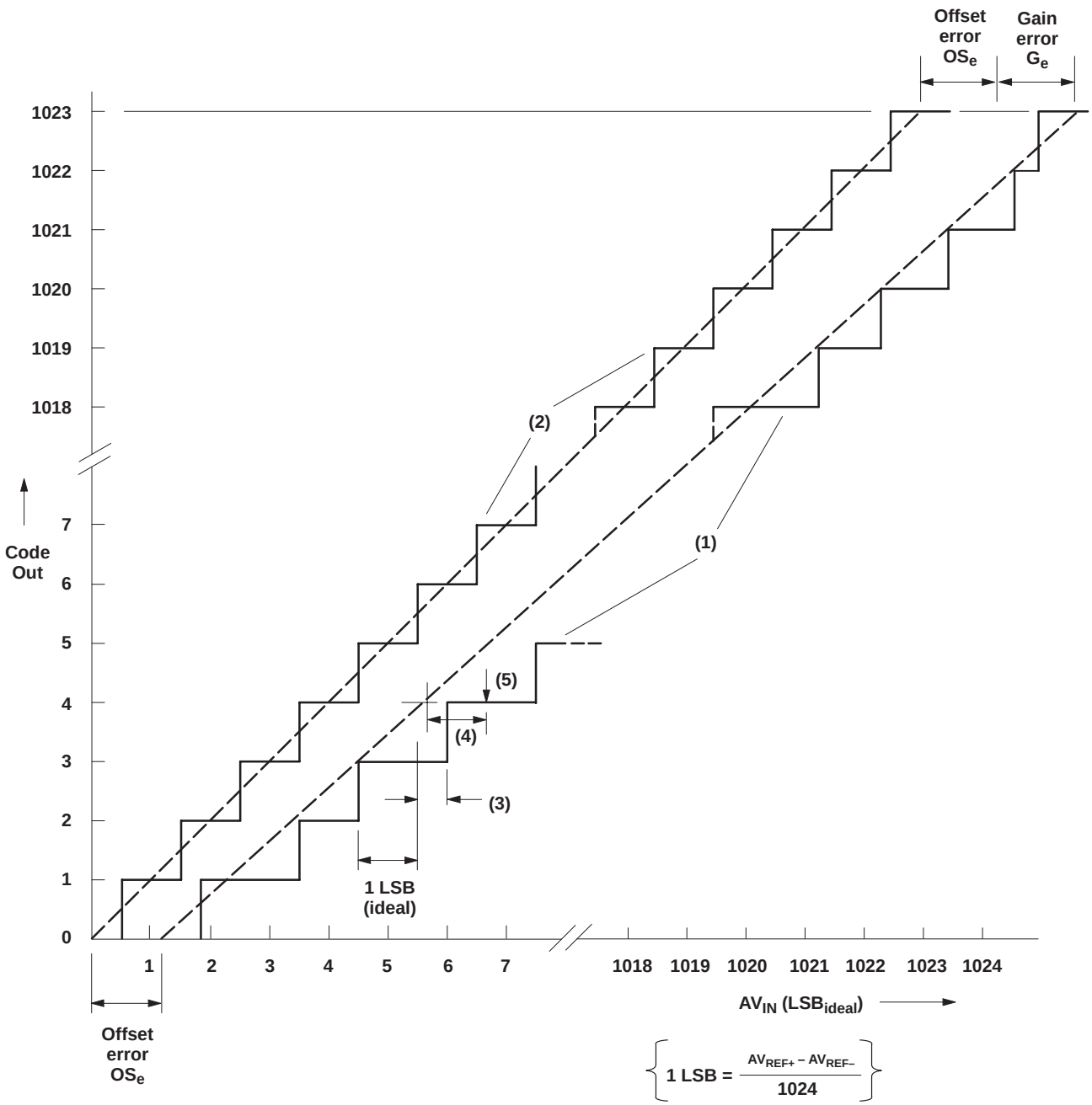
NOTES: See Page 62.

