Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	EBI/EMI, Ethernet, UART/USART
Peripherals	LED, WDT
Number of I/O	32
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	-
Oscillator Type	-
Operating Temperature	-40°C ~ 80°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	100-LQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/wiznet/w7100a-s2e-100

List of Tables

1.4.2.2 Timer

Pin name	Pin number		I/O	Pu/Pd	Description
	100pin	64pin			
Timer0, 1 Interface					
T0	9	-	I	Pu	Timer0 external clock input
T1	10	-	I	Pu	Timer1 external clock input
GATE0	11	-	I	Pd	Timer0 gate control
GATE1	12	-	I	Pd	Timer1 gate control
Timer2 Interface					
T2	13	-	I	Pu	Timer2 external clock input
T2EX	14	-	I	Pu	Timer2 Capture/Reload trigger

1.4.2.3 UART

Pin name	Pin number		I/O	Pu/Pd	Description
	100pin	64pin			
RXD	15	10	I	-	Serial receiver
TXD	17	12	O	-	Serial transmitter

1.4.2.4 DoCD™ Compatible Debugger

Pin name	Pin number		I/O	Pu/Pd	Description
	100pin	64pin			
DCDCLK	18	13	O	-	DoCD clock
DCDDI	19	14	I	Pu	DoCD data input
DCDDO	20	15	O		DoCD data output

1.4.2.5 Interrupt / Clock

Pin name	Pin number		I/O	Pu/Pd	Description
	100pin	64pin			
nINT0	22	17	I	-	External interrupt0
nINT1	23	-	I	Pu	External interrupt1
nINT2	24	-	I	Pu	External interrupt2
nINT3	25	-	I	Pu	External interrupt3
XTLN0	61	40	O	-	Crystal output for WIZnet Core, A parallel-resonant 25MHz crystal or ceramic is connected. If use oscillator, this pin can be floated.
XTLP0	62	41	I	-	Crystal input for WIZnet Core, A parallel-

					resonant 25MHz crystal or ceramic is connected. If use oscillator, this pin connected with 1.8V output of OSC.
XTLN1	67	45	O	-	Crystal output for MCU core, A parallel-resonant 11.0592MHz crystal or ceramic is connected. If oscillator is uesd, this pin can be floated.
XTLP1	66	44	I	-	Crystal input for MCU core, A parallel-resonant 11.0592MHz crystal or ceramic is connected. If oscillator is used, this pin is connected with 1.8V output of OSC.

1.4.2.6 GPIO

Pin name	Pin number		I/O	Pu/Pd	Description
	100pin	64pin			
P0.0	81	52	IO	-	Port0 input/output, Ext Memory Data0, Addr0
P0.1	82	53	IO	-	Port0 input/output, Ext Memory Data1, Addr1
P0.2	84	55	IO	-	Port0 input/output, Ext Memory Data2, Addr2
P0.3	85	56	IO	-	Port0 input/output, Ext Memory Data3, Addr3
P0.4	86	57	IO	-	Port0 input/output, Ext Memory Data4, Addr4
P0.5	88	59	IO	-	Port0 input/output, Ext Memory Data5, Addr5
P0.6	89	60	IO	-	Port0 input/output, Ext Memory Data6, Addr6
P0.7	90	61	IO	-	Port0 input/output, Ext Memory Data7, Addr7
P1.0	26	18	IO	-	Port1 input/output, Ext Memory Addr0
P1.1	27	19	IO	-	Port1 input/output, Ext Memory Addr1
P1.2	28	20	IO	-	Port1 input/output, Ext Memory Addr2
P1.3	29	21	IO	-	Port1 input/output, Ext Memory Addr3
P1.4	30	22	IO	-	Port1 input/output, Ext Memory Addr4
P1.5	31	23	IO	-	Port1 input/output, Ext Memory Addr5
P1.6	32	24	IO	-	Port1 input/output, Ext Memory Addr6
P1.7	33	25	IO	-	Port1 input/output, Ext Memory Addr7
P2.0	91	62	IO	-	Port2 input/output, Ext Memory Addr8
P2.1	93	64	IO	-	Port2 input/output, Ext Memory Addr9
P2.2	94	1	IO	-	Port2 input/output, Ext Memory Addr10
P2.3	95	-	IO	-	Port2 input/output, Ext Memory Addr11
P2.4	96	-	IO	-	Port2 input/output, Ext Memory Addr12
P2.5	97	-	IO	-	Port2 input/output, Ext Memory Addr13
P2.6	98	-	IO	-	Port2 input/output, Ext Memory Addr14

P2.7	99	-	IO	-	Port2 input/output, Ext Memory Addr15
P3.0	34	-	IO	-	Port3 input/output, Ext Memory Addr16
P3.1	35	-	IO	-	Port3 input/output, Ext Memory Addr17
P3.2	37	-	IO	-	Port3 input/output, Ext Memory Addr18
P3.3	39	-	IO	-	Port3 input/output, Ext Memory Addr19
P3.4	40	-	IO	-	Port3 input/output, Ext Memory Addr20
P3.5	41	-	IO	-	Port3 input/output, Ext Memory Addr21
P3.6	42	-	IO	-	Port3 input/output, Ext Memory Addr22
P3.7	43	-	IO	-	Port3 input/output, Ext Memory Addr23

Note: User can control the GPIO I/O driving voltage using PxPU/PxPD SFR.

