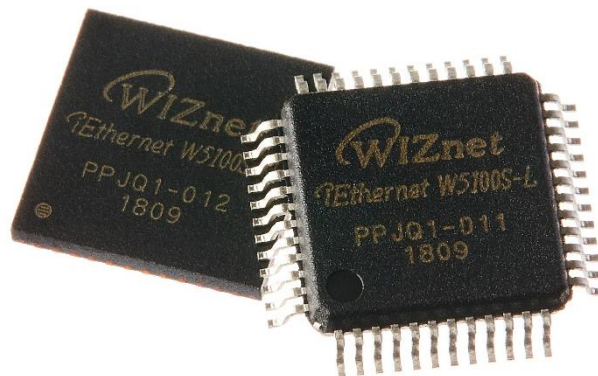


# W5100S

(W5100S-L & W5100S-Q)

Version 1.2.8



## W5100S

W5100S 은 WIZnet 의 Hardwired TCP/IP 기술을 이용한 Embedded Internet Controller Chip 이다. W5100S 는 Internet protocol (TCP/IP) 처리를 위한 Full Hardwired Logic, Ethernet MAC(Media Access Control), 그리고 10Base-T/100Base-TX Ethernet PHY 를 모두 내장한 Internet connectivity One-chip Solution 이다.

W5100S 는 Host(사용자 MCU)의 부담을 최소화할 수 있도록 TCP, UDP, ICMP, IPv4, IGMP, ARP, PPPoE 등의 다양한 통신 프로토콜을 Hardwired Logic 으로 처리할 수 있고, Low-end 급 Host 의 메모리를 극소화할 수 있도록 데이터 송신/수신에 필요한 8KB 의 전용 버퍼 메모리를 각각 내장하고 있다. 또한 사용자는 W5100S 의 독립적인 Hardwired SOCKET 4 개를 동시에 사용하여 다양한 Internet 응용제품을 개발할 수 있다.

W5100S 는 Host 와의 통신을 위해 Serial 방식의 고속 SPI Interface 와 Parallel 방식의 System BUS Interface 를 지원한다. 또한, 시스템의 소비전력을 낮추기 위해 저전력/저발열 설계, WOL (Wake On LAN)와 Ethernet PHY Power Down mode 등을 제공한다.

W5100S 는 제품명 그대로 기존 W5100 제품을 개선한 저가형 제품이다. W5100S 는 W5100 에서 사용되는 Firmware 를 수정하지 않고 곧바로 사용할 수 있는 장점을 가지고 있으며, W5100 보다 작은 48 Pin LQFP & QFN Lead-Free Package 를 채택하여 제품을 소형화할 수 있다.

## Features

- Support Hardwired Internet protocols  
: TCP, UDP, WOL over UDP, ICMP, IGMPv1/v2, IPv4, ARP, PPPoE
- Support 4 independent SOCKETS simultaneously
- Support SOCKET-less Command  
: ARP-Request, PING-Request
- Support Ethernet Power down mode & Main Clock Switching for power save
- Support Wake on LAN over UDP
- Support Serial & Parallel Host Interface  
: High Speed SPI(MODE 0/3), System Bus with 2 Address signal & 8bit Data
- Internal 16Kbytes Memory for TX/ RX Buffers
- 10BaseT/100BaseTX Ethernet PHY Integrated
- Support Auto Negotiation (Full and half duplex, 10 and 100-based)
- Support Auto-MDIX only on Auto-Negotiation Mode
- Not support IP Fragmentation
- 3.3V operation with 5V I/O signal tolerance
- Network Indicator LEDs (Full/Half Duplex, Link, 10/100 Speed, Active)
- 48 Pin LQFP & QFN Lead-Free Package (7x7mm, 0.5mm pitch)

## Target Applications

W5100S 은 다음과 같은 Embedded application 에 적합하다.

- User product based on W5100 : No modify firmware
- Home Network Devices: Set-Top Boxes, PVRs, Digital Media Adapters
- Serial-to-Ethernet: Access Controls, LED displays, Wireless AP relays, etc.
- Parallel-to-Ethernet: POS / Mini Printers, Copiers
- USB-to-Ethernet: Storage Devices, Network Printers
- GPIO-to-Ethernet: Home Network Sensors
- Security Systems: DVRs, Network Cameras, Kiosks
- Factory, Building, Home Automations
- Medical Monitoring Equipment
- Embedded Servers
- Internet of Thing (IoT) Devices
- IoT Cloud Devices

