VERSA 900: 8K Embedded ISP/IAP Flash, 256B RAM 4 Channel 8-bit ADC, 14 Segments x 4 Commons LCD Driver, 25MHz, MCU

Datasheet Rev 1.1

Overview

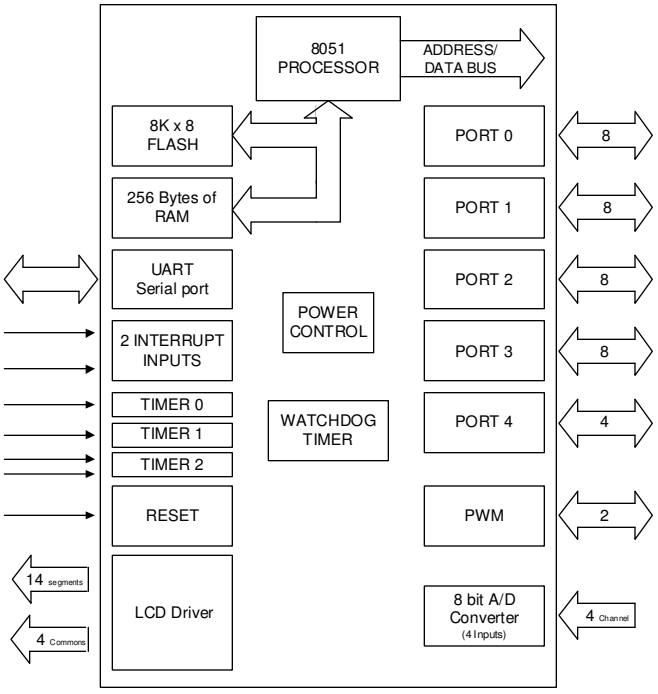
The VRS900 is an 8-bit microcontroller with 8KB of Flash memory and 256 Bytes of RAM based on the architecture of the standard 80C51 microcontroller.

The VRS900 includes extra features such as a 4 Channel 8-bit A/D Converter, 2 PWM outputs and 14 segment x 4 common LCD driver. The VRS900 hardware features make it a versatile and cost-effective controller for a wide range of embedded applications.

The Flash memory can be programmed using a parallel programmer available from Goal Semiconductor. The device will also be supported by 3rd party commercial programmers.

The VRS900 is available in PLCC-44 and QFP-44 packages in the industrial temperature range.

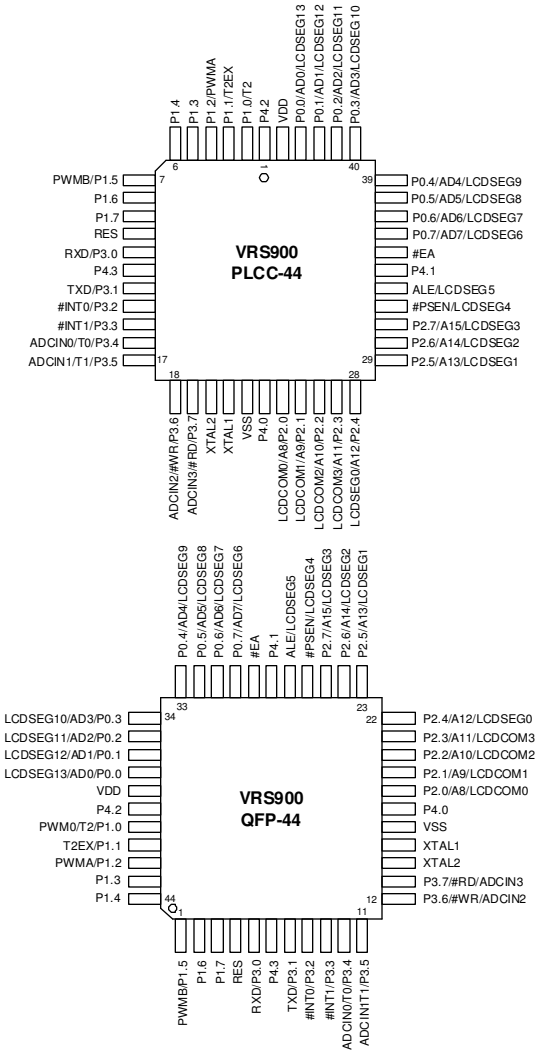
Figure 1: VRS900 Block Diagram



Features

- General 80C51/80C52 pin compatible
- 8KB on-chip Flash memory
- 256 Bytes on-chip data RAM
- Four 8-bit I/O ports and One 4-bit I/O port
- 4 Channel, 8-bit A/D Converter
- LCD Driver: 14 Segments x 4 Commons
- 2-PWM Outputs
- UART serial port
- Three 16-bit Timers/Counters
- Watch Dog Timer
- BCD arithmetic + 8-bit Unsigned Multiply and Division
- Two levels of Interrupt Priority and nested Interrupts
- Power saving modes + Low EMI (ALE disable)
- Code protection function
- Operates at a clock frequency of up to 25MHz
- Industrial Temperature range (-40°C to +85°C)
- 5V version available

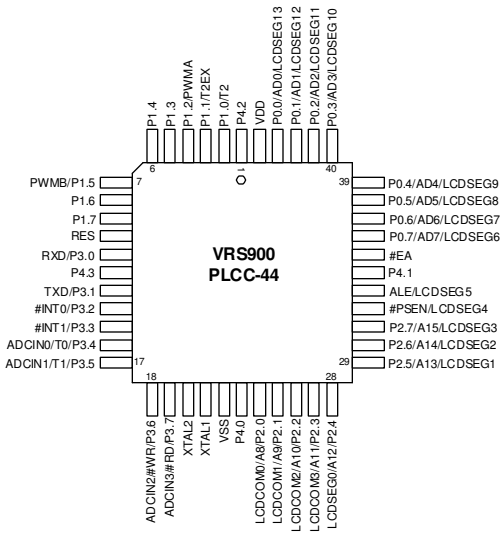
Figure 2: VRS900 PLCC-44 and QFP-44 Pin out Diagrams



Pin Descriptions for PLCC-44

TABLE 1: PIN DESCRIPTIONS FOR PLCC-44

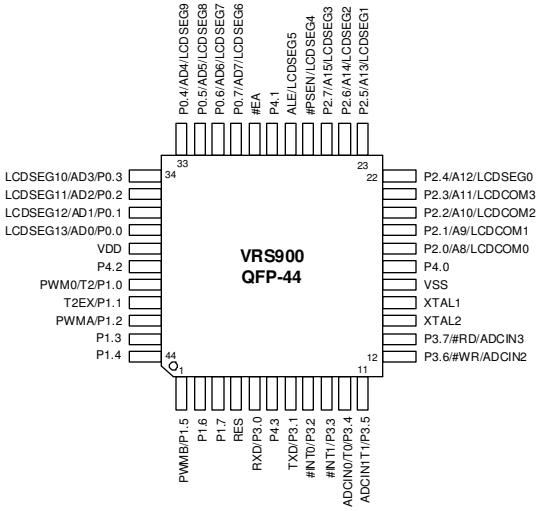
PLCC - 44	Name	I/O	Function
1	P4.2	I/O	Bit 2 of Port 4
2	T2	I	Timer 2 Clock Out
	P1.0	I/O	Bit 0 of Port 1
3	T2EX	I	Timer 2 Control
	P1.1	I/O	Bit 1 of Port 1
4	P1.2	I/O	Bit 2 of Port 1
	PWMA	O	PWM Channel A
5	P1.3	I/O	Bit 3 of Port 1
6	P1.4	I/O	Bit 4 of Port 1
7	PWMB	O	PWM Channel B
	P1.5	I/O	Bit 5 of Port 1
8	P1.6	I/O	Bit 6 of Port 1
9	P1.7	I/O	Bit 7 of Port 1
10	RES	I	Reset
11	RXD	I	Receive Data
	P3.0	I/O	Bit 0 of Port 3
12	P4.3	I/O	Bit 3 of Port 4
13	TXD	O	Transmit Data &
	P3.1	I/O	Bit 1 of Port 3
14	#INT0	I	External Interrupt 0
	P3.2	I/O	Bit 2 of Port 3
15	#INT1	I	External Interrupt 1
	P3.3	I/O	Bit 3 of Port 3
16	ADCIN0	Ain	ADC input 0
	T0	I	Timer 0
	P3.4	I/O	Bit 4 of Port 3
17	ADCIN1	Ain	ADC input 1
	T1	I	Timer 1 & 3
	P3.5	I/O	Bit 5 of Port
18	ADCIN2	Ain	ADC input 2
	#WR	O	Ext. Memory Write
	P3.6	I/O	Bit 6 of Port 3
19	ADCIN3	Ain	ADC input 3
	#RD	O	Ext. Memory Read
	P3.7	I/O	Bit 7 of Port 3
20	XTAL2	O	Oscillator/Crystal Output
21	XTAL1	I	Oscillator/Crystal In
22	VSS	-	Ground
23	P4.0	I/O	Bit 0 of Port 4
24	LCDCOM0	-	LCD Driver Common 0
	P2.0	I/O	Bit 0 of Port 2
	A8	O	Bit 8 of Ext. Memory Address
25	LCDCOM1	-	LCD Driver Common 1
	P2.1	I/O	Bit 1 of Port 2
	A9	O	Bit 9 of Ext. Memory Address
26	LCDCOM2	-	LCD Driver Common 2
	P2.2	I/O	Bit 2 of Port 2
	A10	O	Bit 10 of Ext. Memory Address
27	LCDCOM3	-	LCD Driver Common 3
	P2.3	I/O	Bit 3 of Port 2 &
	A11	O	Bit 11 of Ext. Memory Address
28	LCDSEG0	-	LCD Segment 0
	P2.4	I/O	Bit 4 of Port 2
	A12	O	Bit 12 of Ext. Memory Address
29	LCDSEG1	-	LCD Segment 1
	P2.5	I/O	Bit 5 of Port 2
	A13	O	Bit 13 of External Memory Address
30	LCDSEG2	-	LCD Segment 2
	P2.6	I/O	Bit 6 of Port 2
	A14	O	Bit 14 of External Memory Address
31	LCDSEG3	-	LCD Segment 3
	P2.7	I/O	Bit 7 of Port 2
	A15	O	Bit 15 of External Memory Address
32	LCDSEG4	-	LCD Segment 4
	#PSEN	O	Program Store Enable
33	LCDSEG5	-	LCD Segment 5
	ALE	O	Address Latch Enable
34	P4.1	I/O	Bit 1 of Port 4
35	#EA	I	External Access
36	LCDSEG6	-	LCD Segment 6
	P0.7	I/O	Bit 7 Of Port 0
	AD7	I/O	Data/Address Bit 7 of Ext. Memory
37	LCDSEG7	-	LCD Segment 7
	P0.6	I/O	Bit 6 of Port 0
	AD6	I/O	Data/Address Bit 6 of Ext. Memory
38	LCDSEG8	-	LCD Segment 8
	P0.5	I/O	Bit 5 of Port 0
	AD5	I/O	Data/Address Bit 5 of Ext. Memory
39	LCDSEG9	-	LCD Segment 9
	P0.4	I/O	Bit 4 of Port 0
	AD4	I/O	Data/Address Bit 4 of Ext. Memory
40	LCDSEG10	-	LCD Segment 10
	P0.3	I/O	Bit 3 Of Port 0
	AD3	I/O	Data/Address Bit 3 of Ext. Memory
41	LCDSEG11	-	LCD Segment 11
	P0.2	I/O	Bit 2 of Port 0
	AD2	I/O	Data/Address Bit 2 of Ext. Memory
42	LCDSEG12	-	LCD Segment 12
	P0. 1	I/O	Bit 1 of Port 0 & Data
	AD1	I/O	Address Bit 1 of Ext. Memory
43	LCDSEG13	-	LCD Segment 13
	P0.0	I/O	Bit 0 Of Port 0 & Data
	AD0	I/O	Address Bit 0 of Ext. Memory
44	VDD	-	5V supply



Pin Descriptions for QFP-44

TABLE 2: PIN DESCRIPTIONS FOR PLCC-44

PLCC - 44	Name	I/O	Function
1	PWMB	O	PWM Channel B
	P1.5	I/O	Bit 5 of Port 1
2	P1.6	I/O	Bit 6 of Port 1
3	P1.7	I/O	Bit 7 of Port 1
4	RES	I	Reset
5	RXD	I	Receive Data
	P3.0	I/O	Bit 0 of Port 3
6	P4.3	I/O	Bit 3 of Port 4
7	TXD	O	Transmit Data &
	P3.1	I/O	Bit 1 of Port 3
8	#INT0	I	External Interrupt 0
	P3.2	I/O	Bit 2 of Port 3
9	#INT1	I	External Interrupt 1
	P3.3	I/O	Bit 3 of Port 3
10	ADCIN0	Ain	ADC input 0
	T0	I	Timer 0
	P3.4	I/O	Bit 4 of Port 3
11	ADCIN1	Ain	ADC input 1
	T1	I	Timer 1 & 3
	P3.5	I/O	Bit 5 of Port
12	ADCIN2	Ain	ADC input 2
	#WR	O	Ext. Memory Write
	P3.6	I/O	Bit 6 of Port 3
13	ADCIN3	Ain	ADC input 3
	#RD	O	Ext. Memory Read
	P3.7	I/O	Bit 7 of Port 3
14	XTAL2	O	Oscillator/Crystal Output
15	XTAL1	I	Oscillator/Crystal In
16	VSS	-	Ground
17	P4.0	I/O	Bit 0 of Port 4
18	LCDCOM0	-	LCD Driver Common 0
	P2.0	I/O	Bit 0 of Port 2
	A8	O	Bit 8 of Ext. Memory Address
19	LCDCOM1	-	LCD Driver Common 1
	P2.1	I/O	Bit 1 of Port 2
	A9	O	Bit 9 of Ext. Memory Address
20	LCDCOM2	-	LCD Driver Common 2
	P2.2	I/O	Bit 2 of Port 2
	A10	O	Bit 10 of Ext. Memory Address
21	LCDCOM3	-	LCD Driver Common 3
	P2.3	I/O	Bit 3 of Port 2 &
	A11	O	Bit 11 of Ext. Memory Address
22	LCDSEG0	-	LCD Segment 0
	P2.4	I/O	Bit 4 of Port 2
	A12	O	Bit 12 of Ext. Memory Address
23	LCDSEG1	-	LCD Segment 1
	P2.5	I/O	Bit 5 of Port 2
	A13	O	Bit 13 of External Memory Address
24	LCDSEG2	-	LCD Segment 2
	P2.6	I/O	Bit 6 of Port 2
	A14	O	Bit 14 of External Memory Address
25	LCDSEG3	-	LCD Segment 3
	P2.7	I/O	Bit 7 of Port 2
	A15	O	Bit 15 of External Memory Address
26	LCDSEG4	-	LCD Segment 4
	#PSEN	O	Program Store Enable
27	LCDSEG5	-	LCD Segment 5
	ALE	O	Address Latch Enable
28	P4.1	I/O	Bit 1 of Port 4
29	#EA	I	External Access
30	LCDSEG6	-	LCD Segment 6
	P0.7	I/O	Bit 7 Of Port 0
	AD7	I/O	Data/Address Bit 7 of Ext. Memory
31	LCDSEG7	-	LCD Segment 7
	P0.6	I/O	Bit 6 of Port 0
	AD6	I/O	Data/Address Bit 6 of Ext. Memory
32	LCDSEG8	-	LCD Segment 8
	P0.5	I/O	Bit 5 of Port 0
	AD5	I/O	Data/Address Bit 5 of Ext. Memory
33	LCDSEG9	-	LCD Segment 9
	P0.4	I/O	Bit 4 of Port 0
	AD4	I/O	Data/Address Bit 4 of Ext. Memory
34	LCDSEG10	-	LCD Segment 10
	P0.3	I/O	Bit 3 Of Port 0
	AD3	I/O	Data/Address Bit 3 of Ext. Memory
35	LCDSEG11	-	LCD Segment 11
	P0.2	I/O	Bit 2 of Port 0
	AD2	I/O	Data/Address Bit 2 of Ext. Memory
36	LCDSEG12	-	LCD Segment 12
	P0. 1	I/O	Bit 1 of Port 0 & Data
	AD1	I/O	Address Bit 1 of Ext. Memory
37	LCDSEG13	-	LCD Segment 13
	P0.0	I/O	Bit 0 Of Port 0 & Data
	AD0	I/O	Address Bit 0 of Ext. Memory
38	VDD	-	5V supply
39	P4.2	I/O	Bit 2 of Port 4
40	T2	I	Timer 2 Clock Out
	P1.0	I/O	Bit 0 of Port 1
41	T2EX	I	Timer 2 Control
	P1.1	I/O	Bit 1 of Port 1
42	P1.2	I/O	Bit 2 of Port 1
	PWMA	O	PWM Channel A
43	P1.3	I/O	Bit 3 of Port 1
44	P1.4	I/O	Bit 4 of Port 1



Instruction Set

The following tables describe the instruction set of the VRS900. The instructions are binary code compatible and perform the same functions as the industry standard 8051.

TABLE 3: LEGEND FOR INSTRUCTION SET TABLE

Symbol	Function
A	Accumulator
Rn	Register R0-R7
Direct	Internal register address
@Ri	Internal register pointed to by R0 or R1 (except MOVX)
rel	Two's complement offset byte
bit	Direct bit address
#data	8-bit constant
#data 16	16-bit constant
addr 16	16-bit destination address
addr 11	11-bit destination address

TABLE 4: VRS570/VRS580 INSTRUCTION SET

Mnemonic	Description	Size (bytes)	Instr. Cycles	Op-Code
Arithmetic instructions				
ADD A, Rn	Add register to A	1	1	28h-2Fh
ADD A, direct	Add direct byte to A	2	1	25h
ADD A, @Ri	Add data memory to A	1	1	26h,27h
ADD A, #data	Add immediate to A	2	1	24h
ADDC A, Rn	Add register to A with carry	1	1	38h-3Fh
ADDC A, direct	Add direct byte to A with carry	2	1	35h
ADDC A, @Ri	Add data memory to A with carry	1	1	36h,37h
ADDC A, #data	Add immediate to A with carry	2	1	34h
SUBB A, Rn	Subtract register from A with borrow	1	1	98h-9Fh
SUBB A, direct	Subtract direct byte from A with borrow	2	1	95h
SUBB A, @Ri	Subtract data mem from A with borrow	1	1	96h-97h
SUBB A, #data	Subtract immediate from A with borrow	2	1	94h
INC A	Increment A	1	1	04h
INC Rn	Increment register	1	1	08h-0Fh
INC direct	Increment direct byte	2	1	05h
INC @Ri	Increment data memory	1	1	06h, 07h
DEC A	Decrement A	1	1	14h
DEC Rn	Decrement register	1	1	18h-1Fh
DEC direct	Decrement direct byte	2	1	15h
DEC @Ri	Decrement data memory	1	1	16h,17h
INC DPTR	Increment data pointer	1	2	A3h
MUL AB	Multiply A by B	1	4	A4h
DIV AB	Divide A by B	1	4	84h
DA A	Decimal adjust A	1	1	D4h
Logical Instructions				
ANL A, Rn	AND register to A	1	1	58h-5Fh
ANL A, direct	AND direct byte to A	2	1	55h
ANL A, @Ri	AND data memory to A	1	1	56-57h
ANL A, #data	AND immediate to A	2	1	54h
ANL direct, A	AND A to direct byte	2	1	52h
ANL direct, #data	AND immediate data to direct byte	3	2	53h
ORL A, Rn	OR register to A	1	1	48h-4Fh
ORL A, direct	OR direct byte to A	2	1	45h
ORL A, @Ri	OR data memory to A	1	1	46h,47h
ORL A, #data	OR immediate to A	2	1	44h
ORL direct, A	OR A to direct byte	2	1	42h
ORL direct, #data	OR immediate data to direct byte	3	2	43h
XRL A, Rn	Exclusive-OR register to A	1	1	68h-6Fh
XRL A, direct	Exclusive-OR direct byte to A	2	1	65h
XRL A, @Ri	Exclusive-OR data memory to A	1	1	66h,67h
XRL A, #data	Exclusive-OR immediate to A	2	1	64h
XRL direct, A	Exclusive-OR A to direct byte	2	1	62h
XRL direct, #data	Exclusive-OR immediate to direct byte	3	2	63h
CLR A	Clear A	1	1	E4h
CPL A	Compliment A	1	1	F4h
SWAP A	Swap nibbles of A	1	1	C4h
RL A	Rotate A left	1	1	23h
RLC A	Rotate A left through carry	1	1	33h
RR A	Rotate A right	1	1	03h
RRC A	Rotate A right through carry	1	1	13h
Boolean Instruction				
CLR C	Clear Carry bit	1	1	C3h
CLR bit	Clear bit	2	1	C2h
SETB C	Set Carry bit to 1	1	1	D3h
SETB bit	Set bit to 1	2	1	D2h
CPL C	Complement Carry bit	1	1	B3h
CPL bit	Complement bit	2	1	B2h
ANL C,bit	Logical AND between Carry and bit	2	2	82h
ANL C,#bit	Logical AND between Carry and not bit	2	2	A0h,B0h
ORL C,bit	Logical ORL between Carry and bit	2	2	72h
ORL C,#bit	Logical ORL between Carry and not bit	2	2	A0h
MOV C,bit	Copy bit value into Carry	2	1	A2h
MOV bit,C	Copy Carry value into Bit	2	2	92h
Data Transfer Instructions				
MOV A, Rn	Move register to A	1	1	E8h-Efh
MOV A, direct	Move direct byte to A	2	1	E5h
MOV A, @Ri	Move data memory to A	1	1	E6h,E7h
MOV A, #data	Move immediate to A	2	1	74h
MOV Rn, A	Move A to register	1	1	F8h-FFh
MOV Rn, direct	Move direct byte to register	2	2	A8h-AFh
MOV Rn, #data	Move immediate to register	2	1	78h-7Fh
MOV direct, A	Move A to direct byte	2	1	F5h
MOV direct, Rn	Move register to direct byte	2	2	88h-8Fh
MOV direct, direct	Move direct byte to direct byte	3	2	85h
MOV direct, @Ri	Move data memory to direct byte	2	2	86h,87h
MOV direct, #data	Move immediate to direct byte	3	2	75h
MOV @Ri, A	Move A to data memory	1	1	F6h,F7h
MOV @Ri, direct	Move direct byte to data memory	2	2	A6h,A7h
MOV @Ri, #data	Move immediate to data memory	2	1	76h-77h
MOV DPTR, #data	Move immediate to data pointer	3	2	90h
MOVC A, @A+DPTR	Move code byte relative DPTR to A	1	2	93h
MOVC A, @A+PC	Move code byte relative PC to A	1	2	83h
MOVX A, @Ri	Move external data (A8) to A	1	2	E2h,E3h
MOVX A, @DPTR	Move external data (A16) to A	1	2	E0h
MOVX @Ri, A	Move A to external data (A8)	1	2	F2h,F3h
MOVX @DPTR, A	Move A to external data (A16)	1	2	F0h
PUSH direct	Push direct byte onto stack	2	2	C0h
POP direct	Pop direct byte from stack	2	2	D0h
XCH A, Rn	Exchange A and register	1	1	C8h-CFh
XCH A, direct	Exchange A and direct byte	2	1	C5h
XCH A, @Ri	Exchange A and data memory	1	1	C6h,C7h
XCHD A, @Ri	Exchange A and data memory nibble	1	1	D6h,D7h
Branching Instructions				
ACALL addr 11	Absolute call to subroutine	2	2	11h,31h, 51h,71h, 91h,B1h, D1h,F1h
LCALL addr 16	Long call to subroutine	3	2	12h
RET	Return from subroutine	1	2	22h
RETI	Return from interrupt	1	2	32h
AJMP addr 11	Absolute jump unconditional	2	2	01h,21h, 41h,61h, 81h,A1h, C1h,E1h
LJMP addr 16	Long jump unconditional	3	2	02h
SJMP rel	Short jump (relative address)	2	2	80h
JC rel	Jump on carry = 1	2	2	40h
JNC rel	Jump on carry = 0	2	2	50h
JB bit, rel	Jump on direct bit = 1	3	2	20h
JNB bit, rel	Jump on direct bit = 0	3	2	30h
JBC bit,rel	Jump on direct bit = 1 and clear	3	2	10h
JMP @A+DPTR	Jump indirect relative DPTR	1	2	73h
JZ rel	Jump on accumulator = 0	2	2	60h
JNZ rel	Jump on accumulator 1= 0	2	2	70h
CJNE A, direct, rel	Compare A, direct JNE relative	3	2	B5h
CJNE A, #d, rel	Compare A, immediate JNE relative	3	2	B4h
CJNE Rn, #d, rel	Compare reg, immediate JNE relative	3	2	B8h-BFh
CJNE @Ri, #d, rel	Compare ind, immediate JNE relative	3	2	B6h,B7h
DJNZ Rn, rel	Decrement register, JNZ relative	2	2	D8h-DFh
DJNZ direct, rel	Decrement direct byte, JNZ relative	3	2	D5h
Miscellaneous Instruction				
NOP	No operation	1	1	00h,A5h

Rn: Any of the register R0 to R7
@Ri: Indirect addressing using Register R0 or R1
#data: immediate Data provided with Instruction
#data16: Immediate data included with instruction
bit: address at the bit level
rel: relative address to Program counter from +127 to –128
Addr11: 11-bit address range
Addr16: 16-bit address range
#d: Immediate Data supplied with instruction

Special Function Registers (SFR)

Addresses 80h to FFh of the SFR address space can be accessed in direct addressing mode only. The following table lists the VRS900 Special Function Registers.