DC ELECTRICAL CHARACTERISTICS ANALOG

AV_{DD} = 5 V (± 10%), AV_{SS} = 0 V, Tamb = 0 °C to +70 °C (P8xC557E4EBx).
AV_{DD} = 5 V (± 10%), AV_{SS} = 0 V, Tamb = –40 °C to +85 °C (P8xC557E4EFx).
All voltages with respect to V_{SS} unless otherwise specified.

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS MIN	LIMITS MAX	UNIT
AV_{DD}	Analog supply voltage	$AV_{DD} = V_{DD} \pm 0.2$ V	4.5	5.5	V
AI_{DD}	Analog supply current operating	Port 5 = 0 to AV_{DD} see notes 1 and 2		1.2	mA
	Analog supply current operating: 32 kHz/PLL operation	Port 5 = 0 to AV_{DD} see note 17, 18		7.2	mA
AI_{ID}	Analog supply current Idle Mode	see notes 1 and 3		70	µA
	Analog supply current Idle Mode: 32 kHz/PLL operation	see note 17		6.0	mA
AI_{PD}	Supply current Power-down mode	2 V < V_{PD} < V_{DDmax} see note 4		50	µA
	Supply current Power-down mode: 32 kHz / PLL operation	V_{DD} = 5.5V see note 17		200	µA
Analog Inputs					
AV_{IN}	Analog input voltage		AV_{SS}–0.2	AV_{DD}+0.2	V
AV_{REF}	Reference voltage: AV_{REF-}		AV_{SS}–0.2		V
	AV_{REF+}			AV_{DD}+0.2	V
R_{REF}	Resistance between AV_{REF+} and AV_{REF-}		10	50	kΩ
C_{IA}	Analog input capacitance			15	pF
DL_e	Differential non-linearity [9, 10, 11]			±1	LSB
IL_e	Integral non-linearity [9, 12]			±2	LSB
OS_e	Offset error [9, 13]			±2	LSB
G_e	Gain error [9, 14]			±0.4	%
A_e	Absolute voltage error [9, 15]			±3	LSB
M_{CTC}	Channel to channel matching			±1	LSB
C_t	Crosstalk between inputs of port 5 [16]	0–100kHz		–60	dB

NOTES: See Page 62.



(1) Example of an actual transfer curve.
(2) The ideal transfer curve.
(3) Differential non-linearity (DL_e).
(4) Integral non-linearity (IL_e).
(5) Center of a step of the actual transfer curve.

Figure 56. ADC Conversion Characteristic

11. AC CHARACTERISTICS

AC ELECTRICAL CHARACTERISTICS

V_{DD} = 5 V ± 10% (EBx), V_{SS} = 0 V, t_{CLK} min = 1/fmax (maximum operating frequency)
V_{DD} = 5 V ± 10% (EFx), V_{SS} = 0 V, t_{CLK} min = 1/fmax (maximum operating frequency)
T_{amb} = 0 °C to +70 °C, t_{CLK} min = 63 ns for P8xC557E4EBx
T_{amb} = –40 °C to +85 °C, t_{CLK} min = 63 ns for P8xC557E4EFx
C1 = 100 pF for Port 0, ALE and $\overline{PSEN}$; C1 = 80 pF for all other outputs unless otherwise specified.