## Block Diagram

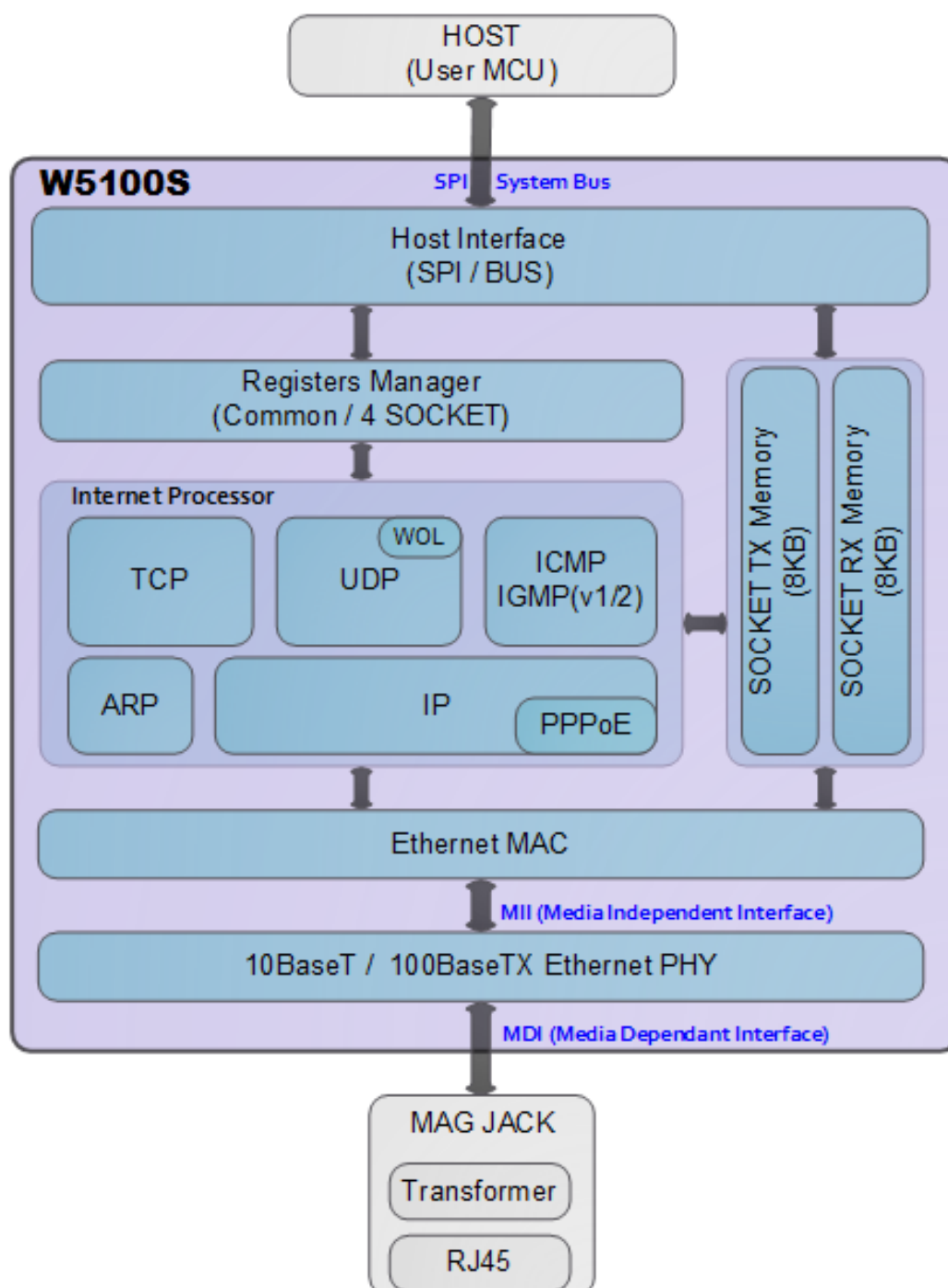


Figure 1 Block Diagram

# Contents

<b>1</b>	<b>PIN Description .....</b>	<b>11</b>
1.1	PIN Description .....	12
<b>2</b>	<b>Memory Map .....</b>	<b>16</b>
2.1	W5100S Registers.....	18
2.1.1	Common registers .....	18
2.1.2	SOCKET Registers.....	20
<b>3</b>	<b>Register Descriptions.....</b>	<b>22</b>
3.1	Common Registers .....	23
3.1.1	MR (Mode Register) .....	23
3.1.2	GWR (Gateway IP Address Register) .....	23
3.1.3	SUBR (Subnet Mask Register) .....	23
3.1.4	SHAR (Source Hardware Address Register) .....	24
3.1.5	SIPR (Source IP Address Register) .....	24
3.1.6	INTPTMR (Interrupt Pending Time Register).....	24
3.1.7	IR (Interrupt Register) .....	24
3.1.8	IMR (Interrupt Mask Register) .....	25
3.1.9	RTR (Retransmission Time Register).....	26
3.1.10	RCR (Retransmission Count Register).....	26
3.1.11	RMSR (RX Memory Size Register) .....	27
3.1.12	TMSR (TX Memory Size Register) .....	27
3.1.13	IR2 (Interrupt Register 2) .....	28
3.1.14	IMR2 (Interrupt Mask Register 2) .....	28
3.1.15	PTIMER (PPP Link Control Protocol Request Timer Register) .....	28
3.1.16	PMAGIC (PPP Link Control Protocol Magic number Register).....	29
3.1.17	UIPR (Unreachable IP Address Register) .....	29
3.1.18	UPORTR (Unreachable Port Register) .....	29
3.1.19	MR2 (Mode Register 2).....	29
3.1.20	PHAR (Destination Hardware Address Register on PPPoE).....	31
3.1.21	PSIDR (Session ID Register on PPPoE).....	31
3.1.22	PMRUR (PPPoE Maximum Receive Unit Register) .....	31
3.1.23	PHYSR0 (PHY Status Register 0) .....	31
3.1.24	PHYSR1 (PHY Status Register 1) .....	32
3.1.25	PHYRAR (PHY Register Address Register) .....	33
3.1.26	PHYDIR (PHY Data Input Register) .....	33
3.1.27	PHYDOR (PHY Data Output Register) .....	33
3.1.28	PHYACR (PHY Access Control Register) .....	34
3.1.29	PHYDIVR (PHY Division Register) .....	34

3.1.30	PHYCRO (PHY Control Register 0) .....	34
3.1.31	PHYCR1 (PHY Control Register 1) .....	34
3.1.32	SLCR (SOCKET-less Command Register) .....	35
3.1.33	SLRTR (SOCKET-less Retransmission Time Register) .....	36
3.1.34	SLRCR (SOCKET-less Retransmission Count Register) .....	36
3.1.35	SLPIPR (SOCKET-less Peer IP Address Register) .....	36
3.1.36	SLPHAR (SOCKET-less Peer Hardware Address Register).....	37
3.1.37	PINGSEQR (PING Sequence-number Register) .....	37
3.1.38	PINGIDR (PING ID Register) .....	37
3.1.39	SLIMR (SOCKET-less Interrupt Mask Register) .....	37
3.1.40	SLIR (SOCKET-less Interrupt Register) .....	38
3.1.41	CLKLCKR (Clock Lock Register) .....	38
3.1.42	NETLCKR (Network Lock Register) .....	39
3.1.43	PHYLCKR (PHY Lock Register) .....	39
3.1.44	VERR (Version Register) .....	39
3.1.45	TCNTR (Ticker Counter Register) .....	39
3.1.46	TCNTCLR (Ticker Counter Clear Register) .....	39
3.2	SOCKET Register.....	40
3.2.1	Sn_MR (SOCKET n Mode Register) .....	40
3.2.2	Sn_CR (SOCKET n Command Register).....	41
3.2.3	Sn_IR (SOCKET n Interrupt Register) .....	43
3.2.4	Sn_SR (SOCKET n Status Register) .....	44
3.2.5	Sn_PORTR (SOCKET n Source Port Register).....	45
3.2.6	Sn_DHAR (SOCKET n Destination Hardware Address Register) .....	45
3.2.7	Sn_DIPR (SOCKET n Destination IP Address Register) .....	46
3.2.8	Sn_DPORTR (SOCKET n Destination Port Register) .....	46
3.2.9	Sn_MSS (SOCKET n Maximum Segment Size Register) .....	46
3.2.10	Sn_PROTOR (SOCKET n IP Protocol Register).....	47
3.2.11	Sn_TOS (SOCKET n IP Type Of Service Register) .....	47
3.2.12	Sn_TTL (SOCKET n IP Time To Live Register) .....	47
3.2.13	Sn_RXBUF_SIZE (SOCKET n RX Buffer Size Register).....	47
3.2.14	Sn_TXBUF_SIZE (SOCKET n TX Buffer Size Register).....	48
3.2.15	Sn_TX_FSR (SOCKET n TX Free Size Register) .....	48
3.2.16	Sn_TX_RD (SOCKET n TX Read Pointer Register).....	49
3.2.17	Sn_TX_WR (SOCKET n TX Write Pointer Register) .....	49
3.2.18	Sn_RX_RSR (SOCKET n RX Received Size Register) .....	49
3.2.19	Sn_RX_RD (SOCKET n RX Read Pointer Register).....	49
3.2.20	Sn_RX_WR (SOCKET n RX Write Pointer Register) .....	50
3.2.21	Sn_IMR (SOCKET n Interrupt Mask Register) .....	50

3.2.22	Sn_FRAGR (SOCKET n Fragment Offset in IP Header Register) .....	50
3.2.23	Sn_MR2 (SOCKET n Mode register 2).....	51
3.2.24	Sn_KPALVTR (SOCKET n Keep Alive Timer Register) .....	52
3.2.25	Sn_RTR (SOCKET n Retransmission Time Register) .....	52
3.2.26	Sn_RCR (SOCKET n Retransmission Count Register) .....	52
<b>4</b>	<b>Functional Description .....</b>	<b>53</b>
4.1	W5100S RESET .....	53
4.2	Initialization .....	53
4.2.1	Basic Setting .....	53
4.2.2	Network Information Setting .....	53
4.2.3	SOCKET TX/RX Buffer Setting .....	54
4.3	TCP.....	55
4.3.1	TCP Server .....	56
4.3.2	TCP Client .....	64
4.3.3	Other Functions .....	66
4.4	UDP.....	67
4.4.1	UDP Unicast .....	67
4.4.2	UDP Broadcast .....	70
4.4.3	UDP Multicast .....	71
4.4.4	Other Functions .....	73
4.5	IPRAW .....	74
4.6	MACRAW .....	77
4.7	SOCKET-less Command (SLCR).....	80
4.7.1	ARP Request (SLCR[ARP] = '1') .....	80
4.7.2	PING Command (SLCR[PING] = '1') .....	82
4.8	Retransmission.....	85
4.8.1	ARP & PING Retransmission .....	85
4.8.2	TCP Retransmission .....	85
4.9	Others Functions .....	87
4.9.1	System Clock(SYS_CLK) Switching .....	87
4.9.2	Ethernet PHY Operation Mode Configuration .....	87
4.9.3	Ethernet PHY Parallel Detection .....	88
4.9.4	Ethernet PHY Auto MDIX .....	88
4.9.5	Ethernet PHY Power Down Mode.....	88
4.9.6	Ethernet PHY's Registers Control .....	89
<b>5</b>	<b>HOST Interface Mode .....</b>	<b>91</b>
5.1	SPI Mode .....	91
5.1.1	SPI Frame.....	91
5.1.2	SPI Write.....	93

---

5.1.3	SPI Read .....	93
5.2	Parallel Bus Mode .....	94
5.2.1	Parallel Bus Data Write.....	94
5.2.2	Parallel Bus Data Read .....	95
<b>6</b>	<b>Clock &amp; Transformer Requirements .....</b>	<b>96</b>
6.1	Quartz Crystal requirements. ....	96
6.2	Oscillator requirements. ....	97
6.3	Transformer Characteristics .....	97
<b>7</b>	<b>Electrical Specification .....</b>	<b>98</b>
7.1	Absolute Maximum ratings .....	98
7.2	Absolute Maximum ratings (Electrical Sensitivity).....	98
7.3	DC Characteristics .....	99
7.4	AC Characteristics.....	100
7.4.1	Reset Timing .....	100
7.4.2	BUS ACCESS TIMING .....	101
7.4.3	SPI ACCESS TIMING .....	102
7.4.4	Transformer Characteristics .....	103
7.4.5	MDIX.....	103
7.5	POWER DISSIPATION .....	104
<b>8</b>	<b>Package Information.....</b>	<b>105</b>
8.1	LQFP48.....	105
8.2	QFN48.....	106
<b>9</b>	<b>Document Revision History .....</b>	<b>108</b>



## List of Figures

Figure 1 Block Diagram .....	4
Figure 2 W5100S Pin Layout.....	11
Figure 3 Memory Map .....	16
Figure 4 State Diagram .....	45
Figure 5 TCP SERVER and TCP CLIENT .....	55
Figure 6 TCP Server Operation Flow .....	56
Figure 7 TCP Client Operation Flow .....	64
Figure 8 UDP Operation Flow .....	67
Figure 9 Received UDP DATA in SOCKET n RX Buffer Block .....	68
Figure 10 IPRAW Operation Flow .....	74
Figure 11 Received Data in IPRAW Mode SOCKET RX Buffer Block .....	75
Figure 12 MACRAW Operation Flow .....	77
Figure 13 Received DATA Format in MACRAW .....	78
Figure 14 SOCKET-less Command Operation Flow .....	80
Figure 15 SCSn controlled by Host .....	91
Figure 16 SPI Mode 0 & Mode 3 .....	91
Figure 17 SPI Frame.....	92
Figure 18 W5100 Mode Write SPI Frame .....	93
Figure 19 W5100 Mode Read SPI Frame .....	93
Figure 20 Direct & Indirect Mode Control by Host .....	94
Figure 21 Parallel Bus N-Bytes Data Write .....	94
Figure 22 Indirect Mode Continuous Read Access .....	95
Figure 23 Quartz Crystal Model.....	96
Figure 24 Transformer Type.....	97
Figure 25 Reset Timing .....	100
Figure 26 Bus Read Timing .....	101
Figure 27 BUS Write Timing .....	101
Figure 28 SPI Read Timing .....	102
Figure 29 SPI Write Timing .....	103
Figure 30 Transformer Type.....	103