TABLE 5: SPECIAL FUNCTION REGISTERS (SFR)

SFR Register	SFR Adrs	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset read Value
P0	80h	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0	11111111b
SP	81h	-	-	-	-	-	-	-	-	00000111b
DPL	82h	-	-	-	-	-	-	-	-	00000000b
DPH	83h	-	-	-	-	-	-	-	-	00000000b
Reserved	84h									10000100b
PCON	87h	SMOD	-	-	-	GF1	GF0	PDOWN	IDLE	00000000b
TCON	88h	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00000000b
TMOD	89h	GATE1	C/T1	T1M1	T1M0	GATE0	C/T0	T0M1	T0M0	00000000b
TL0	8Ah	-	-	-	-	-	-	-	-	00000000b
TL1	8Bh	-	-	-	-	-	-	-	-	00000000b
TH0	8Ch	-	-	-	-	-	-	-	-	00000000b
TH1	8Dh	-	-	-	-	-	-	-	-	00000000b
ADCCTRL	8Eh	ADCEND	ADCCONT	ADCCLK1	ADCCLK0	ADCCH1	ADCCH0	-	-	00000000b
ADCDATA	8Fh	ADCDATA7	ADCDATA6	ADCDATA5	ADCDATA4	ADCDATA3	ADCDATA2	ADCDATA1	ADCDATA0	00000000b
P1	90h	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	11111111b
WDTKEY	97h	WDTKEY7	WDTKEY6	WDTKEY5	WDTKEY4	WDTKEY3	WDTKEY2	WDTKEY1	WDTKEY0	10010111b
SCON	98h	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00000000b
SBUF	99h	-	-	-	-	-	-	-	-	10000000b
P0IOCTRL	9Ah	LCDSEG6	LCDSEG7	LCDSEG8	LCDSEG9	LCDSEG10	LCDSEG11	LCDSEG12	LCDSEG13	00000000b
P1IOCTRL	9Bh		PWMBE				PWMAE			00000000b
P2IOCTRL	9Ch	LCDSEG3	LCDSEG2	LCDSEG1	LCDSEG0	LCDCOM3	LCDCOM2	LCDCOM1	LCDCOM0	00000000b
P3IOCTRL	9Dh	ADCIEN3	ADCIEN2	ADCIEN1	ADCIEN0	-	-	-	-	00000000b
WDTCTRL	9Fh	WDTE	-	WDTCLR	-	-	WDTPS2	WDTPS1	WDTPS0	00000000b
P2	A0h	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	11111111b
PWMACTRL	A3h	-	-	-	-	-	-	PWMACK1	PWMACK0	00000000b
PWMA	A4h	PWMA.4	PWMA.3	PWMA.2	PWMA.1	PWMA.0	NPA.2	NPA.1	NPA.0	00000000b
IEN0	A8h	EA	-	ET2	ES	ET1	EX1	ET0	EX0	00000000b
IEN1	A9h	-	-	-	-	ADCIE	-	-	-	00000000b
IF1	AAh	-	-	-	-	ADCIF	-	-	-	00000000b
PWMD4	ACh	PWMD4.4	PWMD4.3	PWMD4.2	PWMD4.1	PWMD4.0	NP4.2	NP4.1	NP4.0	10101100b
P3	B0h	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0	11111111b
PWMB	B3h	PWMB.7	PWMB.6	PWMB.5	PWMB.4	PWMB.3	PWMB.2	PWMB.1	PWMB.0	00000000b
IP	B8h	-	-	PT2	PS	PT1	PX1	PT0	PX0	00000000b
IP1	B9h	-	-	-	-	ADCIP	-	-	-	00000000b
SYSCON	BFh	WDR	-	-	-	-	-	-	ALEI	00000000b
T2CON	C8h	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2	00000000b
RCAP2L	CAh	-	-	-	-	-	-	-	-	00000000b
RCAP2H	CBh	-	-	-	-	-	-	-	-	00000000b
TL2	CCh	-	-	-	-	-	-	-	-	00000000b
TH2	CDh									00000000b
PSW	D0h	CY	AC	F0	RS1	RS0	OV	-	P	00000001b
PWMBCTRL	D3h	-	-	-	-	-	PWMBRES	PWMBCK1	PWMBCK0	00000000b
P4	D8h	-	-	-	-	P4.3	P4.2	P4.1	P4.0	00001111b
LCDCTRL	DFh	LCDON	LCDEN	LCDPRI	-	-	LCDCLK2	LCDCLK1	LCDCLK0	00000000b
ACC	E0h	-	-	-	-	-	-	-	-	11100000b
LCDBUF0	E1h	SEG0_COM3	SEG0_COM2	SEG0_COM1	SEG0_COM0	SEG1_COM3	SEG1_COM2	SEG1_COM1	SEG1_COM0	11100001b
LCDBUF1	E2h	SEG2_COM3	SEG2_COM2	SEG2_COM1	SEG2_COM0	SEG3_COM3	SEG3_COM2	SEG3_COM1	SEG3_COM0	11100010b
LCDBUF2	E3h	SEG4_COM3	SEG4_COM2	SEG4_COM1	SEG4_COM0	SEG5_COM3	SEG5_COM2	SEG5_COM1	SEG5_COM0	11100011b
LCDBUF3	E4h	SEG6_COM3	SEG6_COM2	SEG6_COM1	SEG6_COM0	SEG7_COM3	SEG7_COM2	SEG7_COM1	SEG7_COM0	11100100b
LCDBUF4	E5h	SEG8_COM3	SEG8_COM2	SEG8_COM1	SEG8_COM0	SEG9_COM3	SEG9_COM2	SEG9_COM1	SEG9_COM0	11100101b
LCDBUF5	E6h	SEG10_COM3	SEG10_COM2	SEG10_COM1	SEG10_COM0	SEG11_COM3	SEG11_COM2	SEG11_COM1	SEG11_COM0	11100110b
LCDBUF6	E7h	SEG12_COM3	SEG12_COM2	SEG12_COM1	SEG12_COM0	SEG13_COM3	SEG13_COM2	SEG13_COM1	SEG13_COM0	1100111b
B	F0h	-	-	-	-	-	-	-	-	00000000b

VRS900 Program Memory

The VRS900 includes 8KB of Flash memory and can be programmed using a parallel programmer.

The System Control Register

The SYSCON register serves to monitor if a reset was used by the Watch Dog Timer overflow and provides the ability to disable the activity on the ALE pin when the VRS900 executes the code from the internal program memory.

TABLE 6: SYSTEM CONTROL REGISTER (SYSCON) – SFR BFH

7	6	5	4	3	2	1	0
WDR	Unused						ALEI

Bit	Mnemonic	Description
7	WDR	This is the Watch Dog Timer reset bit. It will be set to 1 when the reset signal generated by WDT overflows.
6:1	Unused	-
0	ALEI	ALE output inhibit bit, which is used to reduce EMI.

Reduced EMI Function

The VRS900 can be set up to reduce its EMI (electromagnetic interference) emissions by setting bit 0 (ALEI) of the SYSCON register to 1. This function will inhibit the Fosc/6Hz clock signal output to the ALE pin.

Program Status Word Register

The PSW register contains flags that indicate the current state of the processor. These flags may be read or written to by the user.

TABLE 7: PROGRAM STATUS WORD REGISTER (PSW) - SFR DOH

7	6	5	4	3	2	1	0
CY	AC	F0	RS1	RS0	OV	-	P

Bit	Mnemonic	Description
7	CY	Carry Bit
6	AC	Auxiliary Carry Bit from bit 3 to 4.
5	F0	User definer flag
4	RS1	R0-R7 Registers bank select bit 0
3	RS0	R0-R7 Registers bank select bit 1
2	OV	Overflow flag
1	-	-
0	P	Parity flag

RS1	RS0	Active Bank	Address
0	0	0	00h-07h
0	1	1	08h-0Fh
1	0	2	10h-17h
1	1	3	18-1Fh

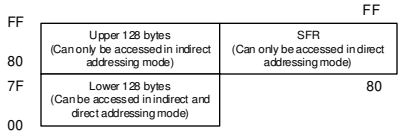
Data Pointer

The VRS900 has one 16-bit data pointer. The DPTR is accessed through two SFR addresses: DPL is located at address 82h and DPH is located at address 83h.

Data Memory

The VRS900 includes 256 Bytes of RAM, which is configured like the internal memory structure of a standard 8052.

FIGURE 1: VRS900 DATA MEMORY STRUCTURE



Lower 128 Bytes (00h to 7Fh, Bank 0 & Bank 1)
The lower 128 Bytes of data memory (from 00h to 7Fh) can be summarized in the following points:

- Address range 00h to 7Fh can be accessed in direct and indirect addressing modes.
- Address range 00h to 1Fh includes R0-R7 registers area.
- Address range 20h to 2Fh is bit addressable.
- Address range 30h to 7Fh is not bit addressable and can be used as general-purpose storage.

Upper 128 Bytes (80h to FFh, Bank 2 & Bank 3)
The upper 128 Bytes of the data memory ranging from 80h to FFh can be accessed using indirect addressing or by using the bank mapping in direct addressing mode.

Stack Pointer

The Stack Pointer is a register located at address 81h of the SFR register area whose value corresponds to the address of the last item that was put on the processor stack. Each time new data is put on the Stack Pointer, the value of the Stack Pointer is incremented.

By default, the Stack Pointer value is 07h, but it is possible to out the processor stack pointer anywhere in the 00h to FFh range of RAM memory.

Each time a function call is performed or an interrupt is serviced, the 16-bit return address (two bytes) is stored on the stack. It is also possible to put data manually on the Stack by using the PUSH and POP functions.

Description of Peripherals Power Control Register

The VRS900 provides two power saving modes: Idle and Power Down, which are controlled by the PDOWN and IDLE bits of the PCON register at address 87h.

TABLE 8: POWER CONTROL REGISTER (PCON) - SFR 87H

7	6:4	3	2	1	0
SMOD		GF1	GF0	PDOWN	IDLE

Bit	Mnemonic	Description
7	SMOD	1: Double the baud rate of the serial port frequency that was generated by Timer 1. 0: Normal serial port baud rate generated by Timer 1.
6		
5		
4		
3	GF1	General Purpose Flag
2	GF0	General Purpose Flag
1	PDOWN	Power down mode control bit
0	IDLE	Idle mode control bit

In Idle mode, the processor's clock is stopped but the peripherals remains active. The content of the RAM, I/O state and SFR registers are maintained. Timer operation is maintained, as well as the external interrupts and UARTs.

The Idle mode is useful for applications in which stopping the processor to save power is required. The processor will be woken up when an external event, triggering an interrupt, occurs. However, because only the processor clock is stopped in Idle Mode, the power saving is likely to be in the order of 50% compared to normal operating mode

In Power Down mode, the oscillator of the VRS900 is stopped. This means that the clock to all peripherals is stopped. The content of the RAM and the SFR registers, however, is maintained. The only way to exit of the Power Down mode is by a hardware Reset.
The Watch Dog Timer is stopped in Power Down.

When the VRS900 is in Power Down, its current consumption drops to about 100uA.

The SMOD bit of the PCON register controls the oscillator divisor applied to the Timer 1 when used as a baud rate generator for the UART. Setting this bit to 1 has an effect of doubling the UART's baud rate generator's frequency.

Input/Output Ports

The VRS900 has 36 bi-directional lines grouped in four 8-bit I/O ports and one 4-bit I/O port. These I/Os can be individually configured as input or output.

Except for the P0 I/Os, which are of the open drain type, each I/O is made of a transistor connected to ground and a dynamic pull-up resistor made of a combination of transistors.

Writing a 0 in a given I/O port bit register will activate the transistor connected to ground, this will bring the I/O to a LOW level.

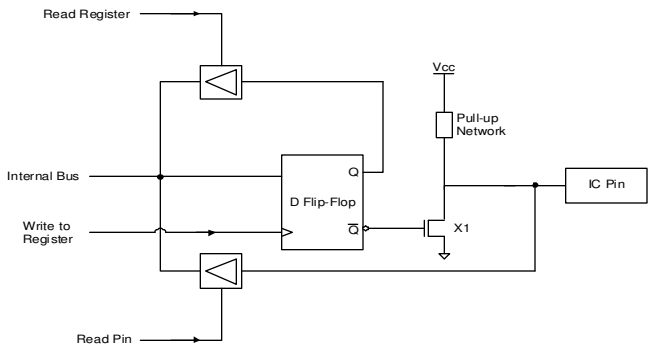
Writing a 1 into a given I/O port bit register de-activates the transistor between the pin and ground. In this case an internal weak pull-up resistor will bring the pin to a HIGH level (except for Port 0 which is open-drain).

To use a given I/O as an input, one must write a 1 into its associated port register bit. By default, upon reset all the I/Os are configured as input. The VRS900 I/O ports are not designed to source current.

Structure of the P1, P2, P3 and P4 Ports

The following figure gives a general idea of the structure of one of the lines of the P1, P2 and P3 ports. For these ports, the output stage is composed of a transistor (X1) and transistors configured as pull-ups. It is important to note that the figure below does not show the intermediary logic that connects the output of the register and the output stage together because this logic varies with the auxiliary function of each port.

FIGURE 2: GENERAL STRUCTURE OF THE OUTPUT STAGE OF P1, P2, P3 AND P4



Each line may be used independently as a logical input or output. When used as an input, as mentioned earlier, the corresponding bit register must be high. This would correspond to #Q=0 in Figure 2.

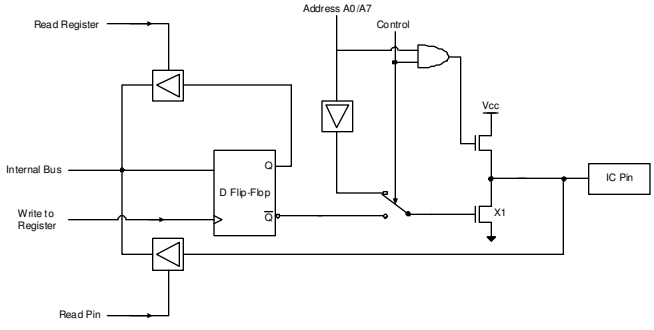
The transistor would be off (open-circuited) and current would flow from the VCC to the pin, generating a logical high at the output. Also, note that if an external device with a logical low value is connected to the pin, the current will flow out of the pin.

The presence of the pull-up resistance even when the I/O's are configured as input means that a small current is likely to flow from the VRS900 I/O's pull-up resistors to the driving circuit when the inputs are driven Low. For this reason, the VRS900 I/O ports P1, P2, P3 and P4 are called "quasi bi-directional".

Structure of Port 0

The internal structure of P0 is shown in the next figure. The auxiliary function of this port requires a particular logic. As opposed to the other ports, P0 is truly bi-directional. In other words, when used as an input, it is considered to be in a floating logical state (high impedance state). This arises from the absence of the internal pull-up resistance. The pull-up resistance is actually replaced by a transistor that is only used when the port functions to access external memory/data bus (EA=0). When used as an I/O port, P0 acts as an open drain port and the use of an external pull-up resistor is likely to be required for some applications.

FIGURE 3: PORT P0'S PARTICULAR STRUCTURE



The alternate function of P0 is to address/Data bus lines AD0 to AD7 when the VRS900 pin EA is held at 0V at reset time or when a MOVX instruction is executed.

The P0 register located at address 80h controls the P0 individual pin direction when used as I/O. The P0 register is bit addressable.

TABLE 9: PORT 0 REGISTER (P0) - SFR 80H

7	6	5	4	3	2	1	0
P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0

Bit	Mnemonic	Description
7	P0.7	For each bit of the P0 register correspond to an I/O line: 0: Output transistor pull the line to 0V 1: The output transistor is blocked so the pull-up brings the I/O to 5V.
6	P0.6	
5	P0.5	
4	P0.4	
3	P0.3	
2	P0.2	
1	P0.1	
0	P0.0	

Port 2

The Port P2 is very similar to Port 1 and Port 3 with the difference that the alternate function of P2 is to act as A8-A15 lines of the address bus when the EA line of the VRS900 is held low at reset time or when MOVX instruction is executed.

Like the P1, P2 and P3 registers, the P2 register is bit addressable.

TABLE 10: PORT 2 REGISTER (P2) - SFR A0H

7	6	5	4	3	2	1	0
P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0

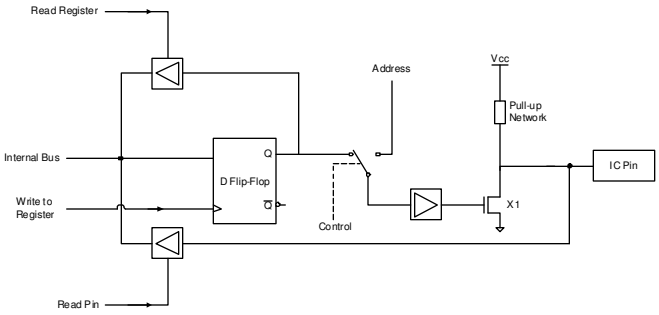
Bit	Mnemonic	Description
7	P2.7	For each bit of the P2 register correspond to an I/O line: 0: Output transistor pull the line to 0V 1: The output transistor is blocked so the pull-up brings the I/O to 5V.
6	P2.6	
5	P2.5	
4	P2.4	
3	P2.3	
2	P2.2	
1	P2.1	
0	P2.0	

Port P0 and P2 as Address and Data Bus

The output stage may receive data from two sources

- The outputs of register P0 or the bus address itself multiplexed with the data bus for P0.
- The outputs of the P2 register or the high part (A8/A15) of the bus address for the P2 port.

FIGURE 4: P2 PORT STRUCTURE



When the ports are used as an address or data bus, the special function registers P0 and P2 are disconnected from the output stage. The 8-bits of the P0 register are forced to 1 and the content of the P2 register remains constant.

Port 1

The P1 register controls the direction of the Port 1 I/O pins. A 1 to the corresponding bit makes the port act as an output presenting a logic 1 to the corresponding I/O pin or renders it possible to use the I/O pin as an input. Writing a 0, activates the output "pull-down" transistor which will force the corresponding I/O line going to a logic Low.

TABLE 11: PORT 1 REGISTER (P2) - SFR 90H

7	6	5	4	3	2	1	0
P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0

Bit	Mnemonic	Description
7	P1.7	For each bit of the P1 register correspond to an I/O line: 0: Output transistor pull the line to 0V 1: The output transistor is blocked so the pull-up bring the I/O to 5V.
6	P1.6	
5	P1.5	
4	P1.4	
3	P1.3	
2	P1.2	
1	P1.1	
0	P1.0	

Auxiliary Port 1 Functions

The Port 1 I/O pins are shared with the PWMA and PWMB outputs, Timer 2 EXT and T2 inputs as shown below:

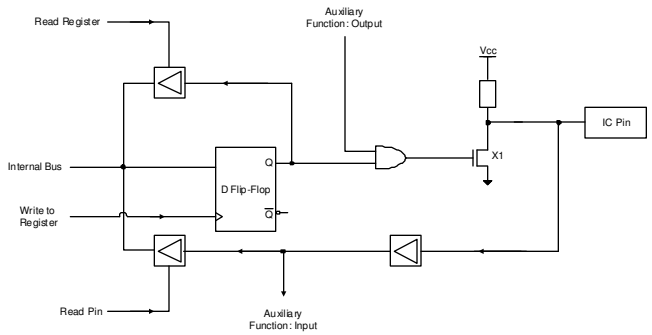
Pin	Mnemonic	Function
P1.0	T2	Timer 2 counter input
P1.1	T2EX	Timer2 Auxiliary input
P1.2	PWMA	PWMA output
P1.3		
P1.4		
P1.5	PWMB	PWMB output
P1.6		
P1.7		

Auxiliary P3 Port Functions

The Port 3 I/O pins are shared with the UART interface, INT0 and INT1 interrupts, Timer 0 and Timer 1 inputs and finally the #WR and #RD lines when external memory access is performed.