SYMBOL	FIGURE	PARAMETER	12MHz CLOCK		16MHz CLOCK		VARIABLE CLOCK		UNIT
			MIN	MAX	MIN	MAX	MIN	MAX	
$1/t_{CLK}$	60	System clock frequency					3.5	16	MHz
t_{LHLL}	60	ALE pulse width	127		85		$2t_{CLK}$–40		ns
t_{AVLL}	60	Address valid to ALE LOW	43		23		t_{CLK}–40		ns
t_{LLAX}	60	Address hold after ALE LOW	53		33		t_{CLK}–30		ns
t_{LLIV}	60	ALE LOW to valid instruction in		234		150		$4t_{CLK}$–100	ns
t_{LLPL}	60	ALE LOW to $\overline{PSEN}$ LOW	53		33		t_{CLK}–30		ns
t_{PLPH}	60	$\overline{PSEN}$ pulse width	205		143		$3t_{CLK}$–45		ns
t_{PLIV}	60	$\overline{PSEN}$ LOW to valid instruction in		145		83		$3t_{CLK}$–105	ns
t_{PXIX}	60	Input instruction hold after $\overline{PSEN}$	0		0		0		ns
t_{PXIZ}	60	Input instruction float after $\overline{PSEN}$		59		38		t_{CLK}–25	ns
t_{AVIV}	60	Address to valid instruction in		312		208		$5t_{CLK}$–105	ns
t_{PLAZ}	60	$\overline{PSEN}$ LOW to address float		10		10		10	ns
Data Memory									
t_{AVLL}	61, 62	Address valid to ALE LOW	43		23		t_{CLK}–40		ns
t_{LLAX}	61, 62	Address hold after ALE LOW	48		28		t_{CLK}–35		ns
t_{RLRH}	61	$\overline{RD}$ pulse width	400		275		$6t_{CLK}$–100		ns
t_{WLWH}	62	$\overline{WR}$ pulse width	400		275		$6t_{CLK}$–100		ns
t_{RLDV}	61	$\overline{RD}$ LOW to valid data in		252		148		$5t_{CLK}$–165	ns
t_{RHDX}	61	Data hold after $\overline{RD}$	0		0		0		ns
t_{RHDZ}	61	Data float after $\overline{RD}$		97		55		$2t_{CLK}$–70	ns
t_{LLDV}	61	ALE LOW to valid data in		517		350		$8t_{CLK}$–150	ns
t_{AVDV}	61	Address to valid data in		585		398		$9t_{CLK}$–165	ns
t_{LLWL}	61, 62	ALE LOW to $\overline{RD}$ or $\overline{WR}$ LOW	200	300	138	238	$3t_{CLK}$–50	$3t_{CLK}$+50	ns
t_{AVWL}	61, 62	Address valid to $\overline{WR}$ LOW or $\overline{RD}$ LOW	203		120		$4t_{CLK}$–130		ns
t_{QVWX}	62	Data valid to $\overline{WR}$ transition	33		13		t_{CLK}–50		ns
t_{QVWH}	62	Data before $\overline{WR}$	433		288		$7t_{CLK}$–150		ns
t_{WHQX}	62	Data hold after $\overline{WR}$	33		13		t_{CLK}–50		ns
t_{RLAZ}	61	$\overline{RD}$ low to address float		0		0		0	ns
t_{WHLH}	61, 62	$\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH	43	123	23	103	t_{CLK}–40	t_{CLK}+40	ns
UART Timing – Shift Register Mode (Test Conditions: T_{amb} = 0 °C to +70 °C; V_{SS} = 0 V; Load Capacitance = 80pF)									
t_{XLXL}	64	Serial port clock cycle time	1.0		0.75		$12t_{CLK}$		µs
t_{QVXH}	64	Output data setup to clock rising edge	700		492		$10t_{CLK}$–133		ns
t_{XHQX}	64	Output data hold after clock rising edge	50		8		$2t_{CLK}$–117		ns
t_{XHDX}	64	Input data hold after clock rising edge	0		0		0		ns
t_{XHDV}	64	Clock rising edge to input data valid		700		492		$10t_{CLK}$–133	ns

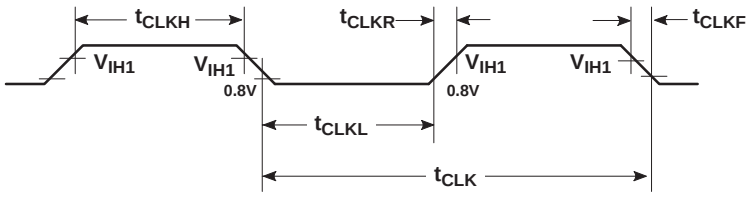
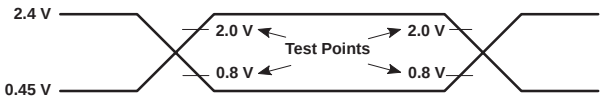
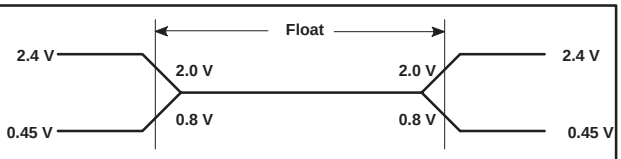


Figure 57. External Clock Drive waveform



NOTE:
AC inputs during testing are driven at 2.4V for a logic 'HIGH' and 0.45V for a logic 'LOW'. Timing measurements are made at 2.0 V for a logic 'HIGH' and 0.8 V for a logic 'LOW'.