Note: In that case, GPIO0~3 is used to transfer External memory address and data. Please refer to the '2.3 External Data Memory Access'

1.4.2.7 External Memory Interface

Pin name	Num	I/O Type	Description
ALE	78	O	Data memory address bus [7:0] latch enable
nWR	79	OL	External data memory write
nRD	80	OL	External data memory read

Note: When user using External memory by standard 8051 interface, P0[7:0] can transfer Data[7:0] or Address[7:0] by ALE pin control.

1.4.2.8 Media Interface

Pin name	Pin number		I/O	Pu/Pd	Description
	100pin	64pin			
TXON	52	32	O	-	TXON/TXOP Signal Pair, The differential data is transmitted to the media on the TXON/TXOP signal pair
TXOP	53	33	O	-	
RXIN	55	35	I	-	RXIN/RXIP Signal Pair, The differential data from the media is received on the RXIN/RXIP Signal pair
RXIP	56	36	I	-	
RESETBG	59	38	I	-	PHY Off-chip resistor, Connect a resistor of 12.3 kΩ±1% to the ground. Refer to the "Reference schematic"

For the best performance,

1. Make the length of RXIP / RXIN signal pair (RX) same if possible.
2. Make the length of TXOP / TXON signal pair (TX) same if possible.
3. Locate the RXIP and RXIN signal as near as possible.

2.2.1 Data Memory Wait States

The Data Memory wait states are managed by CKCON(0x8E). The number of wait states is fixed to the value stored inside CKCON register. Please refer to the section 2.5.10 'New & Extended SFR' for more detailed information.

2.3 External Data Memory Access

The external address pin and data pin has two access modes. The first mode is to use latch to address line in standard 8051. And the second method is directly connecting all lines to address line. Also user can use address pin and data pin as GPIO (General Purpose I/O). Please refer to the section 10 'Electrical specification' for the speed of external memory accessing.

Table 2.1 External memory access mode

Mode	EM[2:0]	P0	P1	P2	P3
Standard 1	001	Addr[7:0]/Data[7:0]	GPIO	Addr[15:8]	GPIO
Standard 2	011	Addr[7:0]/Data[7:0]	GPIO	Addr[15:8]	Addr[23:16]
Direct 1	101	Data[7:0]	Addr[7:0]	Addr[15:8]	GPIO
Direct 2	111	Data[7:0]	Addr[7:0]	Addr[15:8]	Addr[23:16]

2.3.1 Standard 8051 Interface

This method is same as external interface of general 8051. But the range of accessible address is changed refer to the setting of EM[2:0] (External Memory Mode) which spaced WCONF(0xFF) of SFR register. When user sets the EM[2:0] to "001", the port0 is used as address/data bus and the port2 is used as upper side address (A[15:8]). The port1 and port3 is used as GPIO.

Note :

The external memory interface using standard 8051 interface has an erratum where the ALE signal turns on after the WR/RD signal is enabled; the ALE signal is supposed to turn on before the WR/RD signal is enabled. To resolve this erratum, OR the WR/RD signal with ALE signal and connect it to the external memory's nWR and nOE.

For more details, please refer to document 'W7100A Errata sheet - Erratum3.'

It is shown the figure below.

2.5.3 Data Pointer Extended Registers

Data Pointer Extended registers, DPX0, DPX1 and MXAX, hold the most significant part of memory addresses when accessing to data located above 64KB. After reset, DPX0, DPX1, and MXAX restores to the default value 0x00.

DPX0 (0x93)

7	6	5	4	3	2	1	0	Reset
DPXP.7	DPX.6	DPX.5	DPX.4	DPX.3	DPX.2	DPX.1	DPX.0	0x00

Figure 2.16 Data Pointer Extended Register

DPX1 (0x95)

7	6	5	4	3	2	1	0	Reset
DPX1.7	DPX1.6	DPX1.5	DPX1.4	DPX1.3	DPX1.2	DPX1.1	DPX1.0	0x00

Figure 2.17 Data Pointer Extended Register

MXAX (0xEA)

7	6	5	4	3	2	1	0	Reset
MXAM.7	MXAX.6	MXAX.5	MXAX.4	MXAX.3	MXAX.2	MXAX.1	MXAX.0	0x00

Figure 2.18 MOVX @RI Extended Register

When MOVX instruction uses DPTR0/DPTR1 register, the most significant part of the address A[23:16] is always equal to the content of DPX0(0x93)/DPX1(0x95). When MOVX instruction uses R0 or R1 register, the most significant part of the address A[23:16] is always equal to the content of MXAX(0xEA) while another A[15:8] is always equal to P2(0xA0) contents.

2.5.4 Data Pointer Registers

Dual data pointer registers are implemented to speed up data block copying. DPTR0 and DPTR1 are located in four SFR addresses. Active DPTR register is selected by SEL bit (0x86.0). If the SEL bit set to '0', DPTR0 (0x83:0x82) is selected, otherwise DPTR1 (0x85:0x84) is selected.