To maintain the correct functionality of the line in auxiliary function mode, it is necessary that the Q output of register is held stable at 1. This is done by setting the corresponding P3 but to 1.

FIGURE 5: P3 PORT STRUCTURE



The P3 register controls the P3 pins operation.

TABLE 12: PORT 3 REGISTER (P3) - SFR B0H

7	6	5	4	3	2	1	0
P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0

Bit	Mnemonic	Description
7	P3.7	For each bit of the P3 register correspond to an I/O line: 0: Output transistor pull the line to 0V 1: The output transistor is blocked so the pull-up brings the I/O to 5V. To configure P3 pins as input or use alternate P3 function the corresponding bit must be set to 1.
6	P3.6	
5	P3.5	
4	P3.4	
3	P3.3	
2	P3.2	
1	P3.1	
0	P3.0	

The following table describes the auxiliary function of the Port 3 I/O pins.

TABLE 13: P3 AUXILIARY FUNCTION TABLE

Pin	Mnemonic	Function
P3.0	RXD	Serial Port: Receive data in asynchronous mode. Input and output data in synchronous mode.
P3.1	TXD	Serial Port: Transmit data in asynchronous mode. Output clock value in synchronous mode.
P3.2	$\overline{INT0}$	External Interrupt 0 Timer 0 Control Input
P3.3	$\overline{INT1}$	External Interrupt 1 Timer 1 Control Input
P3.4	T0	Timer 0 Counter Input
P3.5	T1	Timer 1 Counter Input
P3.6	$\overline{WR}$	Write signal for external memory
P3.7	$\overline{RD}$	Read signal for external memory

Port 4

Port 4 has four pins and its SFR, P4 address is located at address 0D8H.

TABLE 14: PORT 4 (P4) - SFR D8H

7	6	5	4	3	2	1	0
Unused				P4.3	P4.2	P4.1	P4.0

Bit	Mnemonic	Description
7	Unused	-
6	Unused	-
5	Unused	-
4	Unused	-
3	P4.3	Used to output the setting to pins P4.3, P4.2, P4.1, P4.0 respectively.
2	P4.2	
1	P4.1	
0	P4.0	

Software Particularities Concerning the Ports

Some instructions allow the user to read the logic state of the output pin, while others allow the user to read the content of the associated port register. These instructions are called *read-modify-write* instructions. A list of these instructions may be found in the table below.

Upon execution of these instructions, the content of the port register (at least 1 bit) is modified. The other read instructions take the present state of the input into account. For example, the instruction ANL P3, #01h obtains the value in the P3 register; performs the desired logic operation with the constant 01h and recopies the result into the P3 register. When users want to take the present state of the inputs into account, they must first read these states and perform an AND operation between the reading and the constant.

b w d

When the port is used as an output, the register contains information on the state of the output pins. Measuring the state of an output directly on the pin is inaccurate because the electrical level depends mostly on the type of charge that is applied to it. The functions shown below take the value of the register rather than that of the pin.

TABLE 15: LIST OF INSTRUCTIONS THAT READ AND MODIFY THE PORT USING REGISTER VALUES

Instruction	Function
ANL	Logical AND ex: ANL P0, A
ORL	Logical OR ex: ORL P2, #01110000B
XRL	Exclusive OR ex: XRL P1, A
JBC	Jump if the bit of the port is set to 0
CPL	Complement one bit of the port
INC	Increment the port register by 1
DEC	Decrement the port register by 1
DJNZ	Decrement by 1 and jump if the result is not equal to 0
MOV P.,C	Copy the held bit C to the port
CLR P.x	Set the port bit to 0
SETB P.x	Set the port bit to 1

Port Operation Timing

Writing to a Port (Output)

When an operation induces a modification of the content in a port register, the new value is placed at the output of the D flip-flop during the last machine cycle that the instruction needed to execute.

Reading a Port (Input)

In order to get sampled, the signal duration present on the I/O inputs must have a duration longer than Fosc/12.

I/O Ports Driving Capability

The maximum allowable continuous current that the device can sink on I/O port is defined by the following

Maximum sink current on one given I/O	10mA
Maximum total sink current for P0	26mA
Maximum total sink current for P1, 2, 3,4	15mA
Maximum total sink current on all I/O	71mA

It is not recommended to exceed the sink current expressed in the above table. Doing so is likely to make the low-level output voltage exceed the device's specification and it is likely to affect the device's reliability.

The VRS900 I/O ports are not designed to source current.

Timers

The VRS900 includes three 16-bit timers: Timer 0, Timer 1 and Timer 2.

The timers can operate in two specific modes:
- Event counting mode
- Timer mode

When operating in event counting mode, the counter is incremented each time an external event, such as a transition in the logical state of the timer input (T0, T1, T2 input), is detected. When operating in Timer mode, the counter is incremented by the microcontroller's system clock (Fosc/12) or by a divided version of it.

Timer 0 and Timer 1

Timers 0 and 1 have four modes of operation. These modes allow the user to change the size of the counting register or to authorize an automatic reload when provided with a specific value. Timer 1 can also be used as a baud rate generator to generate communication frequencies for the serial interface.

Timer 1 and Timer 0 are configured by the TMOD and TCON registers.

TABLE 16: TIMER MODE CONTROL REGISTER (TMOD) – SFR 89H

7	6	5	4	3	2	1	0
GATE1	C/T1	T1M1	T1M0	GATE0	C/T0	T0M1	T0M0

Bit	Mnemonic	Description
7	GATE1	1: Enables external gate control (pin INT1 for Counter 1). When INT1 is high, and TRx bit is set (see TCON register), a counter is incremented every falling edge on the T1IN input pin.
6	C/T1	Selects timer or counter operation (Timer 1). 1 = A counter operation is performed 0 = The corresponding register will function as a timer.
5	T1M1	Selects the operating mode of Timer/Counter 1
4	T1M0	
3	GATE0	If set, enables external gate control (pin INT0 for Counter 0). When INT0 is high, and TRx bit is set (see TCON register), a counter is incremented every falling edge on the T0IN input pin.
2	C/T0	Selects timer or counter operation (Timer 0). 1 = A counter operation is performed 0 = The corresponding register will function as a timer.
1	T0M1	Selects the operating mode of Timer/Counter 0.
0	T0M0	

The table below summarizes the four modes of operation of Timers 0 and 1. The timer operating mode is selected by the bits T1M1/T1M0 and T0M1/T0M0 of the TMOD register.

TABLE 17: TIMER/COUNTER MODE DESCRIPTION SUMMARY

M1	M0	Mode	Function
0	0	Mode 0	13-bit Counter
0	1	Mode 1	16-bit Counter
1	0	Mode 2	8-bit auto-reload Counter/Timer. The reload value is kept in TH0 or TH1, while TL0 or TL1 is incremented every machine cycle. When TLx overflows, the value of THx is copied to TLx.
1	1	Mode 3	If Timer 1 M1 and M0 bits are set to 1, Timer 1 stops.

Timer 0 /Timer 1 Counter / Timer Functions

Timing Function

When Timer 1 or Timer 0 is configured to operate as a Timer, its value is automatically incremented at every machine cycle. Once the Timer value rolls over, a flag is raised and the counter acquires a value of zero. The overflow flags (TF0 and TF1) are located in the TCON register.

The TR0 and TR1 bit of the TCON register gates the corresponding timer operation. In order for the Timer to run, the corresponding TRx bit must be set to 1.

The IT0 and IT1 bits of the TCON register controls the event that will trigger the External Interrupt as follow:

IT0 = 0: The INT0, if enabled, occurs if a Low Level is present on P3.2

IT0 = 1: The INT0, if enabled, occurs if a High to Low transition is detected on P3.2

IT1 = 0: The INT1, if enabled, occurs if a Low Level is present on P3.3

IT1 = 1: The INT1, if enabled, occurs if a High to Low transition is detected on P3.3

The IE0 and IE1 bit of the TCON register are External flags that indicate that a transition has been detected on the INT0 and INT1 interrupt pins respectively.

If the external interrupt is configured as edge sensitive, the corresponding IE0 and IE1 flag is automatically cleared when the corresponding interrupt is serviced.

On the other hand, if the external interrupt is configured as level sensitive, then the corresponding flag must be cleared by the software.

TABLE 18: TIMER 0 AND 1 CONTROL REGISTER (TCON) –SFR 88H

7	6	5	4	3	2	1	0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Bit	Mnemonic	Description
7	TF1	Timer 1 Overflow Flag. Set by hardware on Timer/Counter overflow. Cleared by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine.
6	TR1	Timer 1 Run Control Bit. Set/cleared by software to turn Timer/Counter on or off.
5	TF0	Timer 0 Overflow Flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine.
4	TR0	Timer 0 Run Control Bit. Set/cleared by software to turn Timer/Counter on or off.
3	IE1	Interrupt Edge Flag. Set by hardware when external interrupt edge is detected. Cleared when interrupt processed.
2	IT1	Interrupt 1 Type Control Bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.
1	IE0	Interrupt 0 Edge Flag. Set by hardware when external interrupt edge is detected. Cleared when interrupt processed.
0	IT0	Interrupt 0 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.

Counting Function

When operating as a counter, the Timer's register is incremented at every falling edge of the T0 and T1 signals located at the input of the timer.

When the sampling circuit sees a high immediately followed by a low in the next machine cycle, the counter is incremented. Two machine cycles are required to detect and record an event. In order to be properly sampled, the duration of the event present to the Timer input should be greater than 1/24 of the oscillator frequency.

Timer 0 / Timer 1 Operating Modes

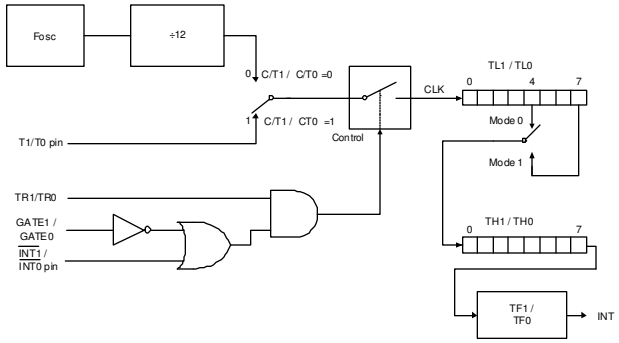
The user may change the operating mode by setting the M1 and M0 bits of the TMOD SFR.

Mode 0

A schematic representation of this mode of operation is presented in the figure below. In Mode 0, the Timer

operates as 13-bit counter the 5 LSB of the counter is made out of the TLx register and the 8 upper bits are made of THx register. When an overflow causes the value of the register to roll over to 0, the TFx interrupt signal goes to 1. The count value is validated as soon as TRx goes to 1 and the GATE bit is 0, or when INTx is 1.

FIGURE 6: TIMER/COUNTER 1 MODE 0: 13-BIT COUNTER



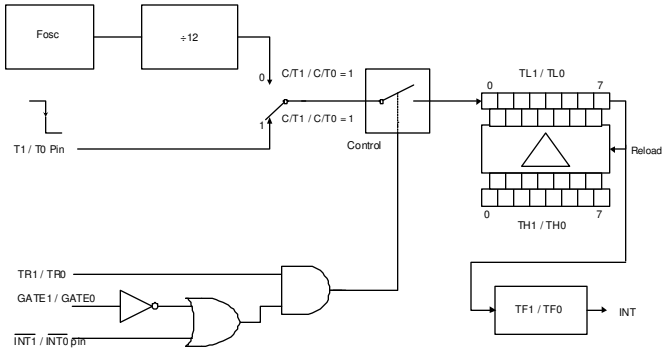
Mode 1

Mode 1 is almost identical to Mode 0. The difference is that in Mode 1, the counter/timer uses the full 16-bits of the Timer.

Mode 2

In this mode, the register of the timer is configured as an 8-bit automatically re-loadable counter/timer. In Mode 2, it is the lower byte TLx that is used as the counter. In the event of a counter overflow, the TFx flag is set to 1 and the value contained in THx, which is preset by software, is reloaded into the TLx counter. The value of THx remains unchanged.

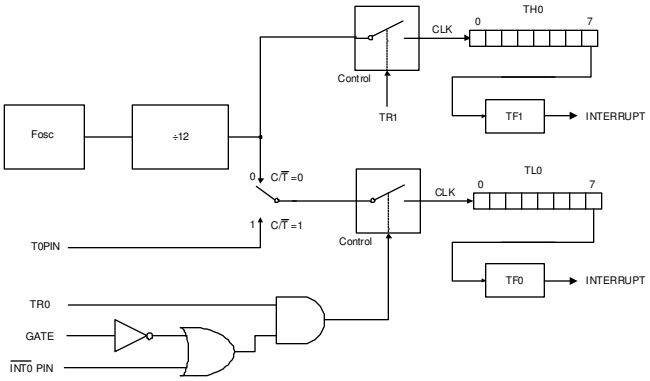
FIGURE 7: TIMER/COUNTER 1 MODE 2: 8-BIT AUTOMATIC RELOAD



Mode 3

In Mode 3, Timer 1 is blocked as if its control bit, TR1, was set to 0. In this mode, Timer 0's registers TL0 and TH0 are configured as two separate 8-bit counters. Also, the TL0 counter uses Timer 0's control bits C/T, GATE, TR0, INT0, TF0 and the TH0 counter is held in Timer Mode (counting machine cycles) and gains control over TR1 and TF1 from Timer 1. At this point, TH0 controls the Timer 1 interrupt.

FIGURE 8: TIMER/COUNTER 0 MODE 3



Timer 2

Timer 2 of the VRS900 is a 16-bit Timer/Counter. Similar to Timers 0 and 1, Timer 2 can operate either as an event counter or as a timer. The user may switch functions by writing to the C/T2 bit located in the T2CON special function register. Timer 2 has three operating modes: "Auto-Load" "Capture", and "Baud Rate Generator". The T2CON SFR configures the modes of operation of Timer 2. The table below describes each bit in the T2CON special function register.

TABLE 19: TIMER 2 CONTROL REGISTER (T2CON) –SFR C8H

7	6	5	4	3	2	1	0
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2

Bit	Mnemonic	Description
7	TF2	Timer 2 Overflow Flag: Set by an overflow of Timer 2 and must be cleared by software. TF2 will not be set when either RCLK =1 or TCLK =1.
6	EXF2	Timer 2 external flag change in state occurs when either a capture or reload is caused by a negative transition on T2EX and EXEN2=1. When Timer 2 is enabled, EXF=1 will cause the CPU to Vector to the Timer 2 interrupt routine. Note that EXF2 must be cleared by software.
5	RCLK	Serial Port Receive Clock Source. 1: Causes Serial Port to use Timer 2 overflow pulses for its receive clock in modes 1 and 3. 0: Causes Timer 1 overflow to be used for the Serial Port receive clock.
4	TCLK	Serial Port Transmit Clock. 1: Causes Serial Port to use Timer 2 overflow pulses for its transmit clock in modes 1 and 3. 0: Causes Timer 1 overflow to be used for the Serial Port transmit clock.
3	EXEN2	Timer 2 External Mode Enable. 1: Allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the Serial Port. 0: Causes Timer 2 to ignore events at T2EX.
2	TR2	Start/Stop Control for Timer 2. 1: Start Timer 2 0: Stop Timer 2
1	C/$\overline{T2}$	Timer or Counter Select (Timer 2) 1: External event counter falling edge triggered. 0: Internal Timer (OSC/12)
0	CP/$\overline{RL2}$	Capture/Reload Select. 1: Capture of Timer 2 value into RCAP2H, RCAP2L is performed if EXEN2=1 and a negative transitions occurs on the T2EX pin. The capture mode requires RCLK and TCLK to be 0. 0: Auto-reload reloads will occur either with Timer 2 overflows or negative transitions at T2EX when EXEN2=1. When either RCK =1 or TCLK =1, this bit is ignored and the timer is forced to auto-reload on Timer 2 overflow.

The possible combinations of control bits that may be used for the mode selection of Timer 2 are shown below:

TABLE 20: TIMER 2 MODE SELECTION BITS

RCLK + TCLK	CP/$\overline{RL2}$	TR2	MODE
0	0	1	16-bit Auto-Reload Mode
0	1	1	16-bit Capture Mode
1	X	1	Baud Rate Generator Mode
X	X	0	Timer 2 stops

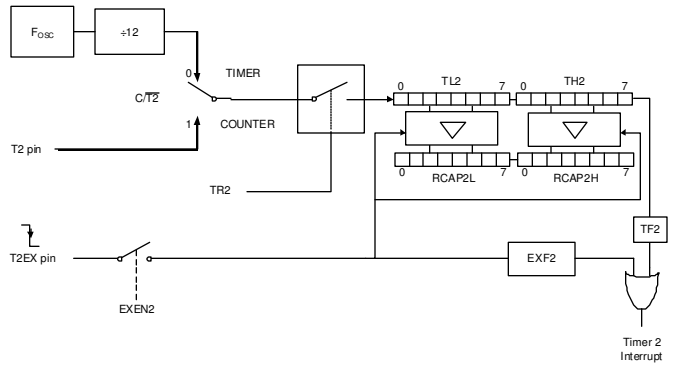
The details of each mode are described below.

Timer 2 Capture Mode

In Capture Mode the EXEN2 bit value defines if the external transition on the T2EX pin will be able to trigger the capture of the timer value.

When EXEN2 = 0, Timer 2 acts as a 16-bit timer or counter, which, upon overflowing, will set bit TF2 (Timer 2 overflow bit). This overflow can be used to generate an interrupt.

FIGURE 9: TIMER 2 IN CAPTURE MODE



When EXEN2 = 1, the above still applies. In addition, it is possible to allow a 1 to 0 transition at the T2EX input to cause the current value stored in the Timer 2 registers (TL2 and TH2) to be captured into the RCAP2L and RCAP2H registers. Furthermore, the transition at T2EX causes bit EXF2 in T2CON to be set, and EXF2, like TF2, can generate an interrupt. Note that both EXF2 and TF2 share the same interrupt vector.

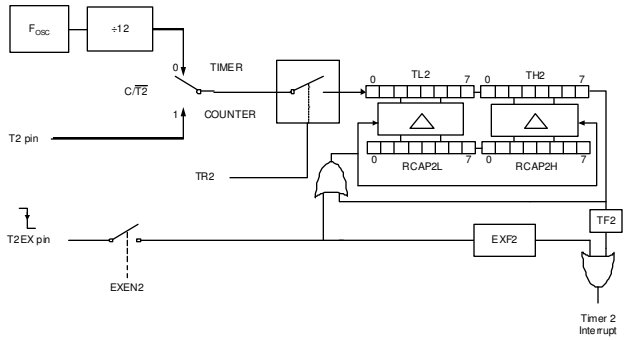
Timer 2 Auto-Reload Mode

In this mode, there are also two options. The user may choose either option by writing to bit EXEN2 in T2CON.

If EXEN2 = 0, when Timer 2 rolls over, it not only sets TF2, but also causes the Timer 2 registers to be reloaded with the 16-bit value in the RCAP2L and RCAP2H registers previously initialised. In this mode, Timer 2 can be used as a baud rate generator source for the serial port.

If EXEN2=1, then Timer 2 still performs the above operation, but a 1 to 0 transition at the external T2EX input will also trigger an anticipated reload of the Timer 2 with the value stored in RCAP2L, RCAP2H and set EXF2.

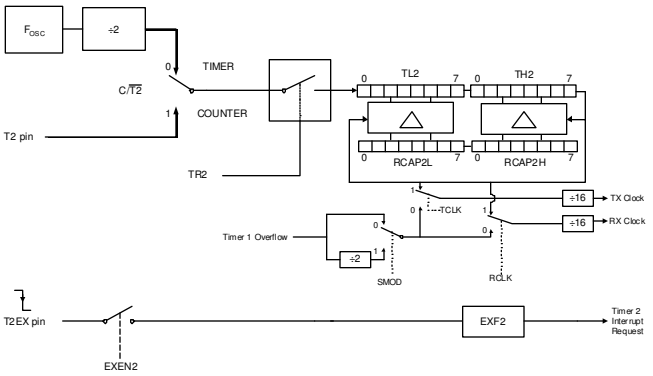
FIGURE 10: TIMER 2 IN AUTO-RELOAD MODE



Timer 2 Baud Rate Generator Mode

Timer 2 can be used for UART Baud Rate. This mode is activated when RCLK is set to 1 and/or TCLK is set to 1. This mode will be described in the serial port section.

FIGURE 11: TIMER 2 IN AUTOMATIC BAUD GENERATOR MODE



UART Serial Port

The serial port on the VRS900 can operate in full duplex; in other words, it can transmit and receive data simultaneously. It is possible to have different communication speeds for Transmission and Reception by assigning one timer for transmission and one timer for reception.

The VRS900 serial port includes a double buffering feature on reception buffer, which allows starting reception of a byte even if the one previously received has not been retrieved from the receive register by the processor. However, if the first byte has still not been read by the time reception of the second byte is complete, the byte present in the receive buffer will be lost.

One SFR register, SBUF, gives access to the Transmit and Receive registers of the serial port. When a read operation is performed on the SBUF register, it will access the receive register. When a write operation is performed on the SBUF, the transmit register will be loaded with the value.

UART Control Register

The serial port control register, SCON contains the 9th data bit for transmit and receive (TB8 and RB8) and all the mode selection bits. SCON also contains the serial port interrupt bits (TI and RI).

TABLE 21: SERIAL PORT CONTROL REGISTER (SCON) – SFR 98H

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

Bit	Mnemonic	Description
7	SM0	Bit to select mode of operation (see table below)
6	SM1	Bit to select mode of operation (see table below)
5	SM2	Multiprocessor communication is possible in Modes 2 and 3. In Modes 2 or 3 if SM2 is set to 1, RI will not be activated if the received 9th data bit (RB8) is 0. In Mode 1, if SM2 = 1 then RI will not be activated if a valid stop bit was not received.
4	REN	Serial Reception Enable Bit This bit must be set by software and cleared by software. 1: Serial reception enabled 0: Serial reception disabled
3	TB8	9th data bit transmitted in Modes 2 and 3 This bit must be set by software and cleared by software.
2	RB8	9th data bit received in Modes 2 and 3. In Mode 1, if SM2 = 0, RB8 is the stop bit that was received. In Mode 0, this bit is not used. This bit must be cleared by software.
1	TI	Transmission Interrupt flag. Automatically set to 1 when: • The 8th bit has been sent in Mode 0. • Automatically set to 1 when the stop bit has been sent in the other modes. This bit must be cleared by software.
0	RI	Reception Interrupt flag Automatically set to 1 when: • The 8th bit has been received in Mode 0. • Automatically set to 1 when the stop bit has been sent in the other modes (see SM2 exception). This bit must be cleared by software.