Figure 58. AC Testing Input/Output



NOTE:
The float state is defined as the point at which a port 0 pins sinks 3.2 mA or sources 400μA at the voltage test levels.

Figure 59. AC Testing, Float Waveform

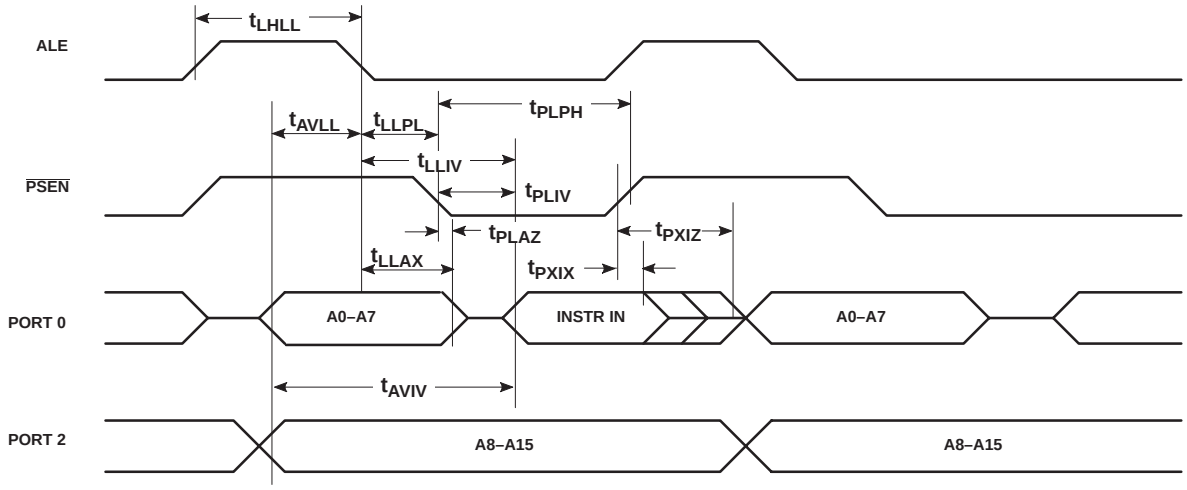


Figure 60. External Program Memory Read Cycle

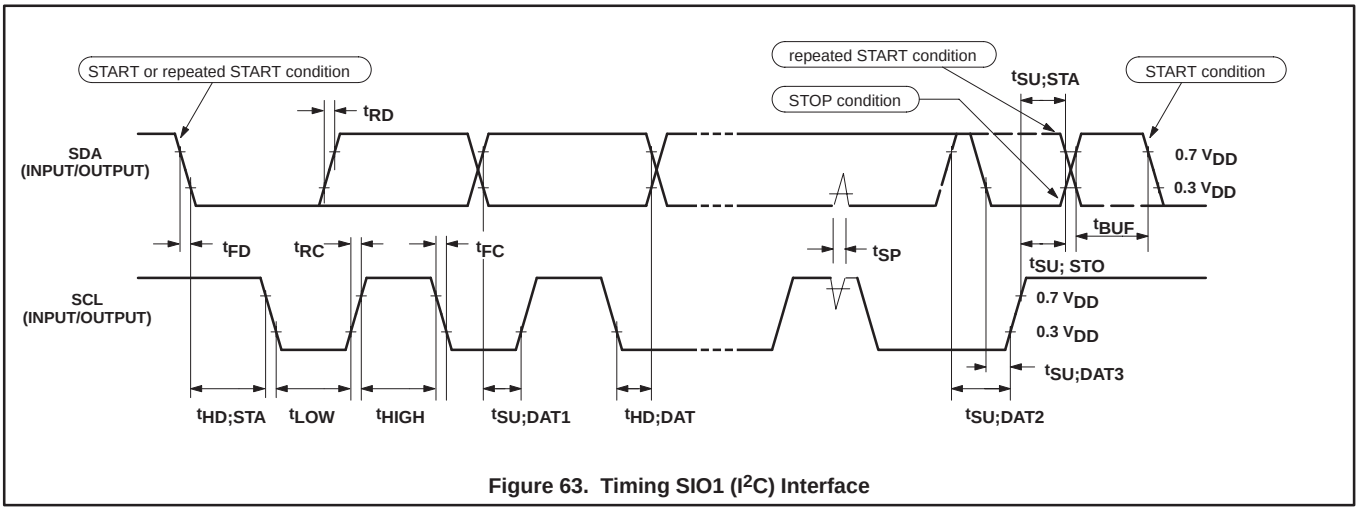