---

## List of Tables

Table 1 Pin Type Notation .....	11
Table 2 PIN Description.....	12
Table 3 Common Registers.....	18
Table 4 Socket Registers .....	20
Table 5 Internet Protocol Supported In IPRAW Mode .....	74
Table 6 W5100 Mode SPI Command .....	92
Table 7 Indirect Mode Address Value .....	94
Table 8 Quartz Crystal.....	96
Table 9 Crystal Recommendation Characteristics .....	96
Table 10 Oscillator Characteristics.....	97
Table 11 Transformer Characteristics .....	97
Table 12 Absolute Maximum ratings .....	98
Table 13 Electro Static Discharge (ESD) .....	98
Table 14 Latch up Test .....	98
Table 15 DC Characteristics .....	99
Table 16 Reset Table .....	100
Table 17 BUS Read Timing .....	101
Table 18 BUS Write timing .....	102
Table 19 SPI Read Timing .....	102
Table 20 SPI Write Timing .....	103
Table 21 Transformer Characteristics .....	103
Table 22 Power Dissipation.....	104
Table 23 LQFP48 VARIATIONS (ALL DEMINSIONS SHOWN IN MM) .....	105
Table 24 QFN48 VARIATIONS (ALL DEMINSIONS SHOWN IN MM) .....	106

## 1 PIN Description

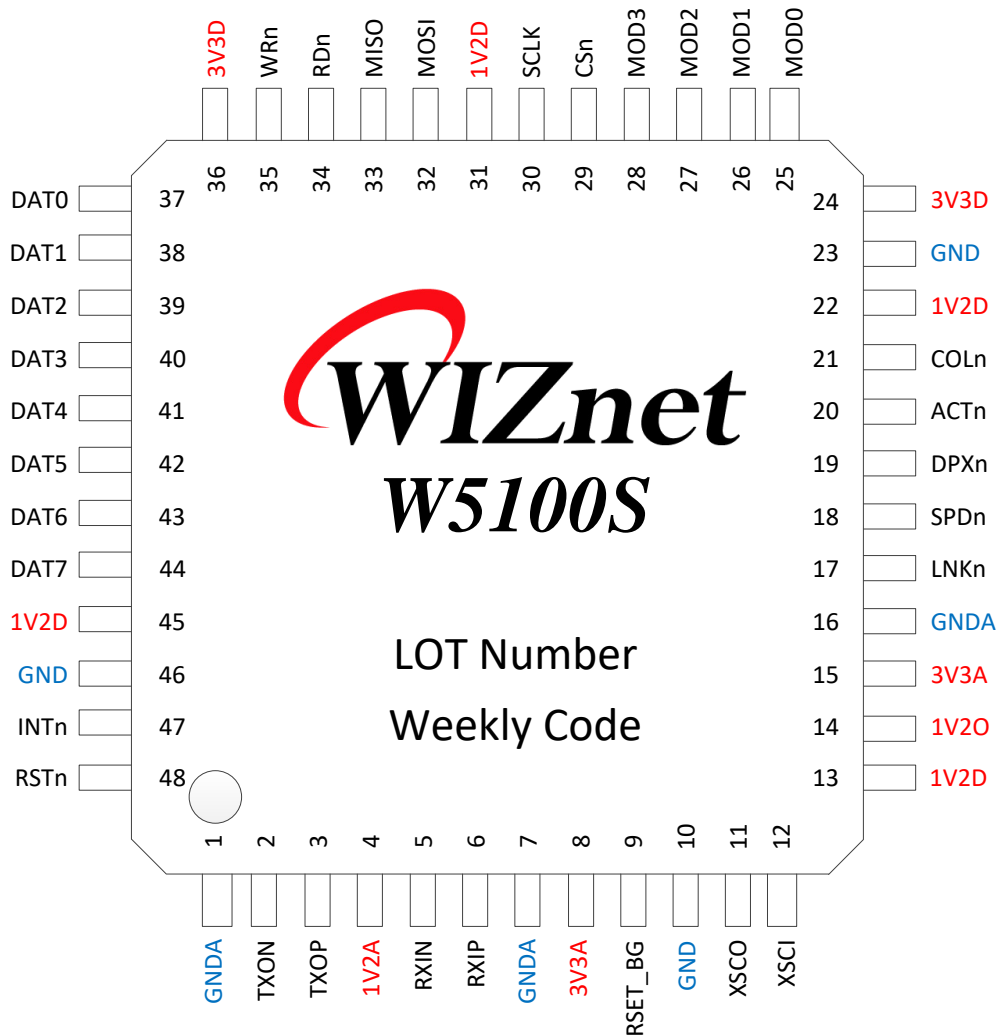


Figure 2 W5100S Pin Layout

Table 1 Pin Type Notation

Type	Description
I	Input
O	Output
M	Alternate (Multi-function) Signal
U	Internal pulled-up 75K $\Omega$ resistor
D	Internal pulled-down 75K $\Omega$ resistor
A	Analog
P	Power & Ground

## 1.1 PIN Description

Table 2 PIN Description

PIN #	Symbol	Type	Description
1	GNDA	AP	<b>Analog Ground</b>
2	TXON	AO	<b>Differential Transmitted Signal Pair</b> 차동 데이터는 MDI 모드에서 TXOP/TXON 신호 쌍을 통해 미디어로 전송된다.
3	TXOP	AO	
4	1V2A	AP	<b>Analog 1.2V Power</b> 1V2O 전압 소스로부터 공급받는다.
5	RXIN	AI	<b>Differential Received Signal Pair</b> 차동 데이터를 MDI 모드에서 RXIP/RXON 신호 쌍을 통해 미디어로부터 수신한다.
6	RXIP	AI	
7	GNDA	AP	<b>Analog Ground</b>
8	3V3A	AP	<b>Analog 3.3V Power</b>
9	RSET_BG	AO	<b>Off-chip Bias Resistor</b> 외부 12.3K $\Omega$ , 오차 1% 저항을 통해 Analog Ground 로 반드시 연결해야 한다.
10	GND	AP	<b>Digital Ground</b>
11	XSCO	AO	<b>25MHz Clock</b> 외부로부터 25MHz clock 을 입력한다. W5100S 는 외부 25MHz 의 Clock 을 25MHz (Low Frequency mode) 이나 4 배의 100MHz (Normal mode) 으로 변환하고 이를 내부 Operation Clock(SYS_CLK) 으로 사용한다. 25MHz Crystal Oscillator(XTAL)나 Oscillator(OSC)로 연결한다. OSC 를 사용할 경우 25MHz@1.2V 를 사용하며 XSCI 만 연결하고 XSCO 는 반드시 Floating 시킨다. 자세한 내용은 <a href="#">Clock Selection Guide</a> 를 참조한다.
12	XSCI	AI	
13	1V2D	P	<b>Digital 1.2V Power</b> 1V2O 전압 소스로부터 공급받는다.
14	1V2O	PO	<b>Internal Regulator 1.2V Power Output</b> W5100S 를 위한 내부 Regulator 의 1.2V Power Output 으로 Max 150mA 를 지원하며, 반드시 외부 안정화 Capacitor 3.3 $\mu$ F 를 통해 W5100S 의 1V2D 와 1V2A 로 공급한다. 1V2O 는 Ferrite Bead 를 사용하여, 1V2D 와 1V2A 분리 하여 공급한다.

			이 Power 는 W5100S 을 위한 것으로 다른 Device 의 Power 로 사용될 수 없다.
15	3V3A	AP	<b>Analog 3.3V Power</b>
16	GNDA	AP	<b>Analog Ground</b>
17	LNKn	OU	<b>Link Status LED</b> SPI, Parallel Bus Mode 일 때만 유효  Low : Link up High : Link down
18	SPDn	OU	<b>Link Speed LED</b> SPI, Parallel Bus Mode 일 때만 유효  Low : 100Mbps High : 10Mbps
19	DPXn	OU	<b>Link Duplex LED</b> SPI, Parallel Bus Mode 일 때만 유효  Low : Full-Duplex High : Half-Duplex
20	ACTn	OU	<b>Link Activity LED</b> SPI, Parallel Bus Mode 일 때만 유효 No Flash : Link up state without TX/RX  Flash : Link up state with TX/RX data High : Link-down state
21	COLn	OU	<b>Link Collision Detect LED</b> SPI, Parallel Bus Mode 일 때만 유효 W5100S Data 송신 시 충돌을 감지한 경우 알려준다.  Low : Collision Detected High : No Collision
22	1V2D	P	<b>Digital 1.2V Power</b> 1V20 로부터 공급 받는다.
23	GND	P	<b>Digital Ground</b>
24	3V3D	P	<b>Digital 3.3V power</b>
25	MOD[0]	ID	<b>W5100S Mode Selection</b>
26	MOD[1]	ID	W5100S 를 MOD[3:0]에 따라 아래와 같이 선택한다. “0000” : SPI Mode “010X” : Parallel Bus Mode
27	MOD[2]	ID	
28	MOD[3]	ID	