TABLE 22: SERIAL PORT MODES OF OPERATION

SM0	SM1	Mode	Description	Baud Rate
0	0	0	Shift Register	$F_{osc}/12$
0	1	1	8-bit UART	Variable
1	0	2	9-bit UART	$F_{osc}/64$ or $F_{osc}/32$
1	1	3	9-bit UART	Variable

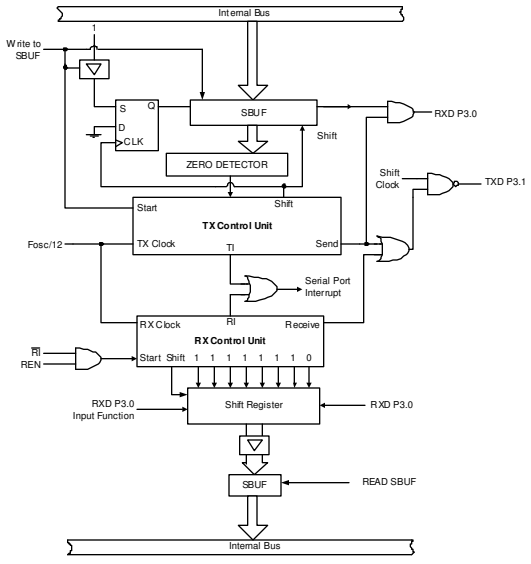
UART Operating Modes

The VRS900's serial port can operate in four different modes. In all four modes, a transmission is initiated by an instruction that uses the SBUF register as a destination register. In Mode 0, reception is initiated by setting RI to 0 and REN to 1. An incoming start bit initiates reception in the other modes provided that REN is set to 1. The following paragraphs describe the four modes.

UART Operation in Mode 0

In this mode, the serial data exits and enters through the RXD pin. TXD is used to output the shift clock. The signal is composed of 8 data bits starting with the LSB. The baud rate in this mode is 1/12 the oscillator frequency.

FIGURE 12: SERIAL PORT MODE 0 BLOCK DIAGRAM



UART Transmission in Mode 0

Any instruction that uses SBUF as a destination register may initiate a transmission. The "write to SBUF" signal also loads a 1 into the 9th position of the transmit shift register and tells the TX control block to begin a transmission. The internal timing is such that one full machine cycle will elapse between a write to SBUF instruction and the activation of SEND.

The SEND signal enables the output of the shift register to the alternate output function line of P3.0 and enables SHIFT CLOCK to the alternate output function line of P3.1.

At every machine cycle in which SEND is active, the contents of the transmit shift register are shifted to the right by one position.

Zeros come in from the left as data bits shift out to the right. The TX control block sends its final shift and de-activates SEND while setting T1 after one condition is fulfilled: When the MSB of the data byte is at the output position of the shift register; the 1 that was initially loaded into the 9th position is just to the left of the MSB; and all positions to the left of that contain zeros. Once these conditions are met, the de-activation of SEND and the setting of T1 occur at T1 of the 10th machine cycle after the "write to SBUF" pulse.

UART Reception in Mode 0

When REN and R1 are set to 1 and 0 respectively, reception is initiated. The bits 11111110 are written to the receive shift register at the end of the next machine cycle by the RX control unit. In the following phase, the RX control unit will activate RECEIVE.

The contents of the receive shift register are shifted one position to the left at the end of every machine cycle during which RECEIVE is active. The value that comes in from the right is the value that was sampled at the P3.0 pin.

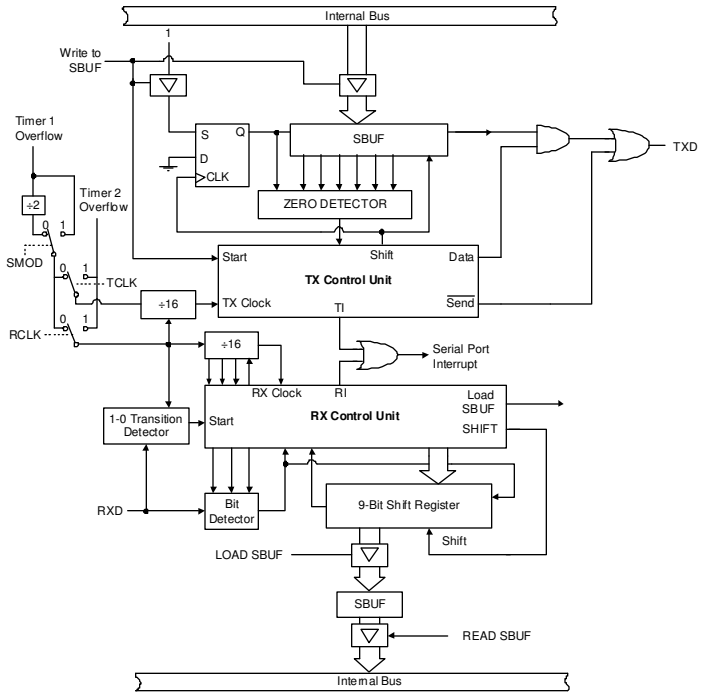
1's are shifted out to the left as data bits are shifted in from the right. The RX control block is flagged to do one last shift and load SBUF when the 0 that was initially loaded into the rightmost position arrives at the leftmost position in the shift register.

UART Operation in Mode 1

For an operation in Mode 1, 10 bits are transmitted (through TXD) or received (through RXD). The transactions are composed of: a Start bit (Low); 8 data bits (LSB first) and one Stop bit (high). The reception is completed once the Stop bit sets the RB8 flag in the SCON register. Either Timer 1 or Timer 2 controls the baud rate in this mode.

The following diagram shows the serial port structure when configured in Mode 1.

FIGURE 13: SERIAL PORT MODE 1 AND 3 BLOCK DIAGRAM



UART Transmission in Mode 1

Transmission is initiated by any instruction that makes use of SBUF as a destination register. The 9th bit position of the transmit shift register is loaded by the "write to SBUF" signal. This event also flags the TX Control Unit that a transmission has been requested.

It is after the next rollover in the divide-by-16 counter when transmission actually begins. It follows that the bit times are synchronized to the divide-by-16 counter and not to the "write to SBUF" signal.

When a transmission begins, it places the start bit at TXD. Data transmission is activated one bit time later. This activation enables the output bit of the transmit shift register to TXD. One bit time after that, the first shift pulse occurs.
In this mode, zeros are clocked in from the left as data bits are shifted out to the right. When the most significant bit of the data byte is at the output position of the shift register, the 1 that was initially loaded into the 9th position is to the immediate left of the MSB, and all positions to the left of that contain zeros. This condition flags the TX Control Unit to shift one more time.

UART Reception in Mode 1

A one to zero transition at pin RXD will initiate reception. For this reason, RXD is sampled at a rate of 16 multiplied by the baud rate that has been established. When a transition is detected, 1FFh is written into the input shift register and the divide-by-16 counter is immediately reset. The divide-by-16 counter is reset in order to align its rollovers with the boundaries of the incoming bit times.
In total, there are 16 states in the counter. During the 7th, 8th and 9th counter states of each bit time; the bit detector samples the value of RXD. The accepted value is the value that was seen in at least two of the three samples. The purpose of doing this is for noise rejection. If the value accepted during the first bit time is not zero, the receive circuits are reset and the unit goes back to searching for another one to zero transition. All false start bits are rejected by doing this. If the start bit is valid, it is shifted into the input shift register, and the reception of the rest of the frame will proceed.
For a receive operation, the data bits come in from the right as 1's shift out on the left. As soon as the start bit arrives at the leftmost position in the shift register, (9-

bit register), it tells the UART's receive controller block to perform one last shift operation: to set RI and to load SBUF and RB8. The signal to load SBUF and RB8, and to set RI, will be generated if, and only if, the following conditions are met at the time the final shift pulse is generated:

- Either SM2 = 0 or the received stop bit = 1
- RI = 0

If both conditions are met, the stop bit goes into RB8, the 8 data bits go into SBUF, and RI is activated. If one of these conditions is not met, the received frame is completely lost. At this time, whether the above conditions are met or not, the unit goes back to searching for a one to zero transition in RXD.

UART Operation in Mode 2

In Mode 2 a total of 11 bits are transmitted (through TXD) or received (through RXD). The transactions are composed of: a Start bit (Low), 8 data bits (LSB first), a programmable 9th data bit, and one Stop bit (High).

For transmission, the 9th data bit comes from the TB8 bit of SCON. For example, the parity bit P in the PSW could be moved into TB8.

In the case of receive, the 9th data bit is automatically written into RB8 of the SCON register.

In Mode 2, the baud rate is programmable to either 1/32 or 1/64 the oscillator frequency.

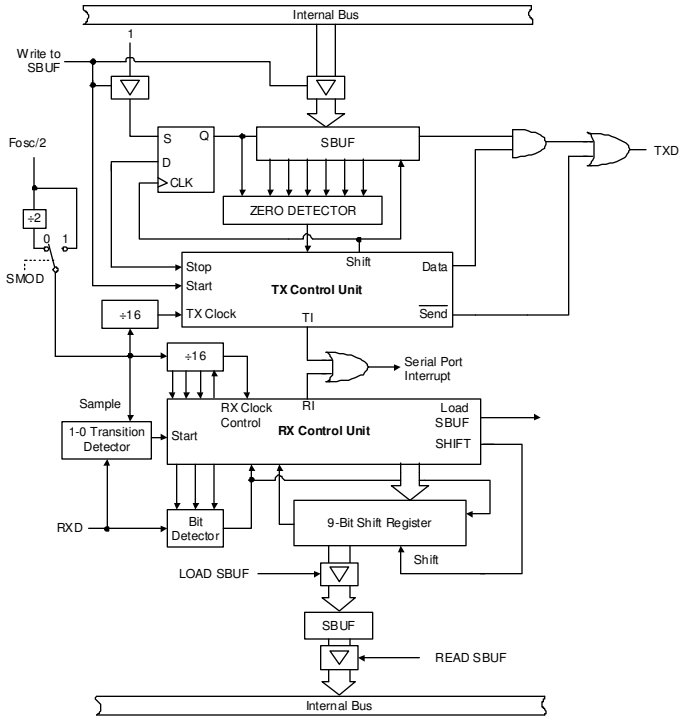


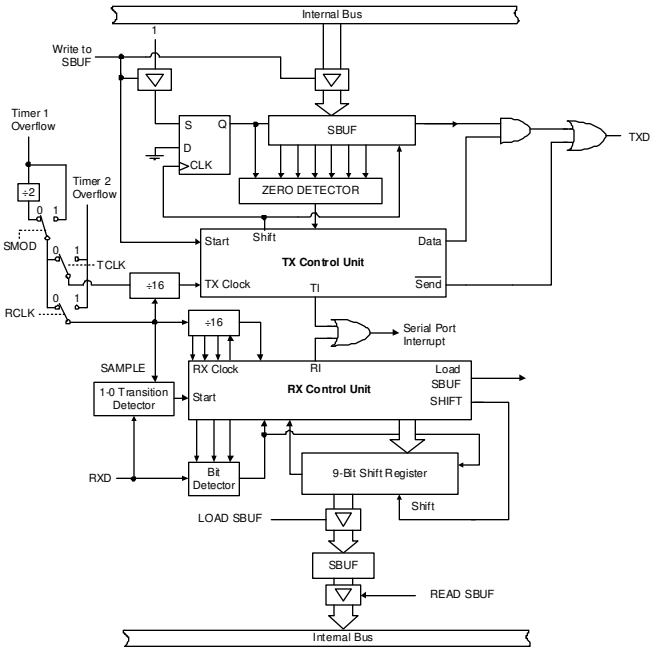
FIGURE 14: SERIAL PORT MODE 2 BLOCK DIAGRAM

UART Operation in Mode 3

In Mode 3, 11 bits are transmitted (through TXD) or received (through RXD). The transactions are composed of: a Start bit (Low), 8 data bits (LSB first), a programmable 9th data bit, and one Stop bit (High).

Mode 3 is identical to Mode 2 in all respects but one: the baud rate. Either Timer 1 or Timer 2 generates the baud rate in Mode 3.

FIGURE 15: SERIAL PORT MODE 3 BLOCK DIAGRAM



UART in Mode 2 and 3: Additional Information

As mentioned earlier, for an operation in Modes 2 and 3, 11 bits are transmitted (through TXD) or received (through RXD). The signal comprises: a logical low Start bit, 8 data bits (LSB first), a programmable 9th data bit, and one logical high Stop bit.
On transmit, (TB8 in SCON) can be assigned the value of 0 or 1. On receive; the 9th data bit goes into RB8 in SCON. The baud rate is programmable to either 1/32 or 1/64 the oscillator frequency in Mode 2. Mode 3 may have a variable baud rate generated from either Timer 1 or Timer 2 depending on the states of TCLK and RCLK.

UART Transmission in Mode 2 and Mode 3

The transmission is initiated by any instruction that makes use of SBUF as the destination register. The 9th bit position of the transmit shift register is loaded by the "write to SBUF" signal. This event also informs the UART transmission control unit that a transmission has been requested. After the next rollover in the divide-by-16 counter, a transmission actually starts at the beginning of the machine cycle. It follows that the bit times are synchronized to the divide-by-16 counter and not to the "write to SBUF" signal, as in the previous mode.

Transmissions begin when the SEND signal is activated, which places the Start bit on TXD pin. Data is activated one bit time later. This activation enables the output bit of the transmit shift register to the TXD pin. The first shift pulse occurs one bit time after that.

The first shift clocks a Stop bit (1) into the 9th bit position of the shift register on TXD. Thereafter, only zeros are clocked in. Thus, as data bits shift out to the right, zeros are clocked in from the left. When TB8 is at the output position of the shift register, the stop bit is just to the left of TB8, and all positions to the left of that contain zeros. This condition signals to the TX control unit to shift one more time and set TI, while deactivating SEND. This occurs at the 11th divide-by-16 rollover after "write to SBUF".

UART Reception in Mode 2 and Mode 3

One to zero transitions on the RXD pin initiate reception. It is for this reason that RXD is sampled at a rate of 16 multiplied by the baud rate that has been established.
When a transition is detected, the 1FFh is written into the input shift register and the divide-by-16 counter is immediately reset.

During the 7^{th}, 8^{th} and 9^{th} counter states of each bit time; the bit detector samples the value of RXD. The accepted value is the value that was seen in at least two of the three samples. If the value accepted during the first bit time is not zero, the receive circuits are reset and the unit goes back to searching for another one to zero transition. If the start bit is valid, it is shifted into the input shift register, and the reception of the rest of the frame will proceed.

For a receive operation, the data bits come in from the right as 1's shift out on the left. As soon as the start bit arrives at the leftmost position in the shift register (9-bit register), it tells the RX control block to do one more shift, to set RI, and to load SBUF and RB8. The signal to set RI and to load SBUF and RB8 will be generated if, and only if, the following conditions are satisfied at the instance when the final shift pulse is generated:

- Either SM2 = 0 or the received 9^{th} bit equal 1
- RI = 0

If both conditions are met, the 9^{th} data bit received goes into RB8, and the first 8 data bits go into SBUF. If one of these conditions is not met, the received frame is completely lost. One bit time later, whether the above conditions are met or not, the unit goes back to searching for a one to zero transition at the RXD input.
Please note that the value of the received stop bit is unrelated to SBUF, RB8 or RI.

UART Baud Rates

In Mode 0, the baud rate is fixed and can be represented by the following formula:

$$\text{Mode 0 Baud Rate} = \frac{\text{Oscillator Frequency}}{12}$$

In Mode 2, the baud rate depends on the value of the SMOD bit in the PCON SFR. From the formula below, we can see that if SMOD = 0 (which is the value on reset), the baud rate is 1/32 the oscillator frequency.

$$\text{Mode 2 Baud Rate} = \frac{2^{SMOD} \times (\text{Oscillator Frequency})}{64}$$

The Timer 1 and/or Timer 2 overflow rate determines the baud rates in Modes 1 and 3.

Generating UART Baud Rate with Timer 1

When Timer 1 functions as a baud rate generator, the baud rate in Modes 1 and 3 are determined by the Timer 1 overflow rate.

$$\text{Mode 1,3 Baud Rate} = \frac{2^{SMOD} \times \text{Timer 1 Overflow Rate}}{32}$$

Timer 1 must be configured as an 8-bit timer (TL1) with auto-reload with TH1 value when an overflow occurs (Mode 2). In this application, the Timer 1 interrupt should be disabled.

The two following formulas can be used to calculate the baud rate and the reload value to put in the TH1 register.

$$\text{Mode 1,3 Baud Rate} = \frac{2^{SMOD} \times Fosc}{32 \times 12(256 - TH1)}$$

The value to put into the TH1 register is defined by the following formula:

$$TH1 = 256 - \frac{2^{SMOD} \times Fosc}{32 \times 12 \times (Baud\ Rate)}$$

Generating UART Baud Rates with Timer 2

Timer 2 is often preferred to generate the baud rate, as it can be easily configured to operate as a 16-bit timer with auto-reload. This allows for much better resolution than using Timer 1 in 8-bit auto-reload mode.

The baud rate using Timer 2 is defined as:

$$Mode\ 1,3\ Baud\ Rate = \frac{Timer\ 2\ Overflow\ Rate}{16}$$

The timer can be configured as either a timer or a counter in any of its 3 running modes. In most typical applications, it is configured as a timer (C/T2 is set to 0).

To make the Timer 2 operate as a baud rate generator, the TCLK and RCLK bits of the T2CON register must be set to 1.

The baud rate generator mode is similar to the auto-reload mode in that an overflow in TH2 causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L, which are preset by software. However, when Timer 2 is configured as a baud rate generator, its clock source is Osc/2.

The following formula can be used to calculate the baud rate in modes 1 and 3 using the Timer 2:

$$Modes\ 1, 3\ Baud\ Rate = \frac{Oscillator\ Frequency}{32 \times [65536 - (RCAP2H, RCAP2L)]}$$

The formula below is used to define the reload value to put into the RCAP2h, RCAP2L registers to achieve a given baud rate.

$$(RCAP2H, RCAP2L) = 65536 - \frac{Fosc}{32 \times [Baud\ Rate]}$$

In the above formula, RCAP2H and RCAP2L are the content of RCAP2H and RCAP2L taken as a 16-bit unsigned integer.
Note that a rollover in TH2 does not set TF2, and will not generate an interrupt. Because of this, the Timer 2 interrupt does not have to be disabled when Timer 2 is configured in baud rate generator mode.

Furthermore, when Timer 2 is configured as UART baud rate generator and running (TR2 is set to 1), the user should not try to perform read or write operations to the TH2 or TL2 and RCAP2H, RCAP2L registers

Timer 1 Reload Value in Modes 1 & 3 for UART Baud Rate

The following table gives examples of Timer 1, 8-bit reload value when used as a UART Baud Rate generator and the SMOD bit of the PCON register is set to 1.

	22.184MHz	16.000MHz	14.745MHz	12.000MHz	11.059MHz	8.000MHz	3.57MHz
115200bps	FFh	-	-	-	-	-	-
57600bps	Feh	-	-	-	FFh	-	-
38400bps	FDh	-	FEh	-	-	-	-
31250bps	-	-	-	FEh	-	-	-
19200bps	FAh	-	FCh	-	FDh	-	-
9600bps	F4h	-	F8h	-	FAh	-	-
2400bps	D0h	DDh	E0h	E6h	E8h	-	-
1200bps	A0h	BBh	C0h	CCh	D0h	DDh	-
300bps	-	-	00h	30h	40h	75h	C2h

Timer 2 Reload Value in Modes 1 & 3 for UART Baud Rate

Here are examples of [RCAP2H, RCAP2L] reload values for Timer 2 when it is used as baud rate generator for the VRS900 UART

	22.184MHz	16.000MHz	14.745MHz	12.000MHz	11.059MHz	8.000MHz	3.57MHz
230400bps	FFFDh	-	FFFEh	-	-	-	-
115200bps	FFFAh	-	FFFCh	-	FFFDh	-	-
57600bps	FFF4h	-	FFF8h	-	FFFAh	-	-
38400bps	FFEEh	FFF3h	FFF4h	-	FFF7h	-	-
31250bps	FFEAh	FFF0h	FFF1h	FFF4h	FFF5h	FFF8h	-
19200bps	FFDCh	FFE6h	FFE8h	-	FFEEh	FFF3h	
9600bps	FFB8h	FFCCh	FFD0h	FFD9h	FFDCh	FFE6h	-
2400bps	FEE0h	FF30h	FF40h	FF64h	FF70h	FF98h	FFD1h
1200bps	FDC0h	FE5Fh	FE80h	FEC7h	FEE0h	FF30h	FFA3h
300bps	F700h	F97Dh	FA00h	FB1Eh	FB80h	FCBEh	FE8Bh

UART initialization in Mode 3 using Timer 1

```
;*** INTIALIZE THE UART @ 9600BPS, Fosc=11.0592MHz

INISER0T1I: MOV  A,T2CON          ;RETRIEVE CURRENT VALUE OF T2CON
            ANL  A,#11001111B     ;RCLK & TCLK BIT = 0 -> TO USE TIMER1
            MOV  T2CON,A          ;BAUD RATE GENERATOR SOURCE FOR UART
            MOV  PCON,#80H        ;SET THE SMOD BIT TO 1
            MOV  TL1,#0FAH        ;CONFIG TIMER1 AT 8BIT WITH AUTO-RELOAD
            MOV  TH1,#0FAH        ;CALCULATE THE TIMER 1 RELOAD VALUE
                                  ;TH1 = [(2^SMOD) * Fosc] / (32 * 12 * Fcomm)
                                  ;TH1 FOR 9600BPS @ 11.059MHz = FAh
            MOV  SCON,#05Ah       ;CONFIG SCON_0 MODE_1
            MOV TMOD,#00100000B ;CONFIG TIMER 1 IN MODE 2, 8BIT
                                  ; + AUTO RELOAD
            MOV  TCON,#01000000B ;START TIMER1


            CLR  SCON.0           ;CLEAR UART RX, TX FLAGS
            CLR  SCON.1

            MOV  SBUF,#DATA       ;SEND ONE BYTE ON THE SERIAL PORT
```

UART initialization in Mode 3, using Timer 2

```
;*** INTIALIZE THE UART @57600BPS, Fosc=11.0592MHz

INISER0T2I: MOV  SCON,#05Ah       ;CONFIG SCON_0 MODE_1,
                                  ;CALCULATE RELOAD VALUE WITH T2
                                  ;RCAP2H,RCAP2L = 65536 - [ Fosc / (32*Fcomm)]
            MOV  RCAP2H,#0FFh     ;RELOAD VALUE 57600bps, 11.059MHz =FFFAh
            MOV  RCAP2L,#0DCh     ;

            MOV T2CON,#034h       ;SERIAL PORT0, TIMER2 RELOAD START


            CLR  SCON.0           ;CLEAR UART RX, TX FLAGS
            CLR  SCON.1

            MOV  SBUF,#DATA       ;SEND ONE BYTE ON THE SERIAL PORT
```

PWM outputs

The VRS900 includes 2 PWM outputs, PWMA and PWMB.