DPTR0(0x83:0x82)

DPH0(0x83)								DPL0(0x82)								Reset
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	0x0000

Figure 2.19 Data Pointer Register DPTR0

DPTR1(0x85:0x84)

DPH1(0x85)								DPL1(0x84)								Reset
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	0x0000

Figure 2.20 Data Pointer 1 Register DPTR1

0	0	-
0	1	Pull-down
1	0	Pull-up
1	1	Keep

P0_PD(0xE3): GPIO0 Pull-down register, the value '1' pull-down the related pin.

P0_PD (0xE3)

7	6	5	4	3	2	1	0	Reset
Port0[7]	Port0[6]	Port0[5]	Port0[4]	Port0[3]	Port0[2]	Port0[1]	Port0[0]	0x00

Figure 4.6 Port0 Pull-down register

P1_PD(0xE4): GPIO1 Pull-down register, the value '1' pull-down the related pin.

P1_PD (0xE4)

7	6	5	4	3	2	1	0	Reset
Port1[7]	Port1[6]	Port1[5]	Port1[4]	Port1[3]	Port1[2]	Port1[1]	Port1[0]	0x00

Figure 4.7 Port1 Pull-down register

P2_PD(0xE5): GPIO2 Pull-down register, the value '1' pull-down the related pin.

P2_PD (0xE5)

7	6	5	4	3	2	1	0	Reset
Port2[7]	Port2[6]	Port2[5]	Port2[4]	Port2[3]	Port2[2]	Port2[1]	Port2[0]	0x00

Figure 4.8 Port2 Pull-down register

P3_PD(0xE6): GPIO3 Pull-down register, the value '1' pull-down the related pin.

P3_PD (0xE6)

7	6	5	4	3	2	1	0	Reset
Port3[7]	Port3[6]	Port3[5]	Port3[4]	Port3[3]	Port3[2]	Port3[1]	Port3[0]	0x00

Figure 4.9 Port3 Pull-down register

P0_PU(0xEB): GPIO0 Pull-up register, the value '1' means pull-up the related pin.

P0_PU (0xEB)

7	6	5	4	3	2	1	0	Reset
Port0[7]	Port0[6]	Port0[5]	Port0[4]	Port0[3]	Port0[2]	Port0[1]	Port0[0]	0x00

Figure 4.10 Port0 Pull-up register

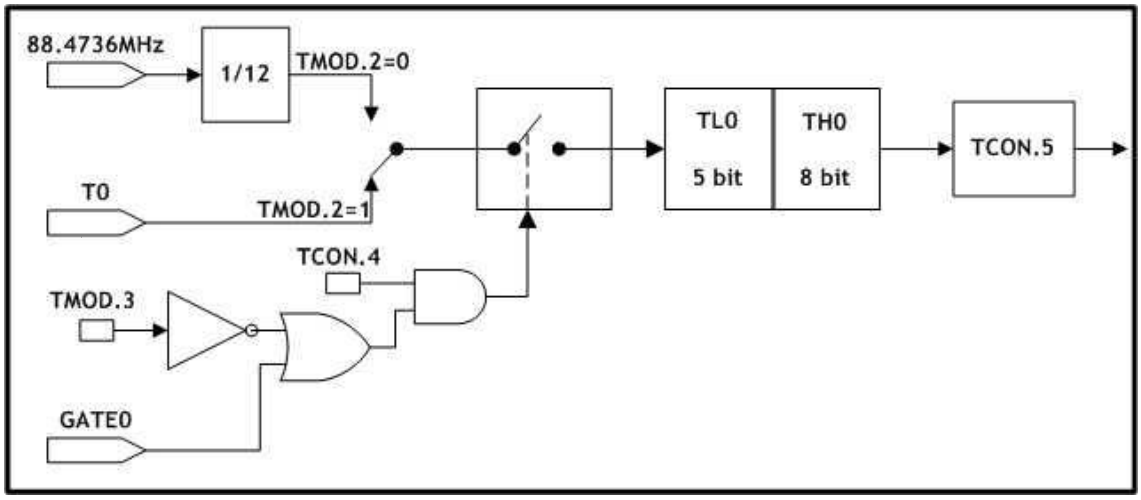


Figure 5.6 Timer Counter0, Mode0: 13-Bit Timer/Counter

5.1.4 Timer0 – Mode1

Mode1 is the same as Mode0, except that the timer register is running with all 16 bits. Mode1 is shown in the Figure below.