29	CSn	IU	<b>W5100S Chip Select</b> Low : Select High : No Select
30	SCLK	ID	<b>SPI Clock</b> SPI Mode 일 때, SPI Clock 을 입력한다. Parallel Bus Mode 인 경우, GND 와 연결하거나 Floating 한다.
31	1V2D	P	<b>Digital 1.2V Power</b> 1V2O 로부터 공급받는다.
32	MOSI /ADDR0	IDM	<b>SPI Master Output Slave Input / Address 0</b> MOSI : SPI mode 일 때, SPI Data 를 HOST 로부터 수신 ADDR0 : Parallel Bus Mode 일 때, Address 0 을 입력
33	MISO /ADDR1	IOPM	<b>SPI Master Input Slave Output / Address 1</b> MISO : SPI mode 일 때, SPI Data 를 HOST 로 송신 ADDR1 : Parallel Bus Mode 일 때, Address 1 을 입력
34	RDn	IU	<b>Read Strobe</b> Parallel Bus Mode 일 경우, Read Operation 을 알린다. SPI Mode 일 경우 3V3D 와 연결하거나 Floating 한다.
35	WRn	IU	<b>Write Strobe</b> Parallel Mode 일 경우, Write Operation 을 알린다.
36	3V3D	P	<b>Digital 3.3V Power</b>
37	DAT0	IOU	<b>8 Bits Data Bus</b> Parallel Bus Mode 일 경우, HOST 의 Data 를 입력하거나, W5100S 으로부터 Data 를 입력 받는다. SPI Mode 인 경우, Floating 한다.
38	DAT1	IOU	
39	DAT2	IOU	
40	DAT3	IOU	
41	DAT4	IOU	
42	DAT5	IOU	
43	DAT6	IOU	
44	DAT7	IOU	
45	1V2D	P	<b>Digital 1.2V Power</b>
46	GND	P	<b>Digital Ground</b>
47	INTn	OP	<b>Interrupt</b> W5100S 의 Ethernet 통신 처리시 Event 가 발생할 경우, HOST 에게 알린다. Low : Interrupt Occurred High : No Interrupt

			(참조 <i>MR2 (Mode Register 2)의 IEN (Interrupt pin Enable), INTPTMR (Interrupt Pending Time Register), IMR (Interrupt Mask Register), IMR2 (Interrupt Mask Register 2), SLIMR (SOCKET-less Interrupt Mask Register)</i> )
48	RSTn	IP	<p><b>Reset</b></p> <p>W5100S 을 초기화 한다. RSTn 신호는 반드시 560ns 이상 Low 를 유지해야 한다. W5100S 는 RSTn 신호가 인가 된 후 60.3ms 이후에 완전히 초기화가 된다. (참조 <a href="#">7.4.1 Reset Timing</a>)</p> <p>Low : W5100S 를 초기화 시킨다. High : W5100S 를 정상 동작 시킨다.</p>

## 2 Memory Map

W5100S 는 W5100 호환성을 위해 동일한 Memory Map 으로 구성되고, 기능 향상을 위해 Figure 3 과 같이 새로운 Common Register 들이 추가된다.

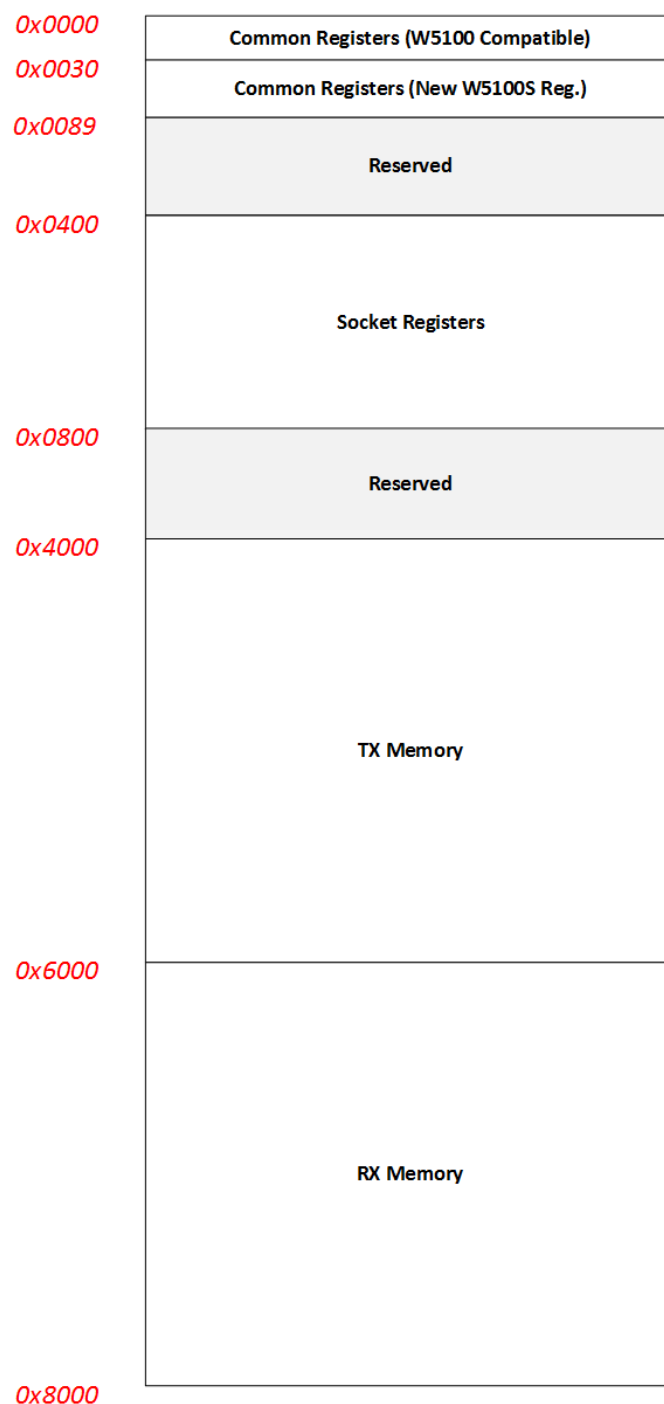


Figure 3 Memory Map



Figure 3 는 Common & SOCKET Register Block 과 TX/RX Memory Block 들의 Address Offset 을 도식화 한다. W5100S Reset 시, W5100S 는 TX/RX Memory Block 의 각 SOCKET TX/RX Buffer Block 을 2KB/2KB 로 설정한다.

W5100S Reset 이후, SOCKET n TX Buffer Block Size 는 TMSR(TX Memory Size Register)를, SOCKET RX Buffer Size 는 RMSR(RX Memory Size Register)를 통해 재설정 가능하다. 또한 W5100S 의 추가된 SOCKET n TX/RX Buffer Size Register(Sn\_TXBUF\_SIZE / Sn\_RXBUF\_SIZE)를 통해 SOCKET n TX/RX Buffer Block Size 를 재설정 가능하다. 설정된 SOCKET n TX/RX Buffer Block 의 총합은 각각 8KB 를 초과할 수 없다.

## 2.1 W5100S Registers

### 2.1.1 Common registers

Table 3 Common Registers

Address	Register	Address	Register
0x0000	Mode (MR)	0x001C	Reserved
0x0001	Gateway Address (GAR0) (GAR1) (GAR2) (GAR3)	~ 0x001F	
0x0002		0x0020	Interrupt2 (IR2)
0x0003		0x0021	Interrupt2 Mask (IMR2)
0x0004		0x0022	Reserved
0x0005	Subnet Mask Address (SUBR0) (SUBR1) (SUBR2) (SUBR3)	~ 0x0027	
0x0006		0x0028	PPP LCP Request Timer (PTIMER)
0x0007		0x0029	PPP LCP Magic Number (PMAGIC)
0x0008		0x002A	Unreachable IP Address (UIPR0) (UIPR1) (UIPR2) (UIPR3)
0x0009	Source Hardware Address (SHAR0) (SHAR1) (SHAR2) (SHAR3) (SHAR4) (SHAR5)	0x002B	
0x000A		0x002C	
0x000B		0x002D	
0x000C		0x002E	Unreachable Port (UPORTR0) (UPORTR1)
0x000D		0x002F	
0x000E		0x0030	Mode2 (MR2)
0x000F	Source IP Address (SIPR0) (SIPR1) (SIPR2) (SIPR3)	0x0031	Reserved
0x0010		0x0032	Destination Hardware Address on PPPoE (PHAR0) (PHAR1) (PHAR2) (PHAR3) (PHAR4) (PHAR5)
0x0011		0x0033	
0x0012		0x0034	
0x0013		0x0035	
0x0014	Interrupt Pending Time (INTPTMR0) (INTPTMR1)	0x0036	
0x0015		0x0037	
0x0016	Interrupt Mask (IMR)	0x0038	Session ID on PPPoE (PSIDR0) (PSIDR1)
0x0017	Retransmission Time (RTR0) (RTR0)	0x0039	
0x0018		0x003A	Maximum Receive Unit on PPPoE (PMRUR0) (PMRUR1)
0x0019	Retransmission Count (RCR)	0x003B	
0x001A	RX Memory Size (RMSR)	0x003C	PHY Status (PHYSR0)
0x001B	TX Memory Size (TMSR)		