PWM Registers - Port1 Configuration Register

The PWM outputs of the VRS900 are shared with the Port 1 I/O. In order to activate the PWM output, the corresponding bit of the P1IOCTRL register must be set.

TABLE 23: PORT1 I/O CONTROL REGISTER (P1IOCTRL, 9BH)

7	6	5	4
-	PWMBE	-	-

3	2	1	0
-	PWMAE	-	-

Bit	Mnemonic	Description
7		
6	PWMBE	PWMB output Enabled when set to 1
5:3	-	
2	PWMAE	PWMA output Enabled when set to 1
1:0	-	-

PWMA Function Description:

The PWMA channel is controlled by two SFR registers; one for the PWM Data and another one to control the PWMA input Clock, PWMACK.

PWMA Data Register

The PWMA data register is composed of two parts: The upper 5 bits control the duty cycle of the PWM output and the remaining 3 bits control the Narrow Pulse Generator (NPA) which will insert narrow pulses among the PWMA frame made of 8 cycles. The number of pulses inserted into the frame cycle corresponds to the values defined into the NPA bits. The insertion of narrow pulses in the PWM frame Cycle permits to achieve a PWM resolution equivalent to 8-bit.

The table below shows the PWMA Data Register. The PWMA bit holds the content of the PWMA Data Register and determines the duty cycle of the PWMA output waveform. The NPA[2:0] bits will insert narrow pulses in the 8-PWM-cycle frame.

TABLE 24: PWMA DATA REGISTER (PWMA) – SFR A4H

7	6	5	4
PWMA.4	PWMA.3	PWMA.2	PWMA.1

3	2	1	0
PWMA.0	NPA.2	NPA.1	NPA.0

Bit	Mnemonic	Description
7:4	PWMA[4:0]	Contents of PWM Data
3:0	NPA[3:0]	Inserts Narrow Pulses in a 8-PWM-Cycle Frame

The table below shows the number of narrow pulses inserted in an 8-cycle frame vs. the NPA number.

NPA[2:0]	Number of Narrow Pulses inserted in an 8-cycle frame
000	1
001	2
010	3
011	4
100	5
101	6
110	7
111	8

PWMA Control Register

The table below shows the structure of the PWMA Control Register. This register controls frequency at which the PWMA operates.

TABLE 25: PWM CONTROL REGISTER (PWMACTRL) – SFR A3H

7	6	5	4	3	2	1	0
Unused						PWMACK1	PWMACK0

Bit	Mnemonic	Description
7:2	Unused	-
1	PWMACK1	Input Clock Frequency Divider Bit 1
0	PWMACK0	Input Clock Frequency Divider Bit 0

The following table shows the relationship between the values of PWMACK1/PWMACK0 and the value of the divider. Numerical values of the corresponding frequencies are also provided.

PWMACK1	PWMACK0	Divider	PWM clock, Fosc=20MHz
0	0	2	10MHz
0	1	4	5MHz
1	0	8	2.5MHz
1	1	16	1.25MHz

The following formulas can be used to calculate the PWMA output frequency as well as the PWMA Frame Rate:

$$\text{PWMA Clock} = \frac{F_{osc}}{2^{(\text{PWMACTRL [1:0]} +1)}}$$

$$\text{PWMA Frame} = \frac{F_{osc}}{32 \times 2^{(\text{PWMACTRL [1:0]} +1)}}$$

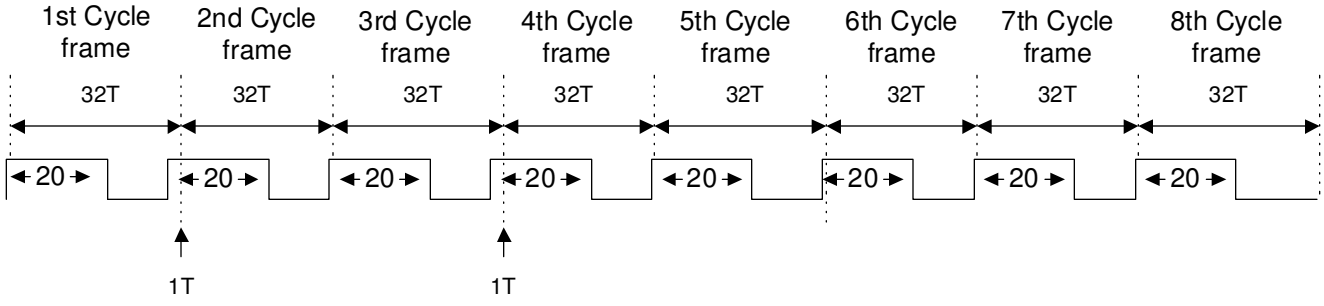
Example of PWM Timing Diagram

The following example shows the configuration of the PWMA.

If Fosc = 20MHz, PWMACTRL = #02H, then PWMA clock = 20MHz/2^(2+1) = 20MHz/8 = 2.5MHz. PWMA output cycle frame frequency = (20MHz/2^(2+1))/32 = 78.1 kHz

```
MOV PWMACTRL,#02H     ; PWMA Clock = Fosc/8
MOV PWMA #82H         ; PWMA[4:0]=10h (=20T high, 12T low), NPA[2:0] = 2
MOV P1IOCTRL, #04H    ; Enable P1.2 as PWMA output pin
```

FIGURE 16: PWMA TIMING DIAGRAM



(Narrow pulse inserted by NPA[2:0]=2)

PWMA clock= 1/T= Fosc / 2^(PWMACK+1)
The PWMA output cycle frame frequency = PWMA clock/32 = [Fosc/2^(PWMACK+1)]/32

.

PWMB Function Description

The VRS900 PWMB has the ability to operate as a true 8-bit PWM or as a 5-Bit PWM. Contrary to the PWMA, when the PWMB is configured to operate in 5-bit resolution, there is no Narrow Pulses inserted into the PWM frame cycle.

The PWMB channel is controlled by two SFR registers; one is for the PWMB Data and one for the PWMBCTRL to select the resolution PWMB and control the input Clock division factor.

PWMB Data Register

The table below shows the PWMB Data Register. The value contained in the PWMB register determines the duty cycle of the PWM output waveform.

When the PWMB is configured to operate in 5-bit resolution (see below) only the 5 LSB of the PWMB register is available.

TABLE 26: PWMB DATA REGISTER (PWMB) – SFR B3HH

7	6	5	4
PWMB.7	PWMB.6	PWMB.5	PWMB.4

3	2	1	0
PWMB.3	PWMB.2	PWMB.1	PWMB.0

Bit	Mnemonic	Description
7:0	PWMB[7:0]	PWM duty cycle control

PWMB Control Register

The table below represents the PWMB Control Register.

TABLE 27: PWMB CONTROL REGISTER (PWMBCTRL) – SFR D3H

7	6	5	4	3	2	1	0
					PWMBRES	PWMBCK1	PWMBCK0

Bit	Mnemonic	Description
[7:3]	Unused	-
2	PWMBRES	0: Set PWMB resolution to 8-bit 1: Set PWMB resolution to 5-bit
1	PWMBCK1	Input Clock Frequency Divider Bit 1
0	PWMBCK0	Input Clock Frequency Divider Bit 0

The following formulas can be used to calculate the PWMB output frequency:

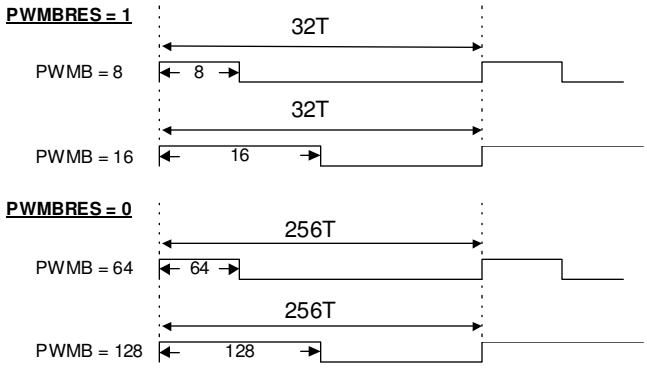
$$\text{PWMB Clock} = \frac{F_{osc}}{2^{(\text{PWMBCK [1:0]} +4)}}$$

The following table gives an example of the PWMB output frequency in relation to the PWMBCK[1:0] bit value when a 20MHz oscillator is used:

PWMBK1	PWMBCK0	Divider	PWMB clock, Fosc=20MHz
0	0	16	1.25MHz
0	1	32	625 KHz
1	0	64	312.5KHz
1	1	128	156.2KHz

The following diagram shows examples of the relationship between the PWMB duty cycle vs. the PWMB data register content and the value of PWMBRES bit value

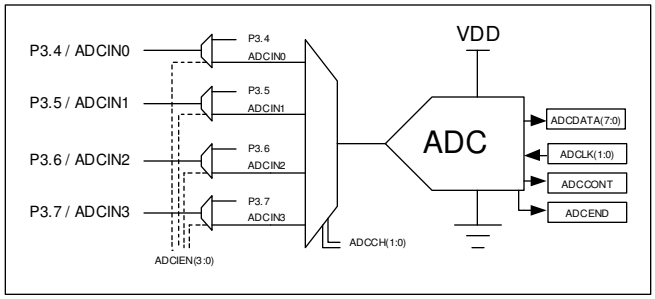
FIGURE 17: PWMB TIMING DIAGRAM EXAMPLES



Analog-to-Digital Converter (ADC)

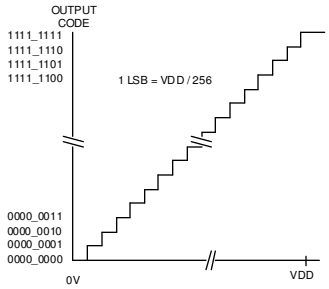
The VRS900 includes 4 Channels, 8-bit A/D converter whose inputs are shared with the I/O Port P3.4 to P3.7. The VRS900's ADC takes its reference from the supply voltage.

FIGURE 18: ADC SRUCTURE



The ADC binary output represents the ratio of the analog voltage present to its input vs. the VRS900 supply as shown in the following figure:

FIGURE 19: ADC OUTPUT VS. ANALOG VOLTAGE PRESENT AT ITS INPUTS



The formula serving to calculate the ADC conversion result is shown below:

$$ADCresult = \frac{Vin}{Vsupply} * 256$$

When the voltage present at the ADC input exceeds the supply voltage, the ADC conversion result will saturate at 0FFh. When the voltage present is lower than the VRS900 ground, the ADC conversion result will remain at 00h

Configuring the VRS900 ADC

The configuration and use of the VRS900 A/D Converter is quite simple and involves the following steps:

- Activate the ADC Input
- Setting the ADC Control Register
- Setting the ADC interrupts (if required)
- Collecting the ADC data

The VRS900 ADC inputs are shared with the upper quartet of Port 3. Writing a 1 into given ADCIENx bit of the P3IOCTRL register assigns the corresponding I/O pin to operate as the ADC inputs. P3 pins for which the ADCIENx bit is kept to 0 can be used as general purpose I/O. When given Port 3 pins are configured as ADC input, writing to the corresponding P3 registers bit assigned to the ADC will have no effect to the device operation and reading these port pins will return the port register value.

TABLE 28: PORT3 CONFIGURATION REGISTER (P3IOCTRL, 9DH)

7	6	5	4
ADCIEN3	ADCIEN2	ADCIEN1	ADCIEN0

3	2	1	0
-	-	-	-

Bit	Mnemonic	Description
7	ADCIEN3	ADC Input 3 Enable 0 = P3.7 I/O 1 = ADC input 3
6	ADCIEN2	ADC Input 2 Enable 0 = P3.6 I/O 1 = ADC input 2
5	ADCIEN1	ADC Input 1 Enable 0 = P3.5 I/O 1 = ADC input 1
4	ADCIEN0	ADC Input 0 Enable 0 = P3.4 I/O 1 = ADC input 0
3:0	-	Unused

The ADCCLTR register allows setting the ADC clock speed value, selecting the channel on which the conversion is to be performed and defines if the ADC will perform a single or continuous conversion of the selected channel.

TABLE 29:ADC CONTROL REGISTER ADCCTRL (8EH)

7	6	5	4
ADCEND	ADCCONT	ADCCLK[1:0]	

3	2	1	0
ADCCH[1:0]		-	-

Bit	Mnemonic	Description
7	ADCEND	ADC End of conversion bit Get set to 1 when the ADC conversion completes. It is cleared when the ADCCTRL is written and when the ADCDATA Register is read.
6	ADCCONT	ADC Continuous conversion Bit 1 = ADC run in continuous and the ADCDATA is refreshed after each conversion is performed on the selected channel. 0 = ADC conversion is performed once
5:4	ADCCLK[1:0]	ADC Clock prescaler (see Table below)
3:2	ADCCH[1:0]	ADC Channel select (See table below)
1:0	-	-

The ADCEND bit is used to monitor the status of the ADC conversion process. At the end of the conversion of the ADCEND Flag is set. Writing to the ADCCTRL register or reading the ADCDATA register automatically clears the ADCEND bit.
When set to 1, the ADCCONT bit of the ADCCTRL register makes the ADC continuously perform conversions on the selected ADC input channel and refreshes the ADCDATA register when the conversion is complete.

Proper operation of the ADC requires feeding a 500KHz to 2.5MHz clock to be feed into the VRS900 ADC. The ADC clock is derived from the VRS900 oscillator and the division factor is controlled by the ADCCLK1 and ADCCLK0 bit of the ADCCTRL register as shown below:

ADCCLK1	ADCCLK0	ADC_CLK
0	0	Fosc / 8*
0	1	Fosc / 16
1	0	Fosc / 32
1	1	Fosc / 64

*Use this Fosc division factor below 20MHz

Operation of the ADC with a clock outside the 500KHz to 2.5MHz frequency range may lead to ADC malfunction.

The bit 3 and 2 of the ADCCTRL register controls the ADC input on which the conversion will be performed.

ADCCH1	ADCCH0	ADC input channel
0	0	ADCIN0
0	1	ADCIN1
1	0	ADCIN2
1	1	ADCIN3

The ADCDATA register is a Read Only register that receives the ADC conversion result.

TABLE 30:ADC DATA REGISTER ADCDATA (8FH)

7	6	5	4	3	2	1	0
ADCDATA[7:0]							

Bit	Mnemonic	Description
7:0	ADCDATA	ADC data register

ADC Conversion Time

The ADC conversion requires 20 ADC clock cycles. Thus the Conversion Rate can be calculated as follows:

$$\textbf{ADC Conv Rate} = \frac{\text{Fadc clock}}{20}$$

$$\textbf{ADC Conv Rate} = \frac{\text{Fosc}}{20 * 2^{(\text{ADCCLK[1:0]} + 3)}}$$

VRS900 ADC Initialization and Use Example

The following example shows the required steps to configure the VRS900 ADC to read channel 0 in continuous mode and use the ADC interrupt to retrieve the conversion result.

```
(...)
        ;*** INITIALIZE THE A/D CONVERTER
        MOV   P3CON,#10000000B          ;CONFIG P3.7 -> ADCIN3

        MOV   ADCCTRL,#0001110B         ;CONFIG ADCCTRL
                                        ;7  ADCEND  = 0
                                        ;6  ADCCONT = SINGLE CONV.
                                        ;5:4 ADCCLK =  Fosc/16
                                        ;3:2 ADCCH =  ADCIN3
                                        ;1:0 UNUSED
                                        ; Fosc = 11.059MHz CONV=34.5KHz
WAITADC: MOV   A,ADCCTRL                ;WAIT FOR ADC CONVERSION TO
        ANL   A,#80H                    ;COMPLETE
        JZ    WAITADC

        MOV   BINL,ADCDATA              ;RETRIEVE ADC DATA
(...)
```

Integrated LCD Driver

The VRS900 includes an on-chip LCD driver which is ideal to drive custom LCD panels often found in consumer, medical and industrial systems.

The LCD driver has provisions to directly drive 14 segments x 4 commons LCD panels, without requiring other external components.

Once configured, the LCD driver peripheral operates independently of the processor and generates the appropriate signals to display the data saved into the LCD buffer registers, accessible from SFR registers.

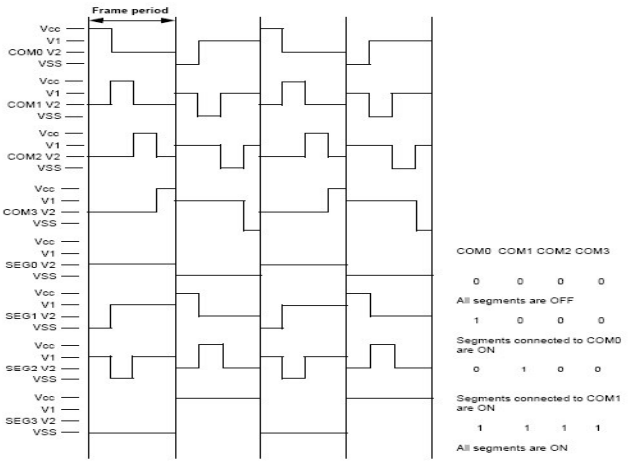
The VRS900 LCD Driver works on 1/4 duty and 1/3 bias. The power consumption of the LCD Driver when activated is about 0.88mA and this figure falls to 1.2uA when deactivated.

Timing Chart of LCD Driver Output

The 14 segment drivers and the 4 common drivers are 4-level outputs that switch between VDD, V1, V2 and VSS LCD driver voltages levels.

The LCD segment/common states are stored into 6 SFR register named LCDBUFx registers.
Each one of the LCDBUFx registers control the state of two LCD segments for each one of the time Slot driven by the activation of the LCDCOMx lines. The following diagram shows a typical LCD driver output timing diagram of the relationship between the segment / common signals.

FIGURE 20: LCD DRIVER OUTPUT TIMING DIAGRAM



Configuring the LCD Driver

The initialization of the LCD driver is performed using the LCDCTRL, P0IOCTRL, P2IOCTRL and the LCDBUF[6:0] registers.

The LCD driver outputs are multiplexed with regular VRS900 I/Os. For this reason, the I/Os required for LCD operation must be assigned to the LCD driver. This is done by setting the corresponding bits of P0IOCTRL and P2IOCTRL registers to 1, and if needed, the LCDPRI bit of the LCDCTRL register.

TABLE 31: PORT 0 I/O CONTROL REGISTER (P0IOCTRL, 9AH)

7	6	5	4
LCDSEG6	LCDSEG7	LCDSEG8	LCDSEG9

3	2	1	0
LCDSEG10	LCDSEG11	LCDSEG12	LCDSEG13

Bit	Mnemonic	Description
7	LCDSEG6	1= Assign P0.7 to LCD Seg. 6 driver
6	LCDSEG7	1= Assign P0.6 to LCD Seg. 7 driver
5	LCDSEG8	1= Assign P0.5 to LCD Seg. 8 driver
4	LCDSEG9	1= Assign P0.4 to LCD Seg. 9 driver
3	LCDSEG10	1= Assign P0.3 to LCD Seg. 10 driver
2	LCDSEG11	1= Assign P0.2 to LCD Seg. 11 driver
1	LCDSEG12	1= Assign P0.1 to LCD Seg. 12 driver
0	LCDSEG13	1= Assign P0.0 to LCD Seg. 13 driver

TABLE 32: PORT 2 I/O CONTROL REGISTER (P2IOCTRL, 9CH)

7	6	5	4
LCDSEG3	LCDSEG2	LCDSEG1	LCDSEG0-

3	2	1	0
LCDCOM3	LCDCOM2	LCDCOM1	LCDCOM0

Bit	Mnemonic	Description
7	LCDSEG3	1= Assign P2.7 to LCD Seg. 3 driver
6	LCDSEG2	1= Assign P2.6 to LCD Seg. 2 driver
5	LCDSEG1	1= Assign P2.6 to LCD Seg. 1 driver
4	LCDSEG0	1= Assign P2.6 to LCD Seg. 0 driver
3	LCDCOM3	1= Assign P2.6 to LCD Com. 3 driver
2	LCDCOM2	1= Assign P2.6 to LCD Com. 2 driver
1	LCDCOM1	1= Assign P2.6 to LCD Com. 1 driver
0	LCDCOM0	1= Assign P2.6 to LCD Com. 0 driver

The activation of the LCD is done by setting the LCDEN bit of the LCDCTRL register.

The LCDON bit of the LCDCTRL serves to turn the display ON so the content of the LCDBUFx registers is sent to the display.

The LCDPRI bit, when set to 1 makes the VRS900 PSEN and ALE pins being assigned as LCDSEG4 and LCDSEG5 lines respectively.

TABLE 33: LCD CONTROL REGISTER (LCDCTRL, DFH)

7	6	5	4
LCDON	LCDEN	LCDPRI	-

3	2	1	0
-	LCDCLK2	LCDCLK1	LCDCLK0

Bit	Mnemonic	Description
7	LCDON	1 = LCD Display is ON 0 = LCD Display is OFF
6	LCDEN	1 = LCD is enabled 0 = LCD is disabled
5	LCDPRI	1 = Give priority of LCD operation on #PSEN/LCDSEG4 and ALE/LCDSEG5 pins
4:3	-	Unused
2:0	LCDCLK[2:0]	LCD prescaler select

The LCD driver's clock can be adjusted to meet specific driving speed requirements of specific LCD displays. The LCDCLK[2:0] bits of the LCDCTRL register are used to define the LCD driver clock prescaler value as shown below:

LCDCLK[2:0]	LCD Clock prescaler value
000	1
001	2
010	4
011	8
100	16
101	32
110	64
111	128

The LCD clock speed can be calculated using the formula below:

$$F_{CLK\_LCD} = \frac{F_{osc}}{2 * 32 * LCDCLK[2:0]}$$

From which the LCD Frame frequency can be defined as follow:

$$F_{LCD\_Frame} = F_{CLKLCD} / 256$$

The typical range of F_{Frame} is:

1026HZ ~ 8HZ at 16MHz (Fosc = 8MHz)

The six LCDBUFx registers contain the mapping of the LCDSEGxx / LCDCOMx segment state. To make a given segment be activated (ON) during one of the 4 time slot of each Frame the bit corresponding to the Segment / Com must be set to 1.