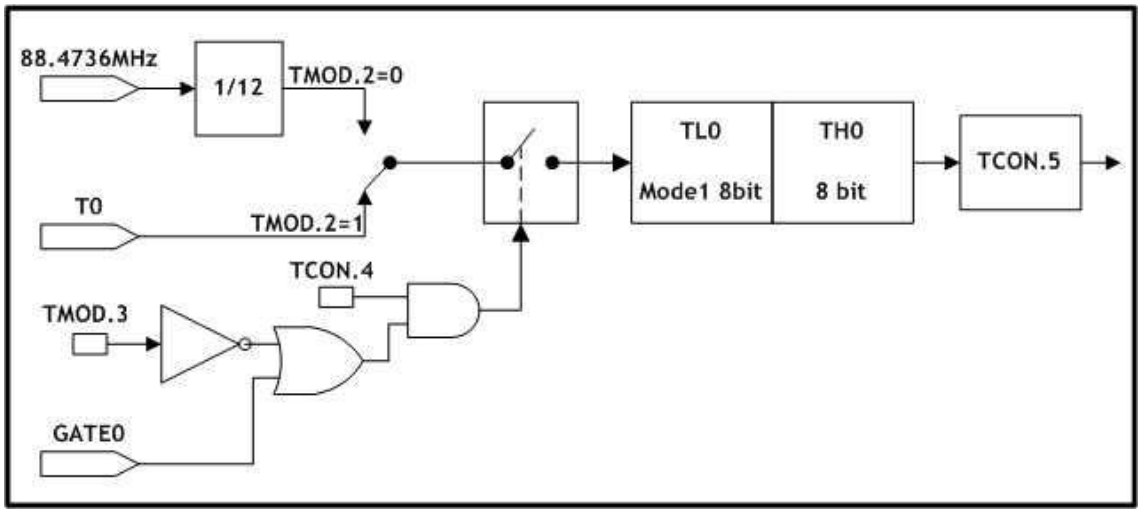


Figure 5.7 Timer/Counter0, Mode1: 16-Bit Timer/Counter

5.1.5 Timer0 – Mode2

Mode2 configures the timer register as a 8-bit counter TL0 with automatic reload as shown in the Figure below. During an overflow from TL0, it sets TF0 and reloads the contents of TH0 into TL0. TH0 remains unchanged after the reload is completed.

5.1.9 Timer1 – Mode2

Mode2 configures timer register as 8-bit counter TL1, with automatic reload as shown in Figure below. Overflow from TL1 only sets TF1, but also automatically reloads TL1 with the contents of TH1. The reload leaves TH1 unchanged.

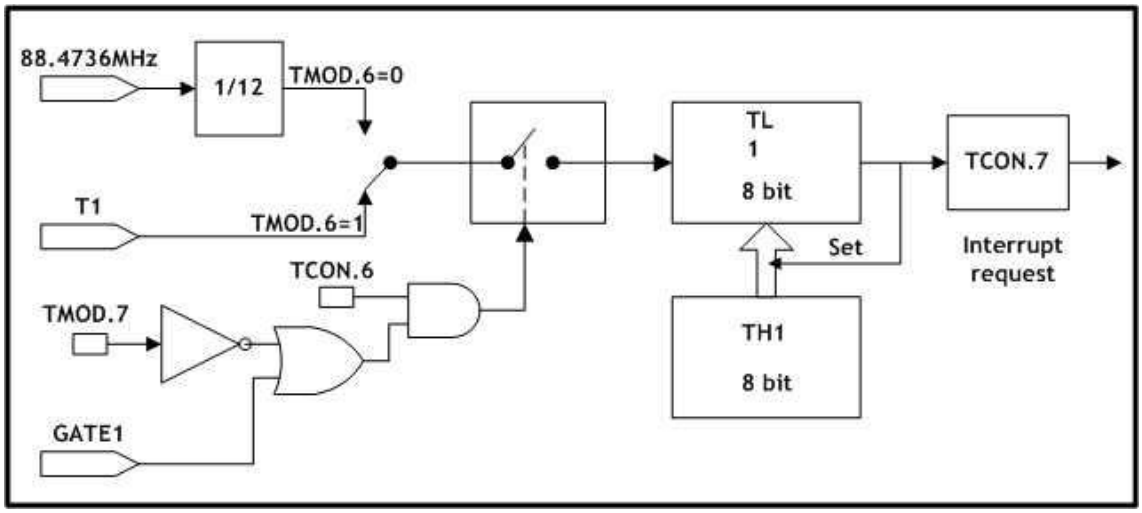


Figure 5.12 Timer/Counter1, Mode2: 8-Bit Timer/Counter with Auto-Reload

5.1.10 Timer1 – Mode3

Timer1 in Mode3 holds counting. The effect is the same as setting TR1 = 0 because it is used for Timer0-Mode3. For more detail, please refer to the section 5.1.6 'Timer0-Mode3'.

EIP (0xF8)

7	6	5	4	3	2	1	0	Reset
-	-	-	PWDI	PINT5	PINT4	PINT3	PINT2	0x00

Figure 7.4 Extended interrupt Priority Register

Note: PWDI - Watchdog priority level control (high level at 1)

Unimplemented bit - Read as 0 or 1

WDCON (0xD8)

7	6	5	4	3	2	1	0	Reset
-	-	-	-	WDIF	WTRF	EWT	RWT	0x00

Figure 7.5 Watchdog Control Register

Note: WDIF - Watchdog Interrupt Flag. WDIF in conjunction with Enable Watchdog Interrupt bit (EIE.4) and EWT, indicates if a watchdog timer event has occurred and what action should be taken. This bit must be cleared by software before exiting the interrupt service routine or another interrupt is generated. Setting WDIF in software will generate a watchdog interrupt if enabled. User must use 'Timed **Access Register**' when clear this WDIF bit. Please refer to the section 7.8 '**Timed Access**' procedure.

All of the bits that generate interrupts can be set or cleared by software, with the same result by hardware. That is, interrupts can be generated or cancelled by software.