Address	Register
0x003E	PHY Address Value (PHYAR)
0x003F	PHY Register Address (PHYRAR)
0x0040	PHY Data Input (PHYDIR0)
0x0041	(PHYDIR1)
0x0042	PHY Data Output (PHYDOR0)
0x0043	(PHYDOR1)
0x0044	PHY Access (PHYACR)
0x0045	PHY Division (PHYDIVR)
0x0046	PHY Control (PHYCR0)
0x0047	(PHYCR1)
0x0048	Reserved
~	
0x004B	
0x004C	SOCKET-less Command (SLCR)
0x004D	SOCKET-less Retransmission Time (SLRTR0)
0x004E	(SLRTR1)
0x004F	SOCKET-less Retransmission Count (SLRCR)
0x0050	SOCKET-less Peer IP Address (SLPIPR0)
0x0051	(SLPIPR1)
0x0052	(SLPIPR2)
0x0053	(SLPIPR3)
0x0054	SOCKET-less Peer Hardware Address (SLPHAR0)
0x0055	(SLPHAR1)
0x0056	(SLPHAR2)
0x0057	(SLPHAR3)
0x0058	(SLPHAR4)
0x0059	(SLPHAR5)
0x005A	PING Sequence Number (PINGSEQR0)
0x005B	(PINGSEQR1)
0x005C	PING ID (PINGIDR0)
0x005D	(PINGIDR1)

Address	Register
0x005E	SOCKET-less Interrupt Mask (SLIMR)
0x005F	SOCKET-less Interrupt (SLIR)
0x0060	Reserved
~	
0x006A	
0x0070	Clock Lock (CLKLCKR)
0x0071	Network Lock (NETLCKR)
0x0072	PHY Lock (PHYLCKR)
0x0073	Reserved
~	
0x007F	
0x0080	Chip Version (VERR)
0x0081	Reserved
0x0082	100us Tick Counter (TCNTR0)
0x0083	(TCNTR1)
0x0084	Reserved
~	
0x0087	
0x0088	TCNTCLR

## 2.1.2 SOCKET Registers

Table 4 Socket Registers

Symbol	Description	Address				
		Sn_	S0_	S1_	S2_	S3_
Sn_MR	SOCKET n Mode	0x0400+(0x0100 x n)	0x0400	0x0500	0x0600	0x0700
Sn_CR	SOCKET n Command	0x0401+(0x0100 x n)	0x0401	0x0501	0x0601	0x0701
Sn_IR	SOCKET n Interrupt	0x0402+(0x0100 x n)	0x0402	0x0502	0x0602	0x0702
Sn_SR	SOCKET n Status	0x0403+(0x0100 x n)	0x0403	0x0503	0x0603	0x0703
Sn_PORTR0	SOCKET n Source Port	0x0404+(0x0100 x n)	0x0404	0x0504	0x0604	0x0704
Sn_PORTR1		0x0405+(0x0100 x n)	0x0405	0x0505	0x0605	0x0705
Sn_DHAR0	SOCKET n Destination Hardware Address	0x0406+(0x0100 x n)	0x0406	0x0506	0x0606	0x0706
Sn_DHAR1		0x0407+(0x0100 x n)	0x0407	0x0507	0x0607	0x0707
Sn_DHAR2		0x0408+(0x0100 x n)	0x0408	0x0508	0x0608	0x0708
Sn_DHAR3		0x0409+(0x0100 x n)	0x0409	0x0509	0x0609	0x0709
Sn_DHAR4		0x040A+(0x0100 x n)	0x040A	0x050A	0x060A	0x070A
Sn_DHAR5		0x040B+(0x0100 x n)	0x040B	0x050B	0x060B	0x070B
Sn_DIPR0	SOCKET n Destination IP Address	0x040C+(0x0100 x n)	0x040C	0x050C	0x060C	0x070C
Sn_DIPR1		0x040D+(0x0100 x n)	0x040D	0x050D	0x060D	0x070D
Sn_DIPR2		0x040E+(0x0100 x n)	0x040E	0x050E	0x060E	0x070E
Sn_DIPR3		0x040F+(0x0100 x n)	0x040F	0x050F	0x060F	0x070F
Sn_DPORTR0	SOCKET n Destination Port	0x0410+(0x0100 x n)	0x0410	0x0510	0x0610	0x0710
Sn_DPORTR0		0x0411+(0x0100 x n)	0x0411	0x0511	0x0611	0x0711
Sn_MSS0	SOCKET n Maximum Segment Size	0x0412+(0x0100 x n)	0x0412	0x0512	0x0612	0x0712
Sn_MSS1		0x0413+(0x0100 x n)	0x0413	0x0513	0x0613	0x0713
Sn_PROTOR	SOCKET n IP Protocol	0x0414+(0x0100 x n)	0x0414	0x0514	0x0614	0x0714
Sn_TOS	SOCKET n IP Type Of Service	0x0415+(0x0100 x n)	0x0415	0x0515	0x0615	0x0715
Sn_TTL	SOCKET n IP Time To Live	0x0416+(0x0100 x n)	0x0416	0x0516	0x0616	0x0716
Reserved	Reserved	0x0417+(0x0100 x n)	0x0417	0x0517	0x0617	0x0717
Reserved	Reserved	0x041D+(0x0100 x n)	0x041D	0x051D	0x061D	0x071D
Sn_RXBUF_SIZE	SOCKET n RX Buffer Size	0x041E+(0x0100 x n)	0x041E	0x051E	0x061E	0x071E
Sn_TXBUF_SIZE	SOCKET n TX Buffer Size	0x041F+(0x0100 x n)	0x041F	0x051F	0x061F	0x071F
Sn_TX_FSR0	SOCKET n	0x0420+(0x0100 x n)	0x0420	0x0520	0x0620	0x0720

Sn_TX_FSR1	TX Free Size	$0x0421 + (0x0100 \times n)$	0x0421	0x0521	0x0621	0x0721
Sn_TX_RD0	SOCKET n	$0x0422 + (0x0100 \times n)$	0x0422	0x0522	0x0622	0x0722
Sn_TX_RD1	TX Read Pointer	$0x0423 + (0x0100 \times n)$	0x0423	0x0523	0x0623	0x0723
Sn_TX_WR0	SOCKET n	$0x0424 + (0x0100 \times n)$	0x0424	0x0524	0x0624	0x0724
Sn_TX_WR1	TX Write Pointer	$0x0425 + (0x0100 \times n)$	0x0425	0x0525	0x0625	0x0725
Sn_RX_RSR0	SOCKET n	$0x0426 + (0x0100 \times n)$	0x0426	0x0526	0x0626	0x0726
Sn_RX_RSR1	RX Received Size	$0x0427 + (0x0100 \times n)$	0x0427	0x0527	0x0627	0x0727
Sn_RX_RD0	SOCKET n	$0x0428 + (0x0100 \times n)$	0x0428	0x0528	0x0628	0x0728
Sn_RX_RD1	RX Read Pointer	$0x0429 + (0x0100 \times n)$	0x0429	0x0529	0x0629	0x0729
Sn_RX_WR0	SOCKET n	$0x042A + (0x0100 \times n)$	0x042A	0x052A	0x062A	0x072A
Sn_RX_WR1	RX Write Pointer	$0x042B + (0x0100 \times n)$	0x042B	0x052B	0x062B	0x072B
Sn_IMR	SOCKET n Interrupt Mask	$0x042C + (0x0100 \times n)$	0x042C	0x052C	0x062C	0x072C
Sn_FRAGRO	SOCKET n	$0x042D + (0x0100 \times n)$	0x042D	0x052D	0x062D	0x072D
Sn_FRAGR1	Fragment Offset in IP Header	$0x042E + (0x0100 \times n)$	0x042E	0x052E	0x062E	0x072E
Sn_MR2	SOCKET n Mode 2	$0x042F + (0x0100 \times n)$	0x042F	0x052F	0x062F	0x072F
Sn_KPALVTR	SOCKET n Keep-alive Timer	$0x0430 + (0x0100 \times n)$	0x0430	0x0530	0x0630	0x0730
Sn_RTR0	SOCKET n	$0x0432 + (0x0100 \times n)$	0x0432	0x0532	0x0632	0x0732
Sn_RTR1	Retransmission Time	$0x0433 + (0x0100 \times n)$	0x0433	0x0533	0x0633	0x0733
Sn_RCR	SOCKET n Retransmission Count	$0x0434 + (0x0100 \times n)$	0x0434	0x0534	0x0634	0x0734

### 3 Register Descriptions

#### Register Notation

* Register Symbol (Register full Name)							
- [Register Type][Address Offset][Reset Value]							
Register Description....							
7	6	5	4	3	2	1	0
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Bit Type	Bit Type	Bit Type	Bit Type	Bit Type	Bit Type	Bit Type	Bit Type
Sn_IR[3:0]은 Register Symbol[Upper Bit:Lower Bit]을 나타낸다.							
Sn_IR[3:0] = “0001”은 Sn_IR[3]='0', Sn_IR[2]='0', Sn_IR[1]='0', Sn_IR[0]='1'을 나타낸다.							

[Register/Bit Type]: Register 의 Type 을 나타낸다.