For example, if one wants to make the LCDSEG 0 active during the second time slot (controlled by LCDCOM2) of each Frame, the bit 6 of the LCDBUF0 must be set to 1. If the LCDSEG 0 must also be active during the LCDCOM1 time slot, then both bits 6 and bit 5 of the LCDBUF0 register must be set.

The following tables describe the LCD Segment/ LCD Common combination each one of the LCDBUFx registers controls.

TABLE 34: LCD BUFFER REGISTER 0 (LCDBUF0, E1H)

7	6	5	4
SEG0_COM3	SEG0_COM2	SEG0_COM1	SEG0_COM0

3	2	1	0
SEG1_COM3	SEG1_COM2	SEG1_COM1	SEG1_COM0

Bit	Mnemonic	Description
7	SEG0_COM3	If set to 1, the LCDSEG0 will be ON during LCDCOM3 time slot
6	SEG0_COM2	If set to 1, the LCDSEG0 will be ON during LCDCOM2 time slot
5	SEG0_COM1	If set to 1, the LCDSEG0 will be ON during LCDCOM1 time slot
4	SEG0_COM0	If set to 1, the LCDSEG0 will be ON during LCDCOM0 time slot
3	SEG1_COM3	If set to 1, the LCDSEG1 will be ON during LCDCOM3 time slot
2	SEG1_COM2	If set to 1, the LCDSEG1 will be ON during LCDCOM2 time slot
1	SEG1_COM1	If set to 1, the LCDSEG1 will be ON during LCDCOM1 time slot
0	SEG1_COM0	If set to 1, the LCDSEG1 will be ON during LCDCOM0 time slot

TABLE 35: LCD BUFFER REGISTER 1 (LCDBUF1, E2H)

7	6	5	4
SEG2_COM3	SEG2_COM2	SEG2_COM1	SEG2_COM0

3	2	1	0
SEG3_COM3	SEG3_COM2	SEG3_COM1	SEG3_COM0

Bit	Mnemonic	Description
7	SEG2_COM3	If set to 1, the LCDSEG2 will be ON during LCDCOM3 time slot
6	SEG2_COM2	If set to 1, the LCDSEG2 will be ON during LCDCOM2 time slot
5	SEG2_COM1	If set to 1, the LCDSEG2 will be ON during LCDCOM1 time slot
4	SEG2_COM0	If set to 1, the LCDSEG2 will be ON during LCDCOM0 time slot
3	SEG3_COM3	If set to 1, the LCDSEG3 will be ON during LCDCOM3 time slot
2	SEG3_COM2	If set to 1, the LCDSEG3 will be ON during LCDCOM2 time slot
1	SEG3_COM1	If set to 1, the LCDSEG3 will be ON during LCDCOM1 time slot
0	SEG3_COM0	If set to 1, the LCDSEG3 will be ON during LCDCOM0 time slot

TABLE 36: LCD BUFFER REGISTER 2 (LCDBUF2, E3H)

7	6	5	4
SEG4_COM3	SEG4_COM2	SEG4_COM1	SEG4_COM0

3	2	1	0
SEG5_COM3	SEG5_COM2	SEG5_COM1	SEG5_COM0

Bit	Mnemonic	Description
7	SEG4_COM3	If set to 1, the LCDSEG4 will be ON during LCDCOM3 time slot
6	SEG4_COM2	If set to 1, the LCDSEG4 will be ON during LCDCOM2 time slot
5	SEG4_COM1	If set to 1, the LCDSEG4 will be ON during LCDCOM1 time slot
4	SEG4_COM0	If set to 1, the LCDSEG4 will be ON during LCDCOM0 time slot
3	SEG5_COM3	If set to 1, the LCDSEG5 will be ON during LCDCOM3 time slot
2	SEG5_COM2	If set to 1, the LCDSEG5 will be ON during LCDCOM2 time slot
1	SEG5_COM1	If set to 1, the LCDSEG5 will be ON during LCDCOM1 time slot
0	SEG5_COM0	If set to 1, the LCDSEG5 will be ON during LCDCOM0 time slot

TABLE 37: LCD BUFFER REGISTER 3 (LCDBUF3, E4H)

7	6	5	4
SEG6_COM3	SEG6_COM2	SEG6_COM1	SEG6_COM0

3	2	1	0
SEG7_COM3	SEG7_COM2	SEG7_COM1	SEG7_COM0

Bit	Mnemonic	Description
7	SEG6_COM3	If set to 1, the LCDSEG6 will be ON during LCDCOM3 time slot
6	SEG6_COM2	If set to 1, the LCDSEG6 will be ON during LCDCOM2 time slot
5	SEG6_COM1	If set to 1, the LCDSEG6 will be ON during LCDCOM1 time slot
4	SEG6_COM0	If set to 1, the LCDSEG6 will be ON during LCDCOM0 time slot
3	SEG7_COM3	If set to 1, the LCDSEG7 will be ON during LCDCOM3 time slot
2	SEG7_COM2	If set to 1, the LCDSEG7 will be ON during LCDCOM2 time slot
1	SEG7_COM1	If set to 1, the LCDSEG7 will be ON during LCDCOM1 time slot
0	SEG7_COM0	If set to 1, the LCDSEG7 will be ON during LCDCOM0 time slot

TABLE 38: LCD BUFFER REGISTER 4 (LCDBUF4, E5H)

7	6	5	4
SEG8_COM3	SEG8_COM2	SEG8_COM1	SEG8_COM0

3	2	1	0
SEG9_COM3	SEG9_COM2	SEG9_COM1	SEG9_COM0

Bit	Mnemonic	Description
7	SEG8_COM3	If set to 1, the LCDSEG8 will be ON during LCDCOM3 time slot
6	SEG8_COM2	If set to 1, the LCDSEG8 will be ON during LCDCOM2 time slot
5	SEG8_COM1	If set to 1, the LCDSEG8 will be ON during LCDCOM1 time slot
4	SEG8_COM0	If set to 1, the LCDSEG8 will be ON during LCDCOM0 time slot
3	SEG9_COM3	If set to 1, the LCDSEG9 will be ON during LCDCOM3 time slot
2	SEG9_COM2	If set to 1, the LCDSEG9 will be ON during LCDCOM2 time slot
1	SEG9_COM1	If set to 1, the LCDSEG9 will be ON during LCDCOM1 time slot
0	SEG9_COM0	If set to 1, the LCDSEG9 will be ON during LCDCOM0 time slot

TABLE 39: LCD BUFFER REGISTER 5 (LCDBUF5, E6H)

7	6	5	4
SEG10_COM3	SEG10_COM2	SEG10_COM1	SEG10_COM0

3	2	1	0
SEG11_COM3	SEG11_COM2	SEG11_COM1	SEG11_COM0

Bit	Mnemonic	Description
7	SEG10_COM3	If set to 1, the LCDSEG10 will be ON during LCDCOM3 time slot
6	SEG10_COM2	If set to 1, the LCDSEG10 will be ON during LCDCOM2 time slot
5	SEG10_COM1	If set to 1, the LCDSEG10 will be ON during LCDCOM1 time slot
4	SEG10_COM0	If set to 1, the LCDSEG10 will be ON during LCDCOM0 time slot
3	SEG11_COM3	If set to 1, the LCDSEG11 will be ON during LCDCOM3 time slot
2	SEG11_COM2	If set to 1, the LCDSEG11 will be ON during LCDCOM2 time slot
1	SEG11_COM1	If set to 1, the LCDSEG11 will be ON during LCDCOM1 time slot
0	SEG11_COM0	If set to 1, the LCDSEG11 will be ON during LCDCOM0 time slot

Table 40: LCD Buffer Register 6 **(LCDBUF6, E7h)**

7	6	5	4
SEG12_COM3	SEG12_COM2	SEG12_COM1	SEG12_COM0

3	2	1	0
SEG13_COM3	SEG13_COM2	SEG13_COM1	SEG13_COM0

Bit	Mnemonic	Description
7	SEG12_COM3	If set to 1, the LCDSEG12 will be ON during LCDCOM3 time slot
6	SEG12_COM2	If set to 1, the LCDSEG12 will be ON during LCDCOM2 time slot
5	SEG12_COM1	If set to 1, the LCDSEG12 will be ON during LCDCOM1 time slot
4	SEG12_COM0	If set to 1, the LCDSEG12 will be ON during LCDCOM0 time slot
3	SEG13_COM3	If set to 1, the LCDSEG13 will be ON during LCDCOM3 time slot
2	SEG13_COM2	If set to 1, the LCDSEG13 will be ON during LCDCOM2 time slot
1	SEG13_COM1	If set to 1, the LCDSEG13 will be ON during LCDCOM1 time slot
0	SEG13_COM0	If set to 1, the LCDSEG13ill be ON during LCDCOM0 time slot

VRS900 LCD Driver Example Program

The program example shows the basic steps required to perform the initialization and use of the VRS900 LCD Driver

```
;** LCD DRIVER INITIALISATION

        MOV  P0IOCTRL,#0FFH          ;Assign I/O pin to LCD driver
        MOV  P2IOCTRL,#0FFH

        MOV LCDCON,#11100110B        ;LCD_ON=1
                                     ;LCD_EN =1 -> ENABLE
                                     ;SEG = 1
                                     ;BIT3, BIT4 = UNUSED
                                     ;LS[2:0] = 110 -> PRESCALER = 64

;**LCD SEGMENTS CONFIGURATION
        MOV LCDBUF0,#00010010B       ;LCDSEG0 is ON during COM0
                                     ;& LCDSEG1 is ON during
                                     ;LCDCOM1 period
        MOV LCDBUF1,#01000000B       ;LCDSEG2 is ON during LCDCOM2
                                     ;period
        MOV LCDBUF2,#11111111B       ;LCDSEG4 & LCDSEG5 are always
                                     ;ON (...)
        MOV LCDBUF6,#00000010B       ;LCDSEG13 is ON during LCDCOM1
```

The Following Table gives a condensed view of the LCD Segment / LCD Common control vs. LCDBUFx registers.

		LCDCOM3	**LCDCOM2**	**LCDCOM1**	**LCDCOM0**	**LCDCOM3**	**LCDCOM2**	**LCDCOM1**	**LCDCOM0**
Mnemonic	Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LCDBUF0	E1H	LCDSEG0	LCDSEG0	LCDSEG0	LCDSEG0	LCDSEG1	LCDSEG1	LCDSEG1	LCDSEG1
LCDBUF1	E2H	LCDSEG2	LCDSEG2	LCDSEG2	LCDSEG2	LCDSEG3	LCDSEG3	LCDSEG3	LCDSEG3
LCDBUF2	E3H	LCDSEG4	LCDSEG4	LCDSEG4	LCDSEG4	LCDSEG5	LCDSEG5	LCDSEG5	LCDSEG5
LCDBUF3	E4H	LCDSEG6	LCDSEG6	LCDSEG6	LCDSEG6	LCDSEG7	LCDSEG7	LCDSEG7	LCDSEG7
LCDBUF4	E5H	LCDSEG8	LCDSEG8	LCDSEG8	LCDSEG8	LCDSEG9	LCDSEG9	LCDSEG9	LCDSEG9
LCDBUF5	E6H	LCDSEG10	LCDSEG10	LCDSEG10	LCDSEG10	LCDSEG11	LCDSEG11	LCDSEG11	LCDSEG11
LCDBUF6	E7H	LCDSEG12	LCDSEG12	LCDSEG12	LCDSEG12	LCDSEG13	LCDSEG13	LCDSEG13	LCDSEG13

Interrupts

The VRS900 has 9 interrupt sources (10 if we include the WDT) and 8 interrupt vectors (including reset) to handle them. The interrupt can be enabled via the IE register shown below:

TABLE 41: IEN0 INTERRUPT ENABLE REGISTER –SFR A8H

7	6	5	4	3	2	1	0
EA	-	ET2	ES	ET1	EX1	ET0	EX0

Bit	Mnemonic	Description
7	EA	Disables All Interrupts 0: no interrupt acknowledgment 1: Each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
6	-	Reserved
5	ET2	Timer 2 Interrupt Enable Bit
4	ES	Serial Port Interrupt Enable Bit
3	ET1	Timer 1 Interrupt Enable Bit
2	EX1	External Interrupt 1 Enable Bit
1	ET0	Timer 0 Interrupt Enable Bit
0	EX0	External Interrupt 0 Enable Bit

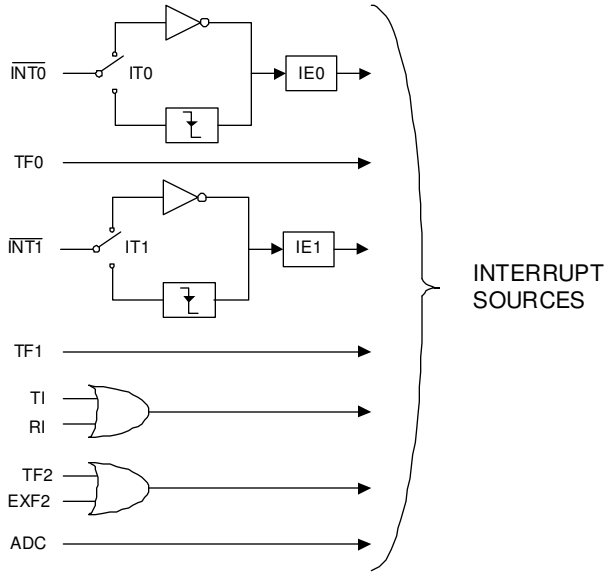
TABLE 42: IEN1 INTERRUPT ENABLE REGISTER 1–SFR A9H

7	6	5	4	3	2	1	0
-	-	-	-	ADCIE	-	-	-

Bit	Mnemonic	Description
7:4	-	-
3	ADCIE	ADC Interru[pt Enable
2:0	-	-

The following figure illustrates the various interrupt sources on the VRS900.

FIGURE 21: INTERRUPT SOURCES



Interrupt Vectors

The table shown below specifies each interrupt source, its flag and its vector address.

TABLE 43: INTERRUPT VECTOR ADDRESS

Interrupt Source	Flag	Vector Address
RESET (+ WDT)	WDR	0000h*
INT0	IE0	0003h
Timer 0	TF0	000Bh
INT1	IE1	0013h
Timer 1	TF1	001Bh
Serial Port	RI+TI	0023h
Timer 2	TF2+EXF2	002Bh
ADC Interrupt	ADCIF	004Bh

Execution of an Interrupt

When the processor receives an interrupt request, an automatic jump to the desired subroutine occurs. This jump is similar to executing a branch to a subroutine instruction: the processor automatically saves the address of the next instruction on the stack. An internal flag is set to indicate that an interrupt is taking place, and then the jump instruction is executed. An interrupt

subroutine must always end with the RETI instruction. This instruction allows the processor to retrieve the return address placed on the stack and update the internal flags of the interrupt controller.

Interrupt Enable and Interrupt Priority

When the VRS900 is reset, the IEN0 and IEN1 registers are cleared, disabling all the interrupts. In order to enable the interrupt the corresponding bits of the IEN0 and IEN1 registers must be set.

The IEN0 register is part of the bit addressable internal RAM. For this reason, it is possible to modify each bit individually in one instruction without having to modify the other bits of the register. The IEN1 register that control the ADC interrupt is not bit addressable. In order to enable the ADC interrupt, a direct write must be performed IEN1 register to set the ADCIE bit to 1.

All interrupts can be inhibited by clearing the EA bit of the IEN0 register.

The order in which interrupts are serviced is shown in the following table:

TABLE 44: INTERRUPT PRIORITY

Interrupt Source
RESET + WDT (Highest Priority)
IE0
TF0
IE1
TF1
RI+TI
TF2+EXF2
ADCIP (Lowest Priority)

Modifying the Order of Priority

The VRS900 allows the user to modify the natural priority of the interrupts. One may modify the order by programming the bits in the IP (Interrupt Priority) register. When any bit in this register is set to 1, it gives the corresponding source a greater priority than interrupts coming from sources that don't have their corresponding IP or IP1 bit set to 1.

The IP and IP1 registers structures are represented in the tables below.

TABLE 45: IP INTERRUPT PRIORITY REGISTER –SFR B8H

7	6	5	4	3	2	1	0
-	-	PT2	PS	PT1	PX1	PT0	PX0

Bit	Mnemonic	Description
7	-	
6	-	
5	PT2	Gives Timer 2 Interrupt Higher Priority
4	PS	Gives Serial Port Interrupt Higher Priority
3	PT1	Gives Timer 1 Interrupt Higher Priority
2	PX1	Gives INT1 Interrupt Higher Priority
1	PT0	Gives Timer 0 Interrupt Higher Priority
0	PX0	Gives INT0 Interrupt Higher Priority

TABLE 46: IP1 INTERRUPT PRIORITY REGISTER 1 –SFR B9H

7	6	5	4	3	2	1	0
-	-	-	-	ADCIP	-	-	-

Bit	Mnemonic	Description
7:4	-	
3	ADCIP	Gives ADC Interrupt Higher Priority
2:0	-	

External Interrupts

The VRS900 has two external interrupt inputs named INT0 and INT1. These interrupt lines are shared with P3.2 and P3.3.

The IE0 and IE1 bit of the TCON register are External flags that indicate a low level or high-to-low transition has been detected on the INT0, INT1 interrupt pins respectively.
These flags are automatically cleared when the corresponding interrupt is serviced.

The bits IT0 and IT1 of the TCON register determine whether the external interrupts are level or edge sensitive.

IT0 = 0: The INT0, if enabled, occurs if a Low Level is present on P3.2

IT0 = 1: The INT0, if enabled, occurs if a High to Low transition is detected on P3.2

IT1 = 0: The INT1, if enabled, occurs if a Low Level is present on P3.3

IT1 = 1: The INT1, if enabled, occurs if a High to Low transition is detected on P3.3

The state of the external interrupt, when enabled, can be monitored using the flags, IE0 and IE1 of the TCON register that are set when the interrupt condition occurs.

In the case where the interrupt is configured as edge sensitive, the associated flag is automatically cleared when the interrupt is serviced.

If the interrupt is configured as level sensitive, then the interrupt flag must be cleared by the software.

Timer 0 and Timer 1 Interrupt

Both Timer 0 and Timer 1 can be configured to generate an interrupt when a rollover of the timer/counter occurs (except Timer 0 in Mode 3).

The TF0 and TF1 flags serve to monitor timer overflow occurring from Timer 0 and Timer 1. These interrupt flags are automatically cleared when the interrupt is serviced.

Timer 2 interrupt

Timer 2 interrupt can occur if TF2 and/or EXF2 flags are set to 1 and if the Timer 2 interrupt is enabled.

The TF2 flag is set when a rollover of Timer 2 Counter/Timer occurs. The EXF2 flag can be set by a 1->0 transition on the T2EX pin by the software.

Note that neither flag is cleared by the hardware upon execution of the interrupt service routine. The service routine may have to determine whether it was TF2 or EXF2 that generated the interrupt. These flag bits will have to be cleared by the software.

Every bit that generates interrupts can either be cleared or set by the software, yielding the same result as when the operation is done by the hardware. In other words, pending interrupts can be cancelled and interrupts can be generated by the software.

Serial Port Interrupt

The serial port can generate an interrupt upon byte reception or when the byte transmission is completed.

Those two conditions share the same interrupt vector and it is up to the interrupt service routine to find out what caused the interrupt by looking at the serial interrupt flags RI and TI.

Note that neither of these flags is cleared by the hardware upon execution of the interrupt service routine. The software must clear these flags.

ADC Interrupt

Like other peripherals of the VRS900, the A/D converter can generate interrupt to the processor once the conversion is completed. The interrupt vector associated to the A/D converter is 04Bh

The IP1, IEN1 and IFI special function register controls the ADC interrupt.

To activate the ADC interrupt the ADCIE bit of the IEN1 register must be set as well as the general interrupt bit, EA bit 7 of IEN0 register.

TABLE 47: INTERRUPT ENABLE REGISTER (IEN1, A9H)

7	6	5	4
-	-	-	-

3	2	1	0
ADCIE	-	-	-

Bit	Mnemonic	Description
7:4	-	Unused
3	ADCIE	ADC Interrupt Enable 0 = ADC interrupt Disabled 1 = ADC interrupt Enabled
2:0	-	Unused

By default, the ADC interrupt is set to low priority. However setting the ADCIP bit of the IP1 register will give the ADC a high priority level.

TABLE 48: INTERRUPT PRIORITY REGISTER (IP1, B9H)

7	6	5	4
-	-	-	-

3	2	1	0
ADCIP	-	-	-

Bit	Mnemonic	Description
7:4	-	Unused
3	ADCIP	ADC Interrupt Priority 0 = ADC interrupt is Low Priority 1 = ADC interrupt is High Priority
2:0	-	Unused

When ADC interrupt is authorized and a conversion is completed, the ACDIF flag of the IF1 register will be set to 1. Once the ADC interrupt routine is executed the ADCIF will be automatically cleared.

TABLE 49: INTERRUPT FLAG REGISTER (IF1, AAH)

7	6	5	4
-	-	-	-

3	2	1	0
ADCIF	-	-	-

Bit	Mnemonic	Description
7:4	-	Unused
3	ADCIF	ADC Interrupt Flag Will be set to 1 if ADC interrupt occurred. Cleared automatically when the interrupt is serviced.
2:0	-	Unused

ADC Initialization & Use (by Interrupt)

The following code program shows the basic steps to configure and start the VRS900 A/D converter and then use the ADC interrupt to retrieve conversion results. It is also possible to monitor the ADCEND bit of the ADCCTRL register to know when the A/D conversion process terminates.

```
;*** RESET VECTOR
ORG   0000H
            LJMP    START

;*** ADC INTERRUPT JUMP VECTOR ***
ORG   04BH
            LJMP   IRQADC           ;JUMP TO ADC INTERRUPT ROUTINE

;*** MAIN PROGRAM START ***
ORG    0100H

START:      MOV    SP,#0C0H         ;INITIALISE STACK POINTER

;*** INITIALIZE THE A/D CONVERTER

            MOV    P3IOCTRL,#01000000B      ;CONFIG P3.6 -> ADCIN2
ADCGO:      MOV    ADCCTRL,#01001000B       ;CONFIG ADCCTRL
                                            ;7   ADCEND  = 0
                                            ;6   ADCCONT = CONT CONV.
                                            ;5:4 ADCCLK =  Fosc/8
                                            ;3:2 ADCCH =   ADCI2
                                            ;1:0 UNUSED
                                            ;WITH Fosc = 11.059MHz
                                            ;CONV RATE 69.1KHz
            MOV    ADCVALUE,#00H
            MOV    IEN1,#00001000B          ;ENABLE ADC INTERRUPT
            SETB   EA                       ;ENABLE  GENERAL  INTERRUPTS

            (...)

;**************************
;* ADC INTERRUPT
;**************************
IRQADC:     MOV    ADCVALUE,ADCDATA         ;RETRIEVE ADCDATA
            RETI
```

The Watch Dog Timer

The VRS900 Watch Dog Timer (WDT) is a 16-bit free-running counter operating from an independent 250KHz internal RC oscillator. The overflow of the Watch Dog Timer counter will reset the processor.