Table 7.1 Watchdog Interrupt

Interrupt Flag	Function	Active Level/Edge	Flag Reset	Vector	Natural Priority
WDIF	Internal, Watchdog	-	Software	0x63	11

7.3 Watchdog Timer Reset

The Watchdog Timer reset operates as follows. Once the timeout interval is initialized, the system restarts the Watchdog first by using RWT. Then, the reset mode is enabled by the EWT (Enable Watchdog Timer reset = WDCON.1) bit. Before the timer reaches the user selected terminal value, the software can set the RWT (Reset Watchdog Timer = WDCON.0) bit. If RWT is set before the timeout is reached, the timer will start over. If the timeout is reached without RWT being set, the Watchdog will reset the MCU. The Hardware automatically clears RWT after sets the RWT by software. When a reset occurs, the WTRF (Watchdog Timer reset Flag = WDCON.2) will automatically set to indicate the cause of the reset; however, software must clear this bit manually. Do not use the watchdog timer reset, because it can cause the abnormal operation the RESET. Instead, we recommend you to use watchdog timer interrupt.

8 TCPIPCore

8.1 Memory Map

TCPIPCore is composed of Common Register, SOCKET Register, TX Memory, and RX Memory as shown below.

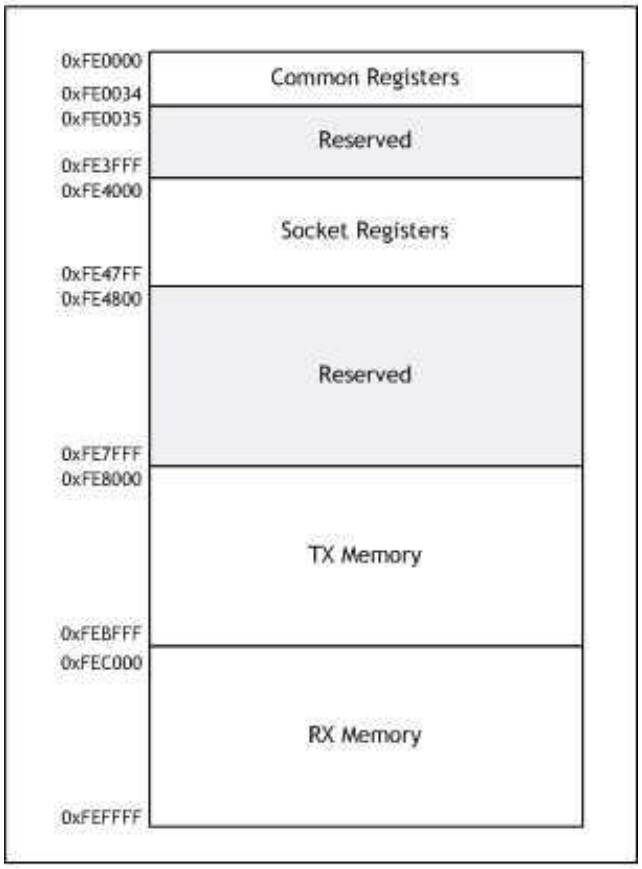


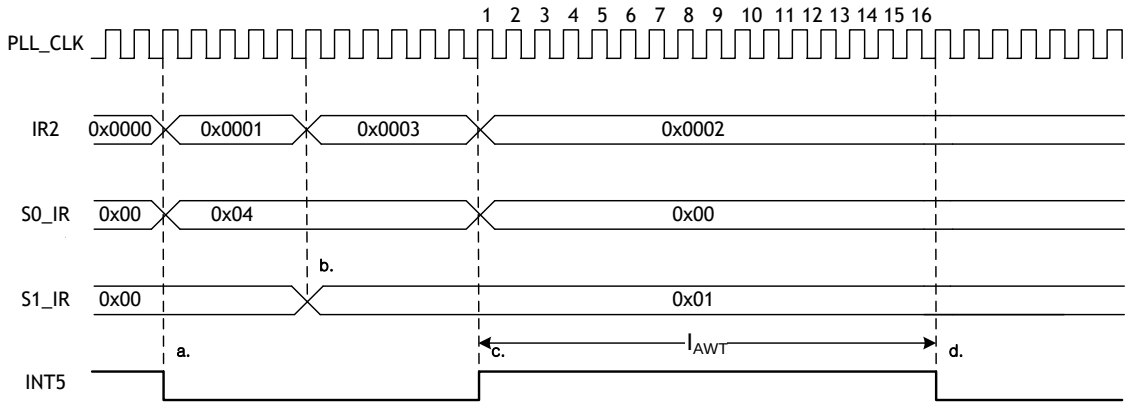
Figure 8.1 TCPIPCore Memory Map

8.2 Registers list

8.2.1 Common Registers

Address offset	Symbol	Description
0xFE0000	MR	Mode Register
0xFE0001	GAR0	GAR (Gateway Address Register)
0xFE0002	GAR1	
0xFE0003	GAR2	
0xFE0004	GAR3	
0xFE0005	SUBR0	SUBR (Subnet Mask Register)
0xFE0006	SUBR1	