- [RW] : Read 와 Write 모두 가능한 Register/Bit
- [R=W] : Write 한 값과 Read 한 값이 같은 Register/Bit
- [RO] : Read Only Register/Bit
- [WO] : Write Only Register/Bit
- [W] : Write Only Register/Bit
- [WC] : Write ‘1’을 해야 Clear 되는 Register/Bit
- [W0] : Register/Bit Should be written to only ‘0’
- [W1] : Register/Bit Should be written to only ‘1’
- [AC] : Auto Clear Registr/Bit
- [1] : Always read ‘1’
- [0] : Always read ‘0’
- [-] : Not available

[Address Offset]: 레지스터의 Address Offset 을 나타낸다.

[Reset Value]: Default Value.

Ex1) 4.1.1 MR (Mode Register)

[RW][0x0000][0x03]

MR 의 Full Name 은 Mode Register 을 의미하며, Read 와 Write 가 모두 가능하고, Address Offset 은 0x0000 이며, Reset 신호가 들어왔을 때 설정되는 값은 0x03 이다.

7	6	5	4	3	2	1	0
RST	-	-	PB	PPPoE	-	AI	IND
AC	W0	W0	R=W	R=W	-	1	1

Ex2) MR[RST]

MR 의 RST Bit

Ex3) MR[7:0]

MR 의 7<sup>th</sup> Bit 부터 0<sup>th</sup> Bit

### 3.1 Common Registers

#### 3.1.1 MR (Mode Register)

[RW][0x0000] [0x03]

MR 은 Reset, PING Block, PPPoE Enable 을 설정한다.

7	6	5	4	3	2	1	0
RST	-	-	PB	PPPoE	-	-	-

Bit	Symbol	Description
7	RST	<b>Reset</b> W5100S 를 초기화한다.  1 : Reset, 3 SYS_CLK 이후 Auto Clear 0 : Normal Operation
[6:5]	-	Reserved
4	PB	<b>PING Response Block</b> PING Request 수신에 대한 응답여부를 결정한다.  1 : Diabale PING Response 0 : Enable PING Response
3	PPPoE	<b>PPPoE Enable</b> PPPoE 를 Enable 한다.  1 : Enable PPPoE 0 : Disable PPPoE
[2:0]	-	Reserved

#### 3.1.2 GWR (Gateway IP Address Register)

[R=W] [0x0001-0x0004] [0x00]

GWR 은 NETLCKR(Network Lock Register)이 Unlock 일 때 Gateway 의 IP Address 를 설정한다.

Ex) GWR = "192.168.0.1"

GWR0(0x0001)	GWR1(0x0002)	GWR2(0x0003)	GWR3(0x0004)
192 (0xC0)	168 (0xA8)	0 (0x00)	1 (0x01)

#### 3.1.3 SUBR (Subnet Mask Register)

[R=W] [0x0005-0x0008] [0x00]

SUBR 은 NETLCKR 이 Unlock 일 때 Subnet Mask 범위를 설정한다.

Ex) SUBR = "255.255.255.255"

SUBR0(0x0005)	SUBR0(0x0006)	SUBR0(0x0007)	SUBR0(0x0008)
255 (0xFF)	255 (0xFF)	255 (0xFF)	255 (0xFF)

### 3.1.4 SHAR (Source Hardware Address Register)

**[R=W] [0x0009-0x000E] [0x00]**

SHAR 은 NETLCKR 이 Unlock 일 때 Source MAC Address 를 설정한다.

Ex) SHAR = "11:22:33:AA:BB:CC"

SHAR0(0x0009)	SHAR1(0x000A)	SHAR2(0x000B)
0x11	0x22	0x33
SHAR3(0x000C)	SHAR4(0x000D)	SHAR5(0x000E)
0xAA	0xBB	0xCC

### 3.1.5 SIPR (Source IP Address Register)

**[R=W] [0x000F-0x0012] [0x00]**

SIPR 은 NETLCKR 이 Unlock 일 때 Source IP Address 를 설정한다.

Ex) SIPR = "192.168.0.100"

SIPR0(0x000F)	SIPR1(0x0010)	SIPR2(0x0011)	SIPR3(0x0012)
192 (0xC0)	168 (0xA8)	0 (0x00)	100(0x64)

### 3.1.6 INTPTMR (Interrupt Pending Time Register)

**[R=W][0x0013-0x0014][0x0000]**

INTPTMR 은 내부 Interrupt Pending Timer Count 를 설정한다. Timer Count 는 INTn 이 High De-assert 되는 경우 INTPTMR 로 초기화되며, Interrupt 가 발생한 시점부터 0 이 될 때까지 SYS\_CLK X 4 시간단위로 1 씩 감소하게 된다.

INTn 은 Interrupt 가 발생하고 해당 Interrupt Mask 가 활성화되고 INTPTMR = 0 인 경우, Low Assert 된다.

Ex) INTPTMR = 1000(0x03EB)

INTPTMR0(0x0013)	INTPTMR1(0x0014)
0x03	0xEB

### 3.1.7 IR (Interrupt Register)

**[RW] [0x0015] [0x00]**

IR 은 W5100S 의 Event 나 SOCKET n 의 Event 가 발생했을 때 해당 Event bit 가 1 로 설정된다.



IR 의 Event 가 발생하고 Interrupt Mask Register(IMR)의 1:1 대응되는 Bit 가 ‘1’로 설정되고, 내부 Interrupt Pending Timer Count 가 0 인 경우, INTn 은 Low assert 되며, IR 의 Event 가 Clear 되거나 해당 Mask Bit 가 ‘0’으로 설정된 경우 High De-assert 된다.

7	6	5	4	3	2	1	0
CONFLICT	UNREACH	PPPTERM	-	S3_INT	S2_INT	S1_INT	S0_INT
WC	WC	WC	-	AC	AC	AC	AC

Bit	Symbol	Description
7	CONFLICT	<b>IP Conflict</b> Read 1 : IP 충돌감지 Read 0 : -
6	UNREACH	<b>Destination Port Unreachable</b> 수신한 Unreachable Packer 의 Unreachable IP Address 와 Port Number 는 각각 UIPR (Unreachable IP Register) 과 UPORTR (Unreachable Port Register)에 저장된다.  Read 1 : Unreachable Packet 수신 Read 0 : -
5	PPPTERM	<b>PPPoE Terminated</b> Read 1 : PPPoE 에서 PPPT 또는 LCPT Packet 수신한 경우 Read 0 : -
4	-	Reserved
[3:0]	Sn_INT	<b>SOCKET n Interrupt</b> Read 1 : Sn_IR(SOCKET n Interrupt Register)가 0 이 아닌 경우 Read 0 : Sn_IR 값이 0x00 인 경우 Sn_INT bit 는 Auto-Clear 된다.

### 3.1.8 IMR (Interrupt Mask Register)

**[R=W] [0x0016] [0x00]**

IMR 은 IR 에 1:1 대응되는 Bit 를 Mask 한다.

7	6	5	4	3	2	1	0
CNFT	UNREACH	PPPTERM	-	S3_INT	S2_INT	S1_INT	S0_INT
R=W	R=W	R=W		R=W	R=W	R=W	R=W

Bit	Symbol	Description
7	CNFT	<b>IP Conflict Interrupt Mask</b> 1 : Enable IP Conflict Interrupt 0 : Disable IP Conflict Interrupt

6	UNREACH	<b>Destination Port Unreachable Interrupt Mask</b> 1 : Enable Destination Port Unreachable Interrupt 0 : Disable Destination Port Unreachable Interrupt
5	PPPTERM	<b>PADT/LCPT Interrupt Mask</b> 1 : Enable PADT/LCPT Interrupt 0 : Disable PADT/LCPT Interrupt
4	-	Reserved
3	S3_INT	<b>SOCKET 3 Interrupt Mask</b> 1 : Enable SOCKET 3 Interrupt 0 : Disable SOCKET 3 Interrupt
2	S2_INT	<b>SOCKET 2 Interrupt Mask</b> 1 : Enable SOCKET 2 Interrupt 0 : Disable SOCKET 2 Interrupt
1	S1_INT	<b>SOCKET 1 Interrupt Mask</b> 1 : Enable SOCKET 1 Interrupt 0 : Disable SOCKET 1 Interrupt
0	S0_INT	<b>SOCKET 0 Interrupt Mask</b> 1 : Enable SOCKET 0 Interrupt 0 : Disable SOCKET 0 Interrupt

### 3.1.9 RTR (Retransmission Time Register)

**[R=W] [0x0017-0x0018] [0x07D0]**

RTR 은 Sn\_RTR(SOCKET n Retransmission Time Register)의 초기값을 설정한다.

단위는 100us 이다. W5100 과 달리 W5100S 는 추가된 Sn\_RTR 을 통해 SOCKET 별로 Timeout 을 설정할 수 있다.

RCR(Retransmission Counter Register)과 함께 Packet (ARP, TCP) 재전송에 관여한다.