Figure 63. Timing SIO1 (I^2C) Interface

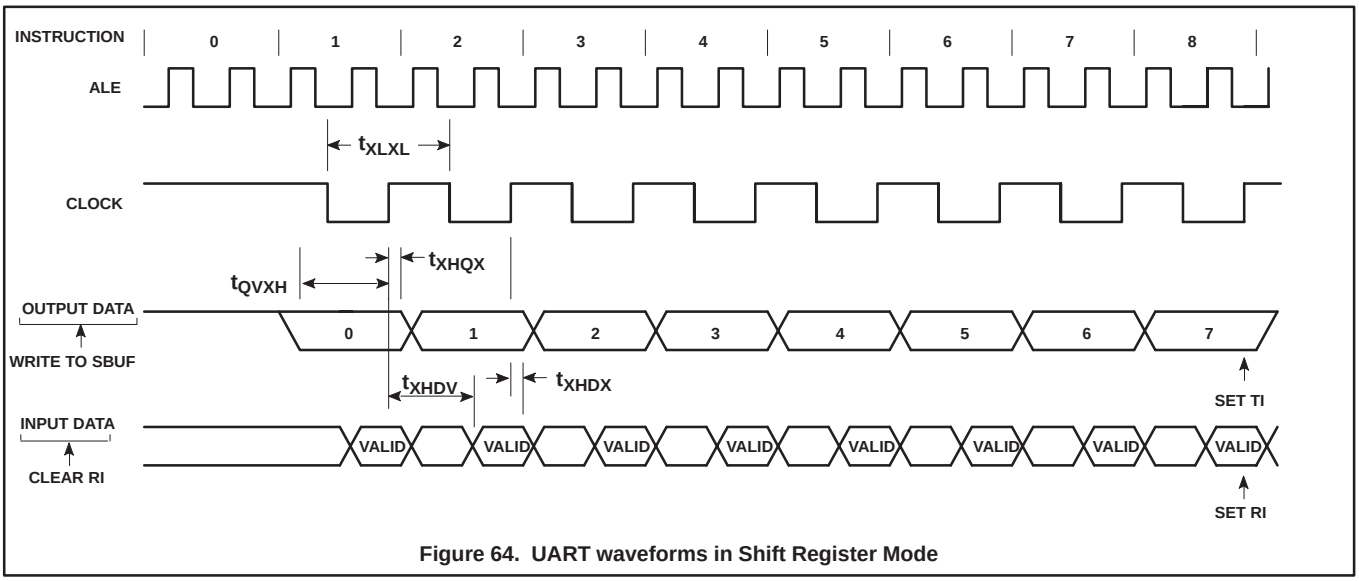
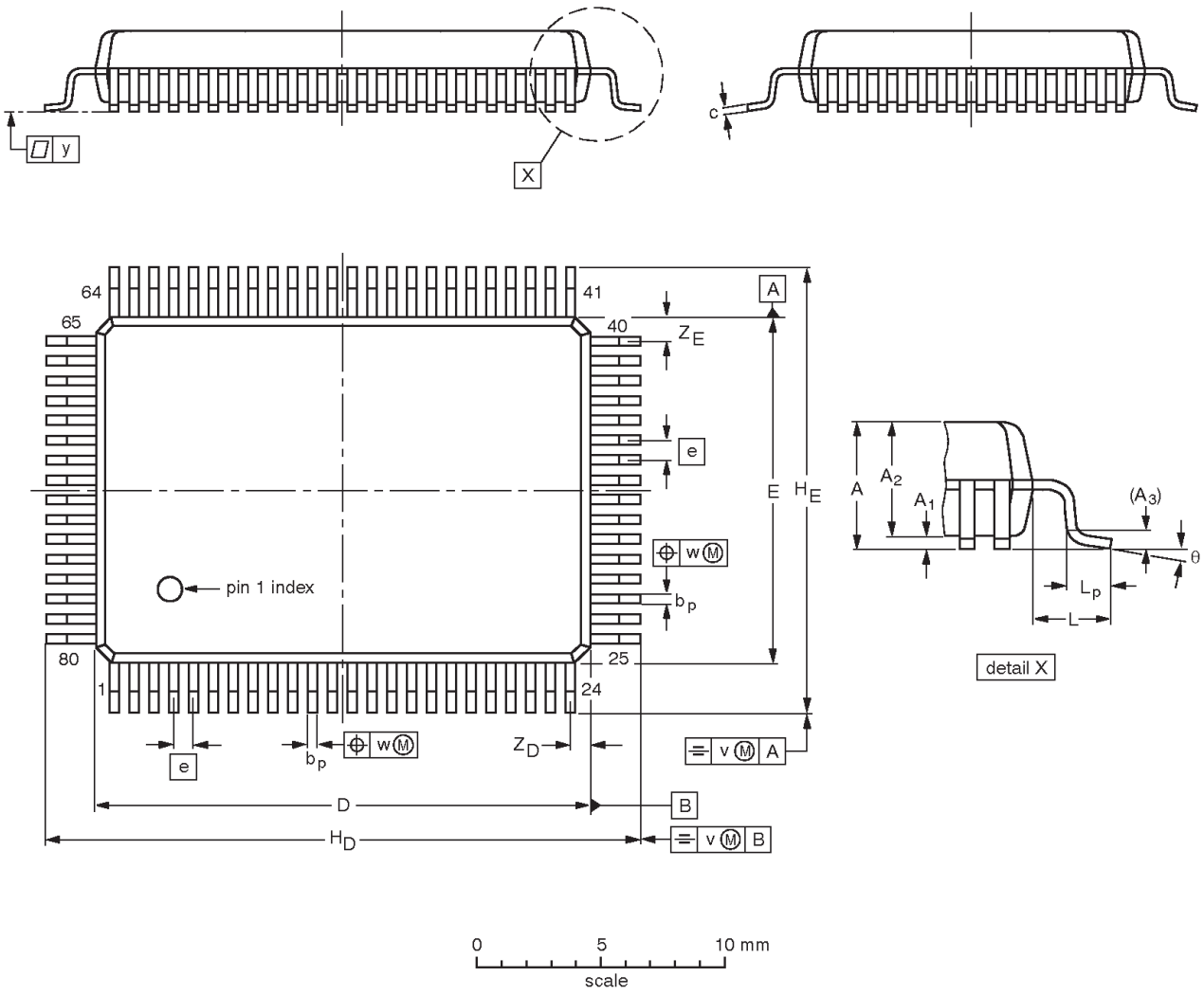


Figure 64. UART waveforms in Shift Register Mode

QFP80: plastic quad flat package;
80 leads (lead length 1.95 mm); body 14 x 20 x 2.7 mm; high stand-off height **SOT318-1**



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_D	H_E	L	L_p	v	w	y	$Z_D^{(1)}$	$Z_E^{(1)}$	θ
mm	3.3	0.36 0.10	2.87 2.57	0.25	0.45 0.30	0.25 0.13	20.1 19.9	14.1 13.9	0.8	24.2 23.6	18.2 17.6	1.95	1.0 0.6	0.2	0.2	0.1	1.0 0.6	1.2 0.8	7^o 0^o

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT318-1						~~95-02-04~~ 97-08-01