0xFE440B	S4_DHAR5	
0xFE440C	S4_DIPR0	S4_DIPR (SOCKET 4 Destination IP Address Register)
0xFE440D	S4_DIPR1	
0xFE440E	S4_DIPR2	
0xFE440F	S4_DIPR3	
0xFE4410	S4_DPORT0	S4_DPORT (SOCKET 4 Destination Port Register)
0xFE4411	S4_DPORT1	
0xFE4412	S4_MSSR0	S4_MSSR (SOCKET 4 Maximum Segment Size Register)
0xFE4413	S4_MSSR1	
0xFE4414	S4_PROTO	SOCKET 4 Protocol of IP Header Field Register in IP raw mode
0xFE4415	S4_TOS	SOCKET 4 IP Type of Service(TOS) Register
0xFE4416	S4_TTL	SOCKET 4 IP Time to Live(TTL) Register
0xFE4417 ~ 0xFE441D		Reserved
0xFE441E	S4_RXMEM_SIZE	SOCKET 4 Receive Memory Size Register
0xFE441F	S4_TXMEM_SIZE	SOCKET 4 Transmit Memory Size Register
0xFE4420	S4_TX_FSR0	S4_TX_FSR (SOCKET 4 Transmit Free Memory Size Register)
0xFE4421	S4_TX_FSR1	
0xFE4422	S4_TX_RD0	S4_TX_RD (SOCKET 4 Transmit Memory Read Pointer Register)
0xFE4423	S4_TX_RD1	
0xFE4424	S4_TX_WR0	S4_TX_WR (SOCKET 4 Transmit Memory Write Pointer Register)
0xFE4425	S4_TX_WR1	
0xFE4426	S4_RX_RSR0	S4_RX_RSR (SOCKET 4 Received Data Size Register)
0xFE4427	S4_RX_RSR1	
0xFE4428	S4_RX_RD0	S4_RX_RD (SOCKET 4 Receive Memory Read Pointer Register)
0xFE4429	S4_RX_RD1	
0xFE442A	S4_RX_WR0	S4_RX_WR (SOCKET 4 Receive Memory Write Pointer Register)
0xFE442B	S4_RX_WR1	
0xFE442C	S4_IMR	SOCKET 4 Interrupt Mask Register
0xFE442D	S4_FRAG0	SOCKET 4 Fragment Field Value in IP Header Register
0xFE442E	S4_FRAG1	



a. At the socket 0, assume an interrupt occurs (S0_IR(3) = '1') and corresponding IR2 bit is set as '1' (IR(S0_IR) = '1'). Then the internal INT5 signal is asserted low.

b. Also assume an interrupt continually occurs (S1_IR(0) = '1') on the socket1 and corresponding IR bit set as '1' (IR(S1_IR) = '1').

c. When the Host clears S0_IR(S0_IR = 0x00), the corresponding IR2 bit is also cleared (IR(S0_IR) = '0'). Internal INT5 signal will be de-asserted high(deactivated) from low(activaed).

d. When the S1_IR is cleared, but the corresponding IR2 is not 0x00 because of socket1 interrupt, internal INT5 signal should be asserted low.

However, as INTLEVEL is 0x000F, the internal INT5 signal is asserted after the I_{AWT}(16 PLL_CLK) time.

IR2 (W7100A SOCKET Interrupt Register)[R/W][0xFE0034][0x00]

IR2 is a Register which notifies the host that a W7100A SOCKET interrupt has occurred. When an interrupt occurs, the related bit in IR2 is enabled. In this case, the INT5 (nINT5: TCPIPcore interrupt) signal is asserted low until all of the bits of IR2 is '0'. Once the IR2 register is cleared out by using the Sn_IR bits, the INT5 signal is asserted high.

7	6	5	4	3	2	1	0
S7_INT	S6_INT	S5_INT	S4_INT	S3_INT	S2_INT	S1_INT	S0_INT

Bit	Symbol	Description
7	S7_INT	**Occurrence of SOCKET 7 Interrupt** When an interrupt occurs at SOCKET 7, it becomes '1'. This interrupt information is applied to S7_IR. This bit is automatically cleared when S7_IR is cleared to 0x00 by host.
6	S6_INT	**Occurrence of SOCKET 6 Interrupt** When an interrupt occurs at SOCKET 6, it becomes '1'. This interrupt

		changed to SOCK_PPPoE when S0_MR(P3:P0) = S0_MR_PPPoE and S0_CR = OPEN.

Below table shows the temporary status which can be observed when changing the Sn_SR.

Value	Symbol	Description
0x15	SOCK_SYNSENT	This status indicates that a connect-request(SYN packet) is sent to a "TCP SERVER". SYNSENT is an intermediate state between SOCK_INT and SOCK_ESTABLISHED. If connect-accept(SYN/ACK packet) is received from a "TCP SERVER", the SOCKET status automatically changes to SOCK_ESTBLISHED. However, if SYN/ACK packet is not received before TCP timeout occurs (Sn_IR(TIMEOUT)='1'), the status is changed to SOCK_CLOSED.
0x16	SOCK_SYNRECV	This status indicate that a connect-request(SYN packet) is received from a "TCP CLIENT". The socket status changes to SOCK_ESTABLISHED when W7100A successfully transmits connect-accept (SYN/ACK packet) to a "TCP CLIENT". If W7100A fails to send and TCP_{TO} occurs (Sn_IR(TIMEOUT)='1'), the status is changed to SOCK_CLOSED.
0x18	SOCK_FIN_WAIT	These statues show the process of terminating a connection. If the termination succeeds or Timeout interrupt is asserted, the socket status is changed to SOCK_CLOSED.
0x1A	SOCK_CLOSING	
0X1B	SOCK_TIME_WAIT	
0X1D	SOCK_LAST_ACK	
0x01	SOCK_ARP	This status indicates an ARP-request is being transmitted to a peer in order to acquire destination hardware address. It appears when the SEND command is used in UDP, IP RAW, and TCP mode, the socket status changes to SOCK_ARP. If the hardware address is successfully acquired from the destination (ARP-response is received), the socket status changes to SOCK_UDP, SOCK_IPRAW or SOCK_SYNSENT. On the other hand, when W7100A fails to acquire the hardware address and an ARP timeout occurs (Sn_IR(TIMEOUT)= '1'), the socket status returns to the previous state in UDP mode and IP RAW mode. In TCP mode, the socket status goes to the SOCK_CLOSED state.