The WDT is a useful safety measure for systems that could be affected by noise, power glitches and other conditions that can cause the software to go into infinite dead loops or runaways by giving a recovery mechanism from abnormal software conditions. The WDT is different from Timer 0, Timer 1 and Timer 2.

Watch Dog Timer Registers

The configuration and use of the VRS900 Watch Dog Timer is handled by three registers: WDTKEY, WDTCTRL and SYSCON.

The WDTKEY register provides protection level to ensure that the Watch Dog Timer doesn't get inadvertently reset in case of program malfunction.

The WDTCTRL register is by default configured as a Read-Only register. To modify its contents, two consecutive write operations to the WDTKEY register must be performed first:

```
MOV   WDTKEY,#01Eh
MOV   WDTKEY,#0E1h
```

Once the configuration or WDT reset operation is completed, the WDTCTRL register can be put back in Read-Only by writing the following sequence into the WDTKEY register:

```
MOV   WDTKEY,#0E1h
MOV   WDTKEY,#01Eh
```

TABLE 50: WATCH DOG TIMER KEY REGISTER: WDTKEY – SFR 97H

7	6	5	4	3	2	1	0
WDTKEY7:0							

Bit	Mnemonic	Description
7:0	WDTKEY	Watch Dog Key

Once the WDT is enabled, the user software must clear it periodically. In the case where the WDT is not cleared, its overflow will trigger a reset of the VRS900.

TABLE 51: WATCH DOG TIMER REGISTERS: WDTCTRL – SFR 9FH

7	6	5	4	3	2	1	0
WDTE	Unused	WDT CLR	Unused		WDT PS2	WDT PS1	WDT PS0

Bit	Mnemonic	Description
7	WDTE	Watch Dog Timer Enable Bit
6	Unused	-
5	WDTCLR	Watch Dog Timer Counter Clear Bit
[4:3]	Unused	-
2	WDTPS2	Clock Source Divider Bit 2
1	WDTPS1	Clock Source Divider Bit 1
0	WDTPS0	Clock Source Divider Bit 0

The WDT timeout delay can be adjusted by configuring the clock divider input for the time base source clock of the WDT. To select the divider value, the [WDTPS2~WDTPS0] bits of the Watch Dog Timer Control Register should be set accordingly.

The next table gives the approximate timeout period the user will obtain for different values of the WDTPSx bits of the Watch Dog Timer Register.

TABLE 52: TIMEOUT PERIOD AT

WDTPS [2:0]	WDT Period
000	2.048ms
001	4.096ms
010	8.192ms
011	16.384ms
100	32.768ms
101	65.536ms
110	131.072ms
111	262.144ms

To enable the WDT, the user must set bit 7 (WDTE) of the WDTCTRL register to 1. Once WDTE has been set to 1, the 16-bit counter will start to count from the 250KHz oscillator divided according to the value of the WDTPS2~WDTPS0 bits.

Clearing the WDT is accomplished by setting the WDTCLR bit of the WDTCTRL to 1. This action will clear the contents of the 16-bit counter and force it to restart.

In the case where the Watch Dog Timer overflows, the Watch Dog Timer will reset the processor, the WDR bit (7) of SYSCON register will be set to 1 and the WDTE bit will be cleared to 0. The user should check the WDR bit if an unpredicted reset has taken place.

The user should check the WDR bit of the SYSCON register whenever an unpredicted reset has taken place.

TABLE 53: WATCH DOG TIMER REGISTER-SYSTEM CONTROL REGISTER (SYSCON)–SFR BFH

7	6	5	4	3	2	1	0
WDR	Unused						ALEI

Bit	Mnemonic	Description
7	WDR	Watch Dog Timer Reset Bit
[6:1]	Unused	-
0	ALEI	1: Enable Electromagnetic Interference Reducer 0: Disable Electromagnetic Interference Reducer

WDT initialization Example

The following program example shows the initialization sequence of the Watch Dog Timer and the routine to periodically clear it.

```
;*** VARIABLE DEFINITION ***

CPTR        EQU         020H
PORTVAL     EQU         00H

;*** PROGRAM START HERE ****
ORG   0000h
            LJMP    START

;*** MAIN PROGRAM START ***
ORG   0100h

;*** CHECK IF RESET WAS CAUSED BY THE WATCHDOG TIMER
START:      MOV     A,SYSCON
            ANL     A,#80H
            JNZ     WDTRESET        ;WDT BIT SET -> WE GOT A WDT RESET

INITWDT:    MOV     WDTKEY,#01EH    ;UNLOCK THE WDTCTRL REG ACCESS IN
            MOV     WDTKEY,#0E1H    ;WRITING MODE

            MOV   WDTCTRL,#10000010B  ;CONFIG THE WATCHDOG TIMER
                                      ;BIT 7  - WDTEN=1 WATCHDOG TIMER ENABLE
                                      ;BIT 6   - UNUSED
                                      ;BIT 5   - WDTCLR=1 WATCHDOG CLEAR
                                      ;BIT 4:3 - UNUSED
                                      ;BIT 2:0 - WDTCLK=010 - WDT TIMEOUT = 8mS

            MOV   WDTKEY,#0E1H    ;LOCK THE WDTCTRL ACCESS IN WRITING
            MOV   WDTKEY,#01EH
            MOV   PORTVAL,#00H    ;INIT PORT VALUE TO 00H

WDTRESET: NOP                     ;IF THE WDT CAUSE THE RESET INIT PORTVAL
            MOV   A,PORTVAL       ;TOGGLE P1 VALUE
            CPL   A
            MOV   PORTVAL,A
            MOV   P1,A

;*** SEQUENCE TO CLEAR THE WATCHDOG TIMER (SAME AS CONFIG)
LOOP:       ;MOV   WDTKEY,#01EH  ;UNLOCK THE WDTCTRL REG ACCESS IN
                                 ;WRITING MODE
            ;MOV   WDTKEY,#0E1H
            ;MOV   WDTCTRL,#10100010B      ;CONFIG THE WDT TIMER
                                 ;BIT 7 - WDTEN=1 WDT ENABLE
                                 ;BIT 6   - UNUSED
                                 ;BIT 5   - WDTCLR=1 WDT CLEAR
                                 ;BIT 4:3 - UNUSED
                                 ;BIT 2:0 - WDTCLK=010 - WDT TIMEOUT = 8mS

            ;MOV   WDTKEY,#0E1H  ;LOCK THE WDTCTRL ACCESS IN WRITING
            ;MOV   WDTKEY,#01EH

            (...)

            LJMP   LOOP
```

Crystal Configuration

The crystal connected to the VRS900 oscillator input should be of a parallel type, operating in fundamental mode.

The following table shows the recommended value of capacitors and feedback resistor that must be used at different operating frequencies.

VRS900 Crystal configuration			
XTAL	3MHz	6MHz	12MHz
C1	30 pF	30 pF	30 pF
C2	30 pF	30 pF	30 pF
R	open	open	open

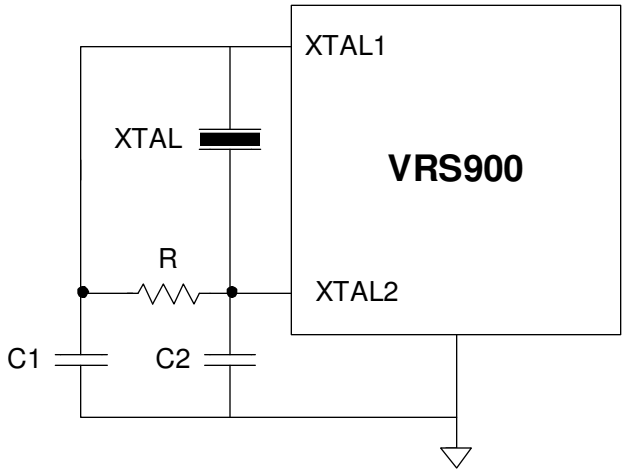
XTAL	16MHz	20MHz	25MHz
C1	30 pF	22 pF	15 pF
C2	30 pF	22 pF	15 pF
R	open	open	62K

Note: Oscillator circuits may differ with different crystals or ceramic resonators in higher oscillation frequency.

Crystals or ceramic resonator characteristics vary from one manufacturer to the other.

The user should check the specific crystal or ceramic resonator technical literature available or contact the manufacturer to select the appropriate values for the external components.

FIGURE 22: CRYSTAL CONFIGURATION



Operating Conditions

TABLE 54: OPERATING CONDITIONS

Symbol	Description	Min.	Typ.	Max.	Unit	Remarks
TA	Operating temperature	-40	25	+85	ºC	Ambient temperature under bias
TS	Storage temperature	-55	25	155	ºC	
VCC5	Supply voltage	4.5	5.0	5.5	V	
Fosc	Oscillator Frequency	3.0	-	25	MHz	

DC Characteristics

TABLE 55: DC CHARACTERISTICS

Symbol	Parameter	Valid	Min.	Max.	Unit	Test Conditions
VIL1	Input Low Voltage	Port 0,1,2,3,4,#EA	-0.5	1.0	V	VCC=5V
VIL2	Input Low Voltage	RES, XTAL1	0	0.8	V	VCC=5V
VIH1	Input High Voltage	Port 0,1,2,3,4,#EA	2.0	VCC+0.5	V	VCC=5V
VI H2	Input High Voltage	RES, XTAL1	70% VCC	VCC+0.5	V	VCC=5V
VOL1	Output Low Voltage	Port 0, ALE, #PSEN		0.4	V	IOL=3.2mA
VOL2	Output Low Voltage	Port 1,2,3,4		0.4	V	IOL=1.6mA
VOH1	Output High Voltage	Port 0	2.4		V	IOH=-800uA
			90%VCC		V	IOH=-80uA
VOH2	Output High Voltage	Port 1,2,3,4,ALE,#PSEN	2.4		V	IOH=-60uA
			90% VCC		V	IOH=-10uA
IIL	Logical 0 Input Current	Port 1,2,4, P3.0-P3.3		-50	uA	Vin=0.45V
ITL	Logical Transition Current	Port 1,2,3,4,P3.0-P3.3		-650	uA	Vin=2.0V
ILI	Input Leakage Current	Port 0, #EA		10	uA	0.45V<Vin< 5V
R RES	Reset Pull-down Resistance	RES	18	90	Kohm	
C 10	Pin Capacitance			10	pF	Fre=1 MHz, Ta=25°C
ICC	Power Supply Current	VDD		20	mA	Active mode, 16MHz
				10	mA	Idle mode, 16MHz
				100	uA	Power down mode

FIGURE 23: ICC ACTIVE MODE TEST CIRCUIT

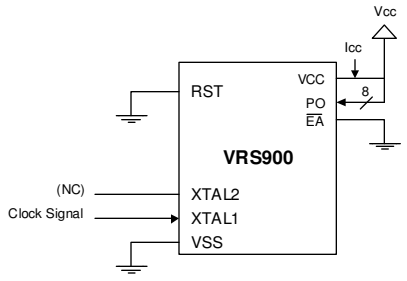
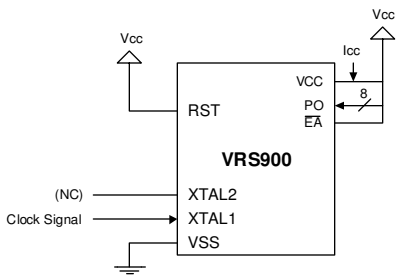


FIGURE 24: ICC IDLE MODE TEST CIRCUIT



AC Characteristics

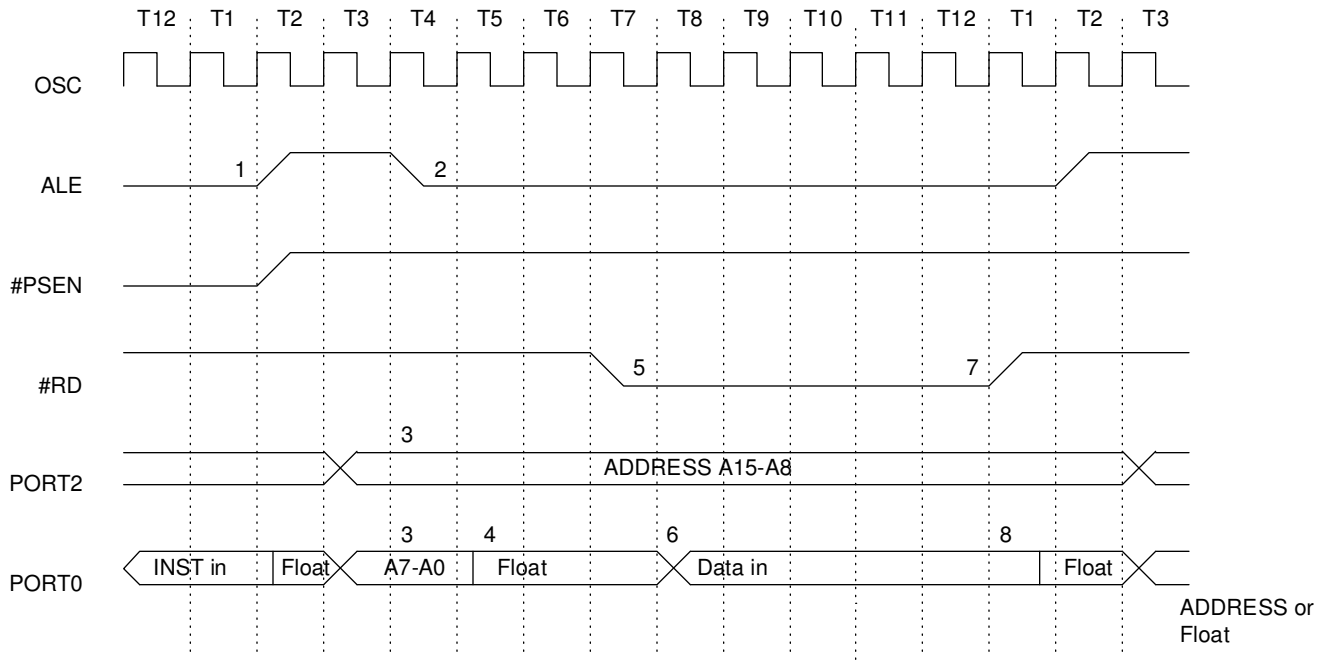
TABLE 56: AC CHARACTERISTICS

Symbol	Parameter	Valid Cycle	Fosc 16			Variable Fosc			Unit
			Min.	Type	Max.	Min.	Type	Max.	
T LHLL	ALE Pulse Width	RD/WRT	115			2xT - 10			nS
T AVLL	Address Valid to ALE Low	RD/WRT	43			T - 20			nS
T LLAX	Address Hold after ALE Low	RD/WRT	53			T - 10			nS
T LLIV	ALE Low to Valid Instruction In	RD			240			4xT - 10	nS
T LLPL	ALE Low to #PSEN low	RD	53			T - 10			nS
T PLPH	#PSEN Pulse Width	RD	173			3xT - 15			nS
T PLIV	#PSEN Low to Valid Instruction In	RD			177			3xT -10	nS
T PXIX	Instruction Hold after #PSEN	RD	0			0			nS
T PXIZ	Instruction Float after #PSEN	RD			87			T + 25	nS
T AVIV	Address to Valid Instruction In	RD			292			5xT - 20	nS
T PLAZ	#PSEN Low to Address Float	RD			10			10	nS
T RLRH	#RD Pulse Width	RD	365			6xT - 10			nS
T WLWH	#WR Pulse Width	WRT	365			6xT - 10			nS
T RLDV	#RD Low to Valid Data In	RD			302			5xT - 10	nS
T RHDX	Data Hold after #RD	RD	0			0			nS
T RHDZ	Data Float after #RD	RD			145			2xT + 20	nS
T LLDV	ALE Low to Valid Data In	RD			590			8xT - 10	nS
T AVDV	Address to Valid Data In	RD			542			9xT - 20	nS
T LLYL	ALE low to #WR High or #RD Low	RD/WRT	178		197	3xT - 10		3xT + 10	nS
T AVYL	Address Valid to #WR or #RD Low	RD/WRT	230			4xT - 20			nS
T QVWH	Data Valid to #WR High	WRT	403			7xT - 35			nS
T QVWX	Data Valid to #WR Transition	WRT	38			T - 25			nS
T WHQX	Data Hold after #WR	WRT	73			T + 10			nS
T RLAZ	#RD Low to Address Float	RD						5	nS
T YALH	#W R or #RD High to ALE High	RD/WRT	53		72	T -10		T+10	nS
T CHCL	Clock Fall Time								nS
T CLCX	Clock Low Time								nS
T CLCH	Clock Rise Time								nS
T CHCX	Clock High Time								nS
T,TCLCL	Clock Period		63				1/fosc		nS

Data Memory Read Cycle Timing

The following timing diagram shows the signal timing of Data Memory Read Cycle.

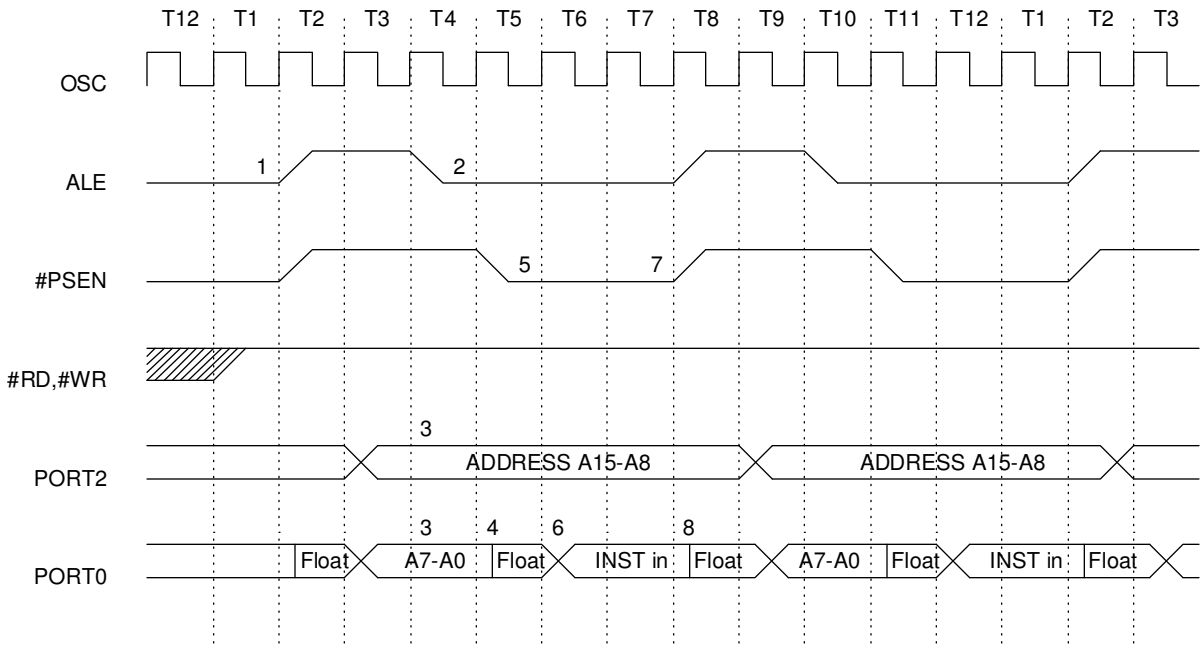
FIGURE 25: DATA MEMORY READ CYCLE TIMING



Program Memory Read Cycle Timing

The following timing diagram shows the signal timing during Program Memory Read Cycle.

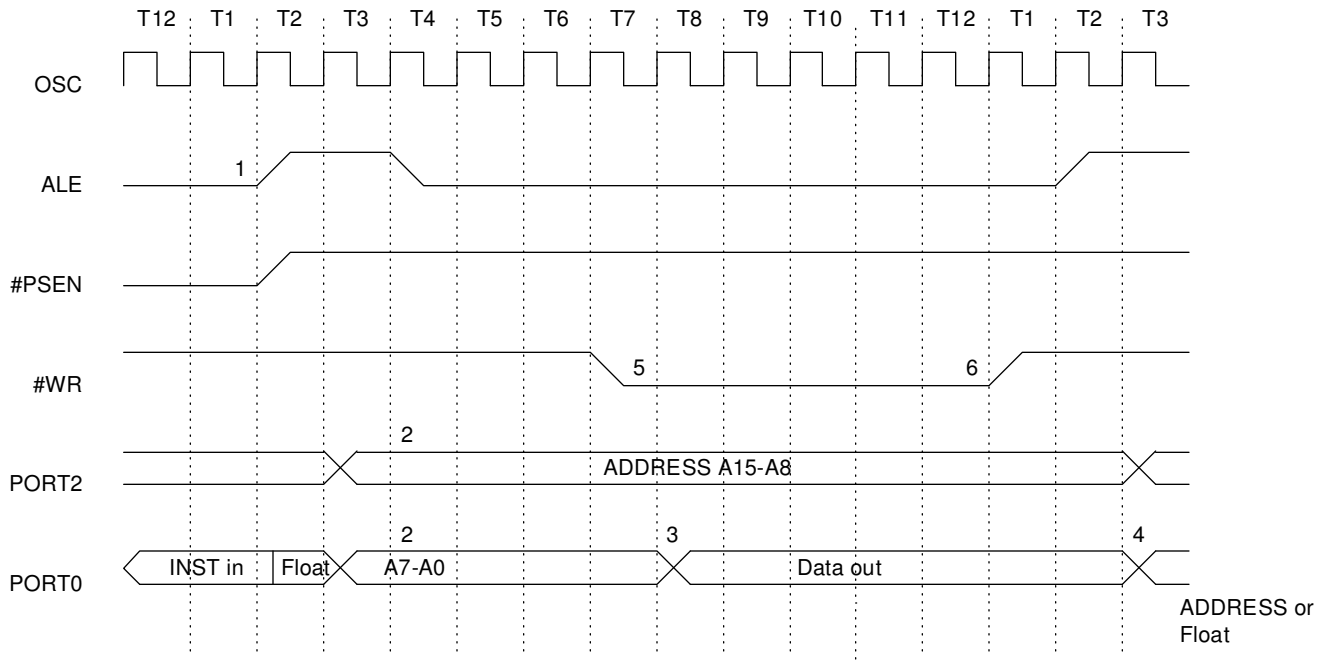
FIGURE 26: PROGRAM MEMORY READ CYCLE



Data Memory Write Cycle Timing

The following timing diagram shows the signal timing during Data Memory Write Cycle.