(참조 [4.8Retransmission](#))

Ex) RTR = 5000 (0x1388)

$$5000 * 100\mu s = 0.5s$$

RTR0(0x0017)	RTR1(0x0018)
0x13	0x88

### 3.1.10RCR (Retransmission Count Register)

**[R=W] [0x0019] [0x07]**

RCR 은 Sn\_RCR(SOCKET n Retransmission Count Register)의 값이 0 인 SOCKET n 의 Retransmission Counter 값을 설정한다. W5100 과 달리 W5100S 는 Sn\_RCR 을 통해 SOCKET 별로 Retransmission Counter 를 설정할 수 있다.

RTR(Retransmission)과 함께 SOCKET 에 의한 Packet (ARP, TCP) 재전송에 관여한다.

(참조 [4.8Retransmission](#))

### 3.1.11RMSR (RX Memory Size Register)

**[R=W] [0x001A] [0x55]**

RMSR 은 각 SOCKET n RX Buffer Block Size 를 설정한다.

SOCKET n RX Buffer Block Size 의 총 합은 8 Kbytes 를 초과할 수 없다. (참조 [Sn\\_RXBUF\\_SIZE](#) ([SOCKET n RX Buffer Size Register](#)))

7	6	5	4	3	2	1	0
SOCKET 3		SOCKET 2		SOCKET 1		SOCKET 0	
S1	S0	S1	S0	S1	S0	S1	S0

각각의 SOCKET 은 S0 와 S1 의 값에 의해서 아래와 같이 Buffer Block 이 설정된다.

Buffer Size	S1	S0
1 KB	0	0
2 KB	0	1
4 KB	1	0
8 KB	1	1

### 3.1.12TMSR (TX Memory Size Register)

**[R=W] [0x001B] [0x55]**

TMSR 은 각 SOCKET n TX Buffer Block Size 를 설정한다.

SOCKET n TX Buffer Block Size 의 총 합은 8 Kbytes 를 초과할 수 없다. (참조 [Sn\\_TXBUF\\_SIZE](#) ([SOCKET n TX Buffer Size Register](#)))

7	6	5	4	3	2	1	0
SOCKET 3		SOCKET 2		SOCKET 1		SOCKET 0	
S1	S0	S1	S0	S1	S0	S1	S0

각각의 SOCKET 은 S0 와 S1 의 값에 의해서 아래와 같이 Buffer Block 이 설정된다.

Buffer Size	S1	S0
1 KB	0	0
2 KB	0	1
4 KB	1	0
8 KB	1	1

### 3.1.13 IR2 (Interrupt Register 2)

**[RW] [0x0020] [0x00]**

IR2 은 WOL Event 가 발생했을 때 IR2 [WOL]이 1 로 설정된다.

IR2 의 WOL Event 가 발생하고 IMR2(Interrupt Mask Register 2)의 해당 1:1 대응되는 Bit 가 '1'로 설정되고, 내부 Interrupt Pending Timer Count 가 0 인 경우, INTn 은 Low assert 되며, IR2 의 WOL Event 가 Clear 되거나 해당 Mask Bit 가 '0'으로 설정된 경우 High De-assert 된다.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	WOL
-	-	-	-	-	-	-	WC

Bit	Symbol	Description
[7:1]	-	Reserved
0	WOL	<b>WOL MAGIC Packet Interrupt</b> 1 : UDP 기반 WOL MAGIC Packet 수신 0 : -

### 3.1.14 IMR2 (Interrupt Mask Register 2)

**[R=W] [0x0021] [0x00]**

IMR2 는 IR2 에 1:1 로 대응되는 Bit 를 Mask 한다.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	WOL
-	-	-	-	-	-	-	R=W

Bit	Symbol	Description
[7:1]	-	Reserved
0	WOL	<b>WOL MAGIC Packet Interrupt Mask</b> 1 : Enable WOL MAGIC Packet Interrupt 0 : Disable WOL MAGIC Packet Interrupt

### 3.1.15 PTIMER (PPP Link Control Protocol Request Timer Register)

**[R=W] [0x0028] [0x28]**

PTIMER 은 LCP Echo Request 주기를 설정한다.

단위는 25ms 이다.

Ex) PTIMER = 200 (0xC8),

200 \* 25ms = 5s

### 3.1.16 PMAGIC (PPP Link Control Protocol Magic number Register)

**[R=W] [0x0029] [0x00]**

PMAGIC 은 LCP Negotiation Packet 의 4 Bytes Magic Number 를 설정한다.

Ex) PMAGIC = 0x01

PMAGIC(0x0029)
0x01
LCP Magic number = 0x01010101

### 3.1.17 UIPR (Unreachable IP Address Register)

**[RO] [0x002A-0x002D] [0x0000]**

UIPR 은 Unreachable Packet 을 수신했을 때(IR[UNR] = '1'), 수신한 Packet 의 IP Address 로 설정된다.

Ex) UIPR = "192.169.0.21"

UIPR0(0x002A)	UIPR1(0x002B)	UIPR2(0x002C)	UIPR3(0x002D)
192(0xC0)	168(0xA8)	0(0x00)	21(0x15)

### 3.1.18 UPORTR (Unreachable Port Register)

**[RO] [0x002E-0x002F] [0x0000]**

UPORTR 은 Unreachable Packet 을 수신했을 때(IR[UNR] = '1'), 수신한 Packet 의 Port 로 설정된다.

Ex) UPORTR = 3000 (0x0BB8)

UPORTR0(0x002E)	UPORTR1(0x002F)
0x0B	0xB8

### 3.1.19 MR2 (Mode Register 2)

**[R=W] [0x0030] [0x40]**

MR2 는 System Operation Clock(SYS\_CLK), INTn 활성화, TCP & UDP Port Scan 방지 등을 설정한다.

7	6	5	4	3	2	1	0
CLKSEL	IEN	NOTCPRST	UDPURB	WOL	-	FARP	-
R=W	R=W	R=W	R=W	R=W	W0	R=W	W0

Bit	Symbol	Description						
7	CLKSEL	<b>System Operation Clock(SYS_CLK) Select</b> CLKLCKR(Clock Lock Register)이 Unlock 일 때만 설정 가능하다. (참조 <a href="#">3.1.41 CLKLCKR (Clock Lock Register)</a> )  1 : 25MHz 0 : SYS_CLK 은 PHYCR1[PWDN] 설정에 따라 변경된다. <table><tr><th>PHYCR1[PWDN]</th><th>SYS_CLK</th></tr><tr><td>0</td><td>100 MHz</td></tr><tr><td>1</td><td>25 MHz</td></tr></table>	PHYCR1[PWDN]	SYS_CLK	0	100 MHz	1	25 MHz
PHYCR1[PWDN]	SYS_CLK							
0	100 MHz							
1	25 MHz							
6	IEN	<b>INTn Enable / Disable</b> 1 : INTn Enable 0 : INTn Disable (INTn is Always High)						
5	NOTCPRST	<b>TCP RST Packet Block</b> 상대방이 Listen 하지 않는 Port 로 접속을 시도할 경우, 일반적으로 RST Packet 을 전송한다. 이는 Port Scan 방식의 공격대상이 되며, 이를 방지하기 위해 RST Packet 전송을 Block 한다.  1 : Block sending RST Packet 0 : Normal Operation						
4	UDPURB	<b>Port Unreachable Packet Block</b> 상대방이 Open 되지 않은 Port 로 UDP Packet 을 전송할 경우, 일반적으로 Destination Port Unreachable Packet 전송한다. 이는 Port scan 방식의 공격대상이 되며, 이를 방지하기 위해서 Port Unreachable Packet 전송을 Block 한다.  1 : Block sending Port Unreachable Packet 0 : Normal Operation						
3	WOL	<b>Wake On LAN</b> WOL Packet 의 수신 여부를 설정한다. PHYCR1[WOL]과 함께 설정해야 한다.  1 : Receive WOL Packet 0 : No Receive WOL Packet						
2	-	Reserved						
1	FARP	<b>Force ARP</b> UDP 와 IPRAW Mode 에서, 일반적으로 Sn_DIPR (Destination IP Address)이 변경된 경우 ARP 과정을 수행하나, 이 Bit 가 ‘1’인 경우 매 Packet 전송 마다 ARP 과정을 수행한다.						

		1 : Set Force ARP 0 : Normal Operation
0	-	Reserved

### 3.1.20 PHAR (Destination Hardware Address Register on PPPoE)

**[R=W] [0x0032-0x0037] [0x0000]**

PPPoE Destination Hardware Address 를 설정한다.

PHAR 은 PPPoE 일 때만 유효하다.

Ex) PHAR = "11:22:33:AA:BB:CC"

PHAR0(0x0032)	PHAR1(0x0033)	PHAR2(0x0034)
0x11	0x22	0x33
PHAR3(0x0035)	PHAR4(0x0036)	PHAR5(0x0037)
0xAA	0xBB	0xCC

### 3.1.21 PSIDR (Session ID Register on PPPoE)

**[R=W] [0x0038-0x0039] [0x0000]**

PPPoE Session ID 를 설정한다.

PSIDR 은 PPPoE 일 때만 유효하다.

Ex) PSIDR = 0x1234

PSIDR0(0x0038)	PSIDR1(0x0039)
0x12	0x34

### 3.1.22 PMRUR (PPPoE Maximum Receive Unit Register)

**[R=W] [0x003A-0x003B] [0xFFFF]**

PMRUR 은 PPPoE 에서 MRU(Maximum Receive Unit)를 설정하며, 1472 보다 큰 값을 설정하면 자동으로 1472 로 설정된다. PMRUR 은 SOCKET 생성(Sn\_CR [OPEN] = '1') 전에 설정되어야 한다. PMRUR 은 PPPoE 일 때만 유효하다.