9.2.1.1 TCP SERVER

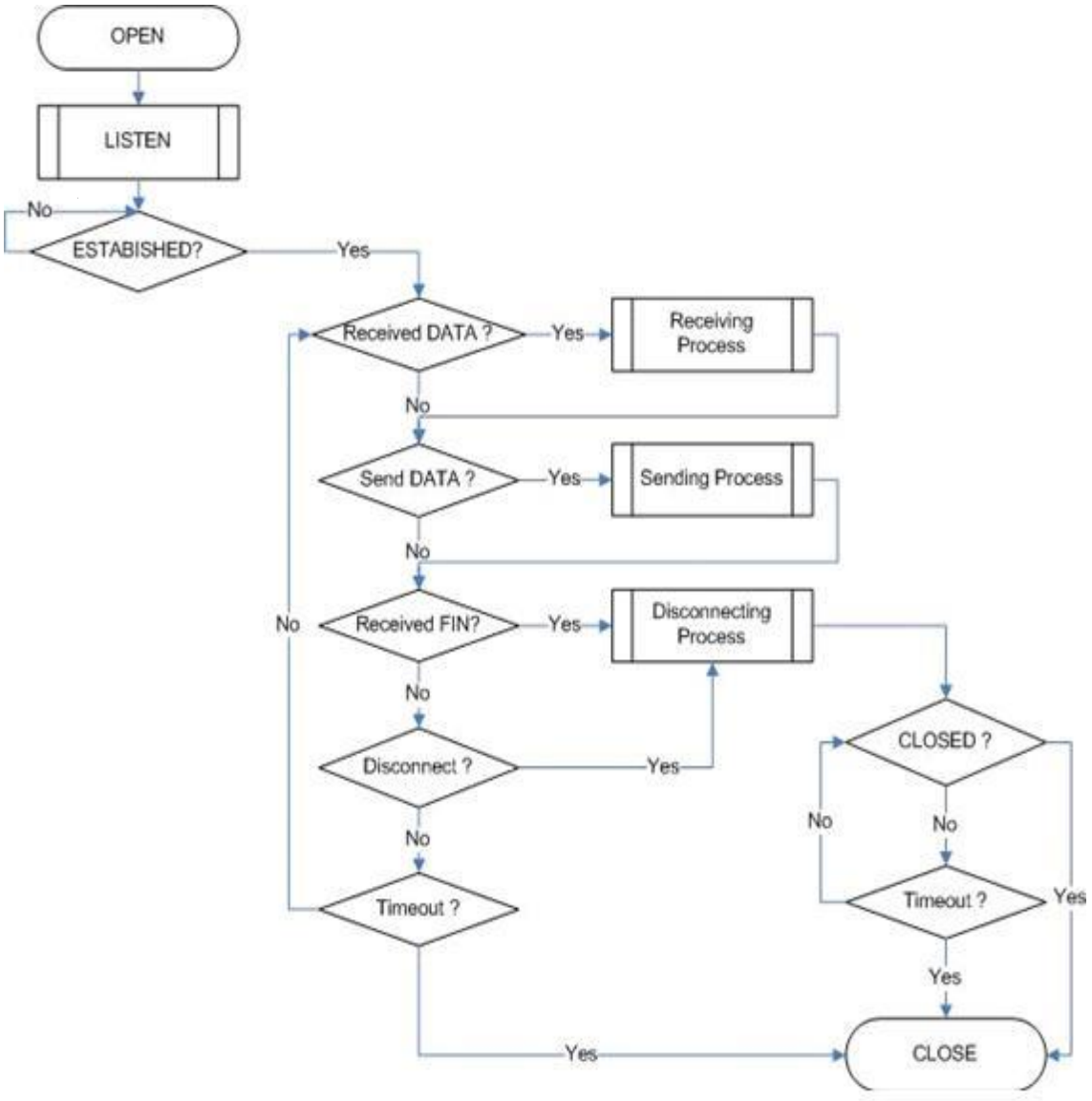


Figure 9.3 "TCP SERVER" Operation Flow

■ SOCKET Initialization

SOCKET initialization is required for TCP data communication. The initialization is opening the SOCKET. The SOCKET opening process selects one SOCKET from 8 SOCKETS of the W7100A, and sets the protocol mode (Sn_MR(P3:P0)) and Sn_PORT0 which is source port number (Listen port number in "TCP SERVER") in the selected SOCKET, and then executes OPEN command. After the OPEN command, if the status of Sn_SR is changed to SOCK_INIT, the SOCKET initialization process is completed.