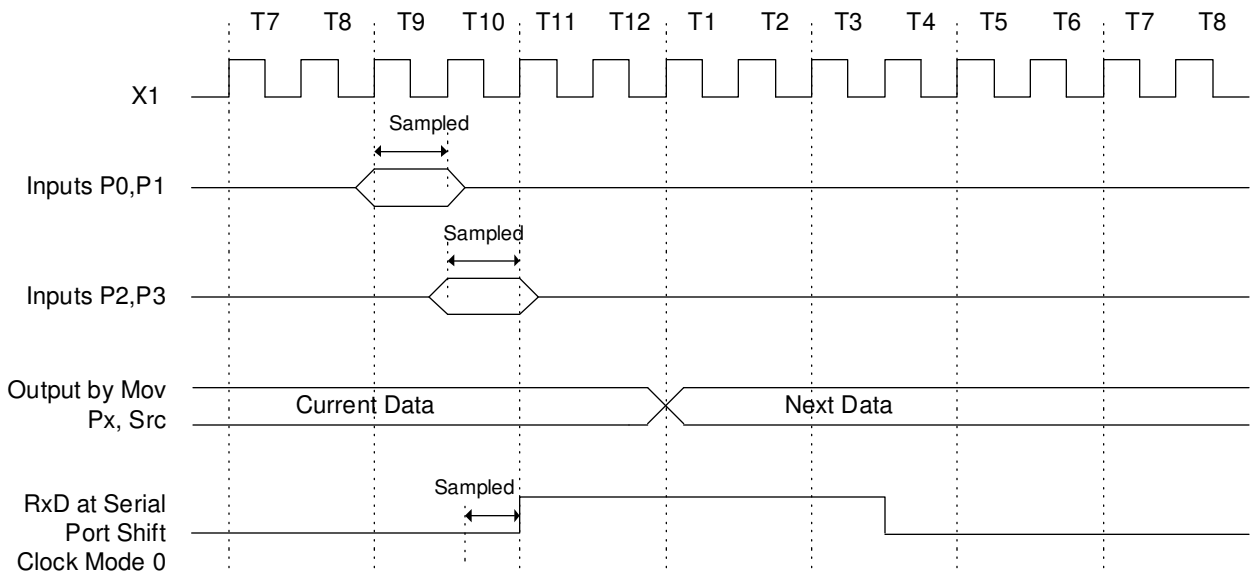
FIGURE 27: DATA MEMORY WRITE CYCLE TIMING



I/O Ports Timing

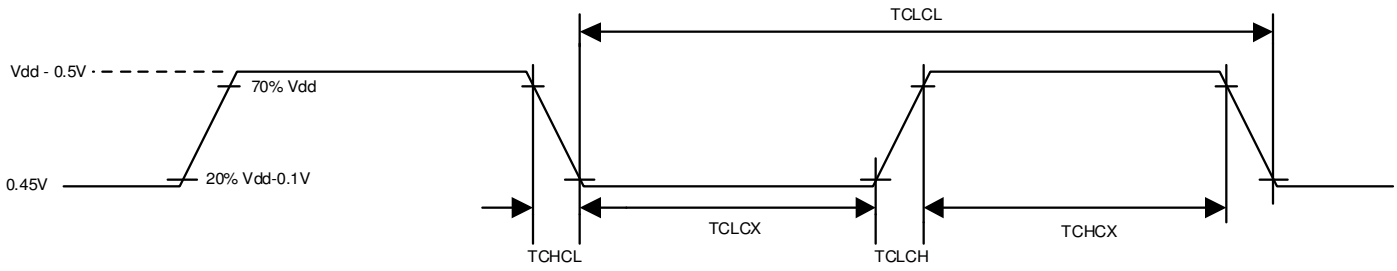
The following timing diagram shows I/O Port Timing.

FIGURE 28: I/O PORTS TIMING



Timing Requirement of the External Clock (VSS = 0v is assumed)

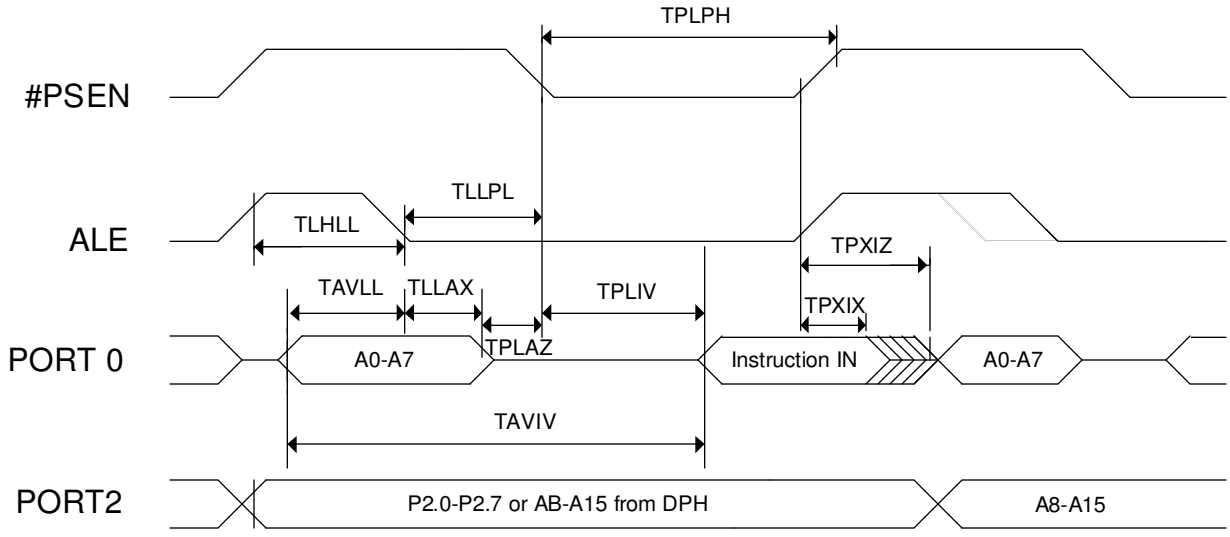
FIGURE 29: TIMING REQUIREMENT OF EXTERNAL CLOCK (VSS= 0.0V IS ASSUMED)



External Program Memory Read Cycle

The following timing diagram shows the signal timing during an External Program Memory Read Cycle.

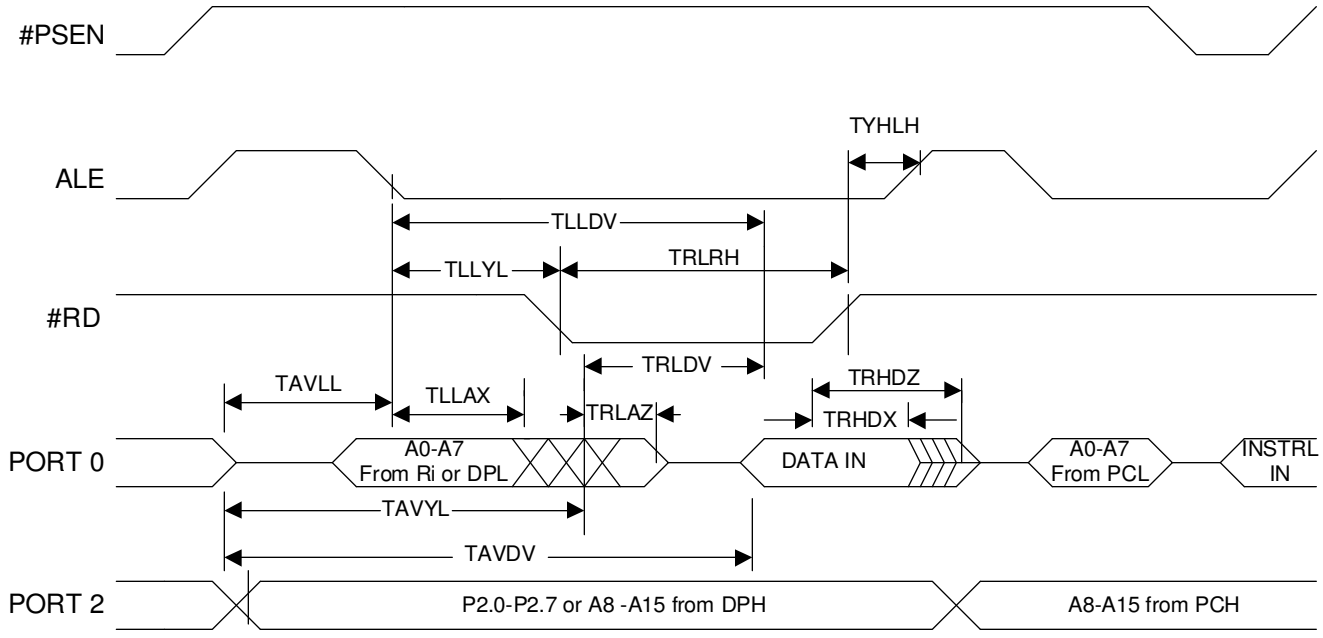
FIGURE 30: EXTERNAL PROGRAM MEMORY READ CYCLE



External Data Memory Read Cycle

The following timing diagram shows the signal timing during an External Data Memory Read Cycle.

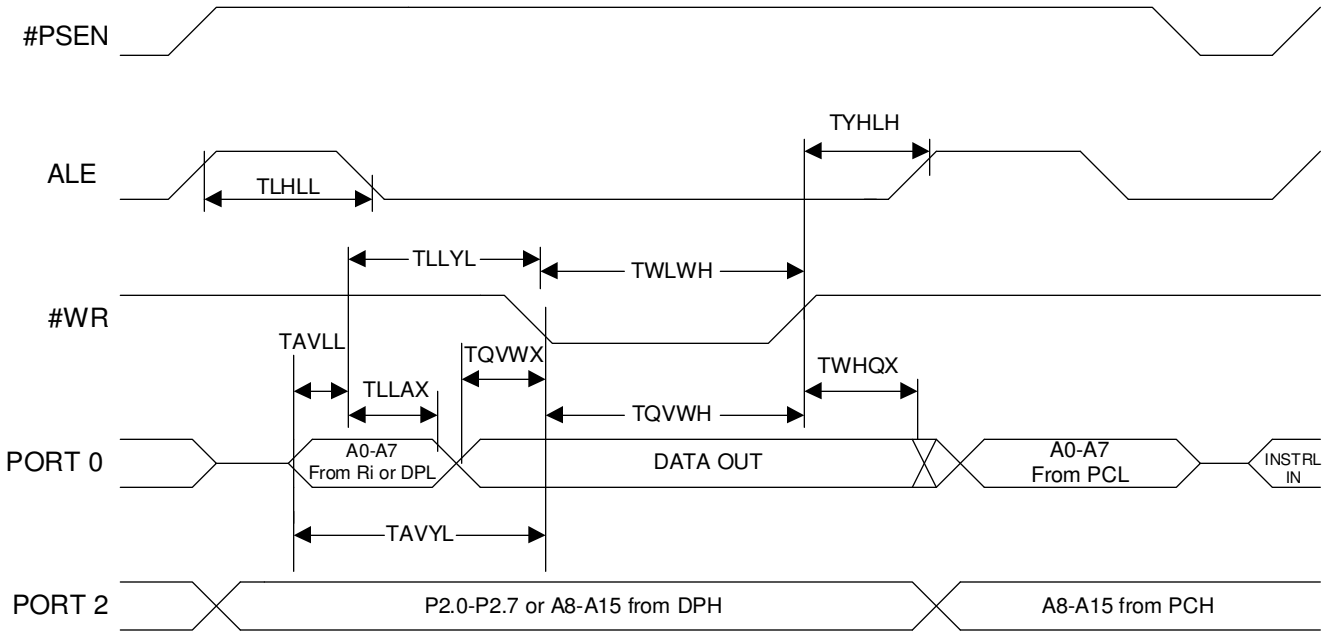
FIGURE 31: EXTERNAL DATA MEMORY READ CYCLE



External Data Memory Write Cycle

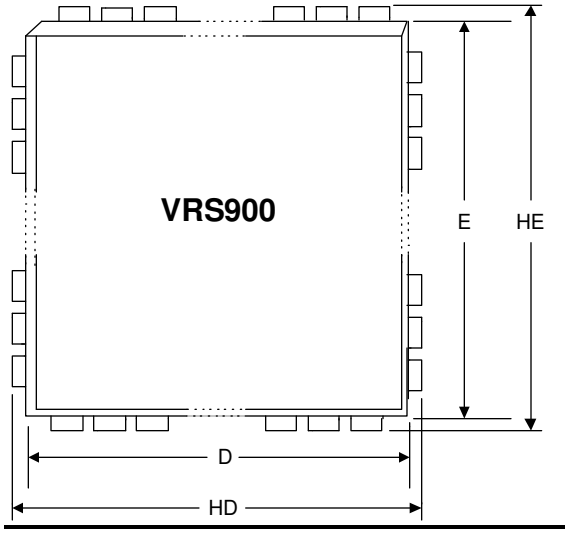
The following timing diagram shows the signal timing during an External Data Memory Write Cycle.

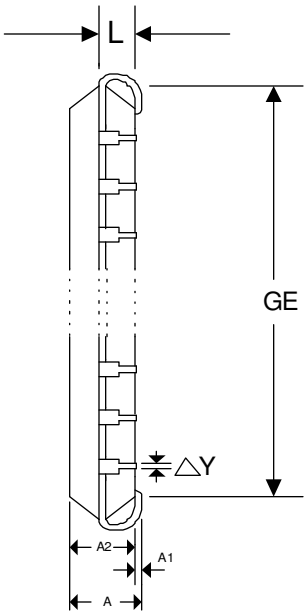
FIGURE 32: EXTERNAL DATA MEMORY WRITE CYCLE

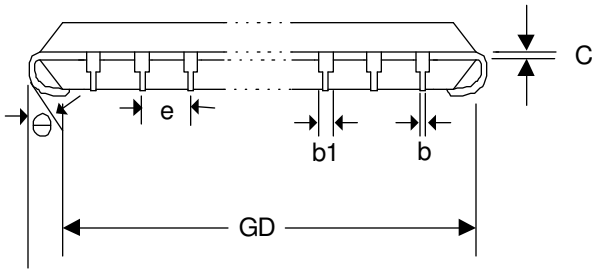


.

Plastic Chip Carrier (PLCC)





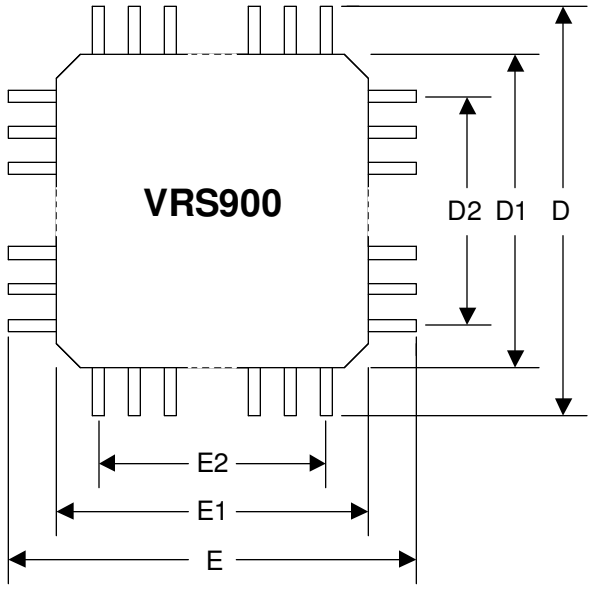


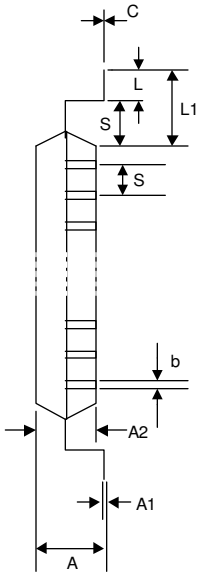
Note:
1. Dimensions D & E do not include interlead Flash.
2. Dimension B1 does not include dambar protrusion/intrusion.
3. Controlling dimension: Inch
4. General appearance spec should be based on final visual inspection spec.

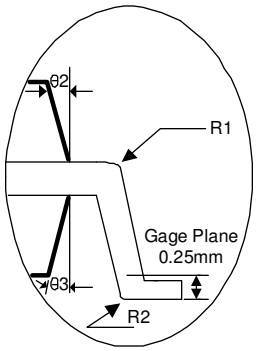
TABLE 57: DIMENSIONS OF PLCC-44 CHIP CARRIER

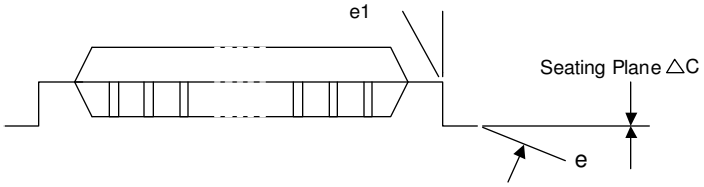
Symbol	Dimension in inch	Dimension in mm
	Minimal/Maximal	Minimal/Maximal
A	-/0.185	-/4.70
Al	0.020/-	0.51/
A2	0.145/0.155	3.68/3.94
bl	0.026/0.032	0.66/0.81
b	0.016/0.022	0.41/0.56
C	0.008/0.014	0.20/0.36
D	0.648/0.658	16.46/16.71
E	0.648/0.658	16.46/16.71
e	0.050 BSC	1.27 BSC
GD	0.590/0.630	14.99/16.00
GE	0.590/0.630	14.99/16.00
HD	0.680/0.700	17.27/17.78
HE	0.680/0.700	17.27/17.78
L	0.090/0.110	2.29/2.79
	-/0.004	-/0.10
y	/	/

Plastic Quad Flat Package (QFP)









Note:

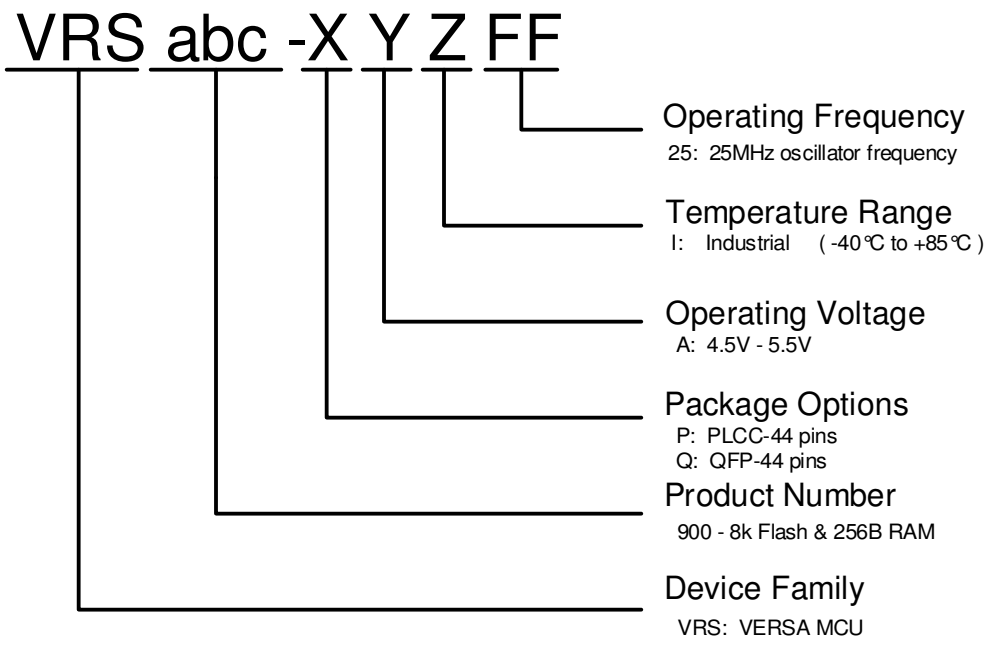
1. Dimensions D1 and E1 do not include mold protrusion.
2. Allowance protrusion is 0.25mm per side.
3. Dimensions D1 and E1 do not include mold mismatch and are determined datum plane.
4. Dimension b does not include dambar protrusion.
5. Allowance dambar protrusion shall be 0.08 mm total in excess of the b dimension at maximum material condition. Dambar cannot be located on the lower radius of the lead foot.

TABLE 58: DIMENSIONS OF QFP-44 CHIP CARRIER

Symbol	Dimension in in.	Dimension in mm
	Minimal/Maximal	Minimal/Maximal
A	-/0.100	-/2.55
Al	0.006/0.014	0.15/0.35
A2	0.071 / 0.087	1.80/2.20
b	0.012/0.018	0.30/0.45
c	0.004 / 0.009	0.09/0.20
D	0.520 BSC	13.20 BSC
D1	0.394 BSC	10.00 BSC
D2	0.315	8.00
E	0.520 BSC	13.20 BSC
E1	0.394 BSC	10.00 BSC
E2	0.315	8.00
e	0.031 BSC	0.80 BSC
L	0.029 / 0.041	0.73/1.03
L1	0.063	1.60
R1	0.005/-	0.13/-
R2	0.005/0.012	0.13/0.30
S	0.008/-	0.20/-
0	0˚/7˚	as left
1	0˚/ -	as left
2	10˚ REF	as left
3	7˚ REF	as left
C	0.004	0.10

Ordering Information

Device Number Structure



VRS900 Ordering Options

Device Number	Flash Size	RAM Size	Package Option	Voltage	Temperature	Frequency
VRS900-PAI25	8K	256B	PLCC-44	4.5V to 5.5V	-40°C to +85°C	25MHz
VRS900-QAI25	8K	256B	QFP-44	4.5V to 5.5V	-40°C to +85°C	25MHz

Disclaimers

Right to make change - Goal Semiconductor reserves the right to make changes to its products - including circuitry, software and services - without notice at any time. Customers should obtain the most current and relevant information before placing orders.
Use in applications - Goal Semiconductor assumes no responsibility or liability for the use of any of its products, and conveys no license or title under any patent, copyright or mask work right to these products and makes no representations or warranties that these products are free from patent, copyright or mask work right infringement unless otherwise specified. Customers are responsible for product design and applications using Goal Semiconductor parts. Goal Semiconductor assumes no liability for applications assistance or customer product design.
Life support – Goal Semiconductor products are not designed for use in life support systems or devices. Goal Semiconductor customers using or selling Goal products for use in such applications do so at their own risk and agree to fully indemnify Goal Semiconductor for any damages resulting from such applications.