Ex) PMUR = 1000 (0x03E8)

PMUR0(0x0038)	PMUR1(0x0039)
0x03	0xE8

### 3.1.23 PHYSR0 (PHY Status Register 0)

**[RO] [0x003C] [0x00]**

PHYSR0 은 PHYCR0(PHY Control Register 0)을 통해 설정된 PHY Operation Mode 및 LINK 상태를 확인 할 수 있다.

7	6	5	4	3	2	1	0
CABOFF	-	MODE2	MODE1	MODE0	FDPX	FSPD	LINK
RO		RO	RO	RO	RO	RO	RO

Bit	Symbol	Description				
7	CABOFF	<b>Cable OFF Bit</b> 1 : Cable Unplugged 0 : Cable Plugged				
6	-	Reserved				
[5:3]	MODE		MODE2	MODE1	MODE0	Description
			0	x	x	Auto Negotiation
			1	0	0	100BASE-TX FDX
			1	0	1	100BASE-TX HDX
			1	1	0	10BASE-TX FDX
			1	1	1	10BASE-TX HDX
2	DPX	<b>Duplex Bit (When Link up)</b> 1 : Half Duplex 0 : Full Duplex				
1	SPD	<b>Speed Bit (When Link up)</b> 1 : 10Mbps 0 : 100Mbps				
0	LNK	<b>Flag Link Bit</b> 1 : Link Up 0 : Link Down				

### 3.1.24 PHYSR1 (PHY Status Register 1)

**[RO][0x003D][0x81]**

PHYSR1 은 PHY 의 ACT, RXP, LPI Mode 및 CAL 상태를 확인할 수 있다.

7	6	5	4	3	2	1	0
ACT					RXP	LPI	CAL
RO					RO	RO	RO

Bit	Symbol	Description
7	ACT	PHY Act 1 : PHY Act 0 : PHY No Act



[6:3]	-	Reserved
2	RXP	RX PAGE 1 : A New Page Received 0 : A New Page is not Received
1	LPI	LPI(Low Power Idle) Mode 1 : Enable LPI Mode 0 : Disable LPI Mode
0	CAL	Calibrate Done 1 : Calibration is Removed 0 : Normal

### 3.1.25 PHYRAR (PHY Register Address Register)

**[R=W] [0x003F] [0x00]**

내장 Ethernet PHY 의 Register Address 를 설정한다.

7	6	5	4	3	2	1	0
-	-	-	A4	A3	A2	A1	A0
			R=W	R=W	R=W	R=W	R=W

Bit	Symbol	Description
[7:5]	-	Reserved
[4:0]	A[4:0]	PHY Register Address

### 3.1.26 PHYDIR (PHY Data Input Register)

**[R=W] [0x0040-0x0041] [0x0000]**

PHYDIR 은 PHYAR 로 지정된 PHY Register 값을 설정한다.

Ex) PHYDIR = 0x1234

PHYDIR0(0x0040)	PHYDIR1(0x0041)
0x34	0x12

### 3.1.27 PHYDOR (PHY Data Output Register)

**[RO] [0x0042-0x0043] [0x0000]**

PHYDOR 은 PHYAR 로 지정된 PHY Register 의 값을 Read 한다.

Ex) PHYDOR = 0x1234

PHYDOR0(0x0042)	PHYDPR1(0x0043)
0x34	0x12

### 3.1.28 PHYACR (PHY Access Control Register)

[AC] [0x0044] [0x00]

PHYRAR 로 지정된 PHY Register 의 Access Type 를 설정한다.

Access Type	Value
Write	0x01
Read	0x02

### 3.1.29 PHYDIVR (PHY Division Register)

[R=W] [0x0045] [0x01]

내장 Ethernet PHY 제어를 위해 System Operation Clock(SYS\_CLK)에서 분주된 Clock 을 사용하며, 분주된 이 Clock 은 2.5MHz 를 초과하지 않도록 주의한다.

Value	Divider	SYS_CLK=100MHz	SYS_CLK=25MH
0x00	1/32	3.125MHz (N/A)	781.25KHz
0x01	1/64	1.5625MHz	390.625KHz
others	1/128	781.25KHz	195.3125KHz

### 3.1.30 PHYCR0 (PHY Control Register 0)

[WO] [0x0046] [0x00]

PHYCR0 는 PHYLOCKR(PHY Lock Register)이 Unlock 일 때 Ethernet PHY Operation Mode 를 설정한다. PHYCR0 를 통해 설정된 Bit 들은 PHYSR0 [5:3]로 확인한다.

7	6	5	4	3	2	1	0
-	-	-	-	-	MODE2	MODE1	MODE0
					WO	WO	WO

Bit	Symbol	Description				
[7:3]	-	Reserved				
[2:0]	MODE		MODE2	MODE1	MODE0	Description
			0	x	x	Auto Negotiation
			1	0	0	100BASE-TX FDX
			1	0	1	100BASE-TX HDX
			1	1	0	10BASE-TX FDX
			1	1	1	10BASE-TX HDX

### 3.1.31 PHYCR1 (PHY Control Register 1)

[R=W] [0x0047] [0x41]

PHYCR1 는 PHYLOCKR(PHY Lock Register)를 Unlock 일 때, PHY 의 Power Down, HW Reset 을 설정한다.

7	6	5	4	3	2	1	0
WOL	-	PWDN	-	-	-	-	RST
W0		R=W					AC

Bit	Symbol	Description
7	WOL	<b>Wake On LAN</b> WOL Packet 의 수신 여부를 설정한다. MR2[WOL]과 함께 설정되어야 한다.  1 : Receive WOL Packet 0 : No Receive WOL Packet
6	-	Reserved
5	PWDN	<b>PHY Power Down</b> 1 : Enable Power Down Mode, MR2[CLKSEL]='1'인 경우, SYS_CLK 은 25MHz 로 변경된다. 0 : Disable Power Down Mode  (참조 <a href="#">7.4.1 Reset Timing</a> )
[4:1]	-	Reserved
0	RST	<b>PHY Reset</b> PHY HW Reset 시, SYS_CLK 은 25MHz 로 변경된다. PHY Reset 완료는 PHYSR1[ACT]='1'을 통해 확인할 수 있으며 Reset 이후 SYS_CLK 은 이전 설정 상태로 변경된다. (참조 <a href="#">7.4.1 Reset Timing</a> )  1 : PHY HW Reset 0 : Normal Operation

### 3.1.32 SLCR (SOCKET-less Command Register)

**[RW] [0x004C] [0x00]**

SLCR 은 ARP 또는 PING Request 전송 Command 를 수행한다. Command 는 수행 완료 후 Auto Clear 되며, Auto Clear 전에 다른 Command 를 수행할 수 없다. Command 수행 결과는 SLIR(SOCKET-less Interrupt Register)로 확인한다.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	ARP	PING

-	-	-	-	-	-	AC	AC
---	---	---	---	---	---	----	----

Bit	Symbol	Description
[7:2]	-	Reserved
1	ARP	<b>ARP Request Transmission Command</b> 1 : ARP Request 를 전송을 수행한다. 0 : Ready
0	PING	<b>PING Request Transmission Command</b> 1 : PING Request 를 전송한다. 0 : Ready

### 3.1.33 SLRTR (SOCKET-less Retransmission Time Register)

[R=W] [0x004D-0x004E] [0x07D0]

SLRTR 은 SLCR 의 Retransmission Time 을 설정한다. 단위는 100us 이다. SLCR 에 의해 전송된 ARP 또는 PING Request 에 대한 응답이 없는 경우 Request 를 재전송하며, SLRCR (SOCKET-less Retransmission Count) 초과 전송 시 Timeout 시 발생한다. (SLIR [TIMEOUT] = '1').

(참조 [4.8 Retransmission](#))

Ex) SLRTR = 5000 (0x1388),

$$5000 * 100\mu s = 0.5s$$

SLRTR0(0x004D)	SLRTR1(0x004E)
0x013	0x88

### 3.1.34 SLRCR (SOCKET-less Retransmission Count Register)

[R=W] [0x004F] [0x00]

SLRCR 은 SLCR 의 Retransmission Counter 를 설정한다.

Retransmission Counter 가 SLRCR 을 초과하면 SLIR[TIMEOUT] = '1'이 된다.

(참조 [4.8 Retransmission](#))

### 3.1.35 SLPIPR (SOCKET-less Peer IP Address Register)

[R=W] [0x0050-0x0053] [0x00000000]

SLPIPR 은 SLCR 에 의한 ARP 또는 PING Request Packet 을 전송할 Peer IP Address 를 설정한다.

Ex) SLPIPR = "192.169.0.21"

SLPIPR0(0x0050)	SLPIPR1(0x0051)	SLPIPR2(0x0052)	SLPIPR3(0x0053)
192(0xC0)	168(0xA8)	0(0x00)	21(0x15)

### 3.1.36 SLPHAR (SOCKET-less Peer Hardware Address Register)

**[RO] [0x0054-0x0059] [0x000000000000]**

SLPHAR 은 SLCR 에 의한 ARP Reply 를 수신한 경우(SLIPR