The SOCKET initialization process is identically applied in "TCP SEVER" and "TCP CLIENT". The Initialization process of SOCKET *n* in TCP mode is shown below.

```
{
START:
  Sn_MR = 0x0001;                    // sets TCP mode
```

10 Electrical Specification

10.1 Absolute Maximum Ratings

Symbol	Parameter	Rating	Unit
V_{DD}	DC supply voltage	-0.5 to 3.6	V
V_{IN}	DC input voltage	-0.5 to 5.5 (5V tolerant)	V
V_{OUT}	DC output voltage	0 to 3.3 (GPIO)	V
		-0.5 to 3.6 (Others)	
I_{IN}	DC input current	±5	mA
I_{OUT}	DC output current	2 to 8	mA
T_{OP}	Operating temperature	-40 to 80	°C
T_{STG}	Storage temperature	-55 to 125	°C

***COMMENT**: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage.

10.2 DC Characteristics

Symbol	Parameter	Test Condition	Min	Typ	Max	Unit
V_{DD}	DC Supply voltage	Junction temperature -55°C ~ 125°C	3.0	3.3	3.6	V
V_{IH}	High level input voltage		2.0		5.5	V
V_{IL}	Low level input voltage		-0.5		0.8	V
V_{OH}	High level output voltage	I_{OH} = 8mA	2.4			V
V_{OL}	Low level output voltage	I_{OL} = 8mA			0.4	V
I_{lkg}	Input Leakage Current	V_{IN} = V_{DD} or 0	-10	±1	10	μA
	Input Leakage Current with pull-up resistance	V_{IN} = 0	-15	-45	-85	μA
	Input Leakage Current with pull-down resistance	V_{IN} = V_{DD}	15	45	85	μA
I_{OZ}	Tri-state output leakage current	V_{OUT} = V_{DD}	2		8	μA
I_{OL}	Low level output current	V_{OL} = 0.8V, 25°C, V_{DD} = 3.3V		52		mA
I_{OH}	High level output current	V_{OL} = 2.4V, 25°C, V_{DD} = 3.3V		52		mA

13.2.2 Subtraction

■ Immediate data

The following code performs immediate data (constant) subtraction from an 8-bit register.

Rx = Rx - *#n*

Mnemonic		Opcode	Byte	80C51 Cycle	W7100A Cycle	
					ISP / wizmemcpy	FLASH / user code
MOV	A, Rx	E8h – EFh	1	12	1	4
SUBB	A, *#n*	24h	2	12	2	4
MOV	Rx, A	F8h – FFh	1	12	1	4
		Sum :		36	4	12

■ Direct addressing

The following code performs direct addressing subtraction from an 8-bit register.

Rx = Rx – (dir)

Mnemonic		Opcode	Byte	80C51 Cycle	W7100A Cycle	
					ISP / wizmemcpy	FLASH / user code
MOV	A, Rx	E8h – EFh	1	12	1	4
SUBB	A, dir	25h	2	12	2	4
MOV	Rx, A	F8h – FFh	1	12	1	4
		Sum :		36	4	12

■ Indirect addressing subtraction

The following code performs indirect addressing subtraction from an 8-bit register.

Rx = Rx – (@Ry)

Mnemonic		Opcode	Byte	80C51 Cycle	W7100A Cycle	
					ISP / wizmemcpy	FLASH / user code
MOV	A, Rx	E8h – EFh	1	12	1	4
SUBB	A, @Ry	26h – 27h	1	12	2	4
MOV	Rx, A	F8h – FFh	1	12	1	4
		Sum :		36	4	12

■ Register addressing subtraction

The following code performs an 8-bit register from register subtraction.

RaRb = RaRb + RxRy

Mnemonic		Opcode	Byte	80C51 Cycle	W7100A Cycle	
					ISP / wizmemcpy	FLASH / user code
MOV	A, Rb	E8h - EFh	1	12	1	4
ADD	A, Ry	28h - 2Fh	1	12	1	4
MOV	Rb, A	F8h - FFh	1	12	1	4
MOV	A, Ra	E8h - EFh	1	12	1	4
ADDC	A, Rx	38h - 3Fh	1	12	1	4
MOV	Ra, A	F8h - FFh	1	12	1	4
		Sum :		72	6	24

13.3.2 Subtraction

The following code performs 16-bit subtraction. The first operand and result are located in registers pair RaRb. The second operand is located in registers pair RxRy.

RaRb = RaRb - RxRy

Mnemonic		Opcode	Byte	80C51 Cycle	W7100A Cycle	
					ISP / wizmemcpy	FLASH / user code
CLR	C	C3h	1	12	1	4
MOV	A, Rb	E8h - EFh	1	12	1	4
SUBB	A, Ry	98h - 9Fh	1	12	1	4
MOV	Rb, A	F8h - FFh	1	12	1	4
MOV	A, Ra	E8h - EFh	1	12	1	4
SUBB	A, Rx	98h - 9Fh	1	12	1	4
MOV	Ra, A	F8h - FFh	1	12	1	4
		Sum :		84	7	28

13.3.3 Multiplication

The following code performs 16-bit multiplication. The first operand and result are located in registers pair RaRb. The second operand is located in registers pair RxRy.

RaRb = RaRb * RxRy

Mnemonic		Opcode	Byte	80C51 Cycle	W7100A Cycle	
					ISP / wizmemcpy	FLASH / user code
MOV	A, Rb	E8h - EFh	1	12	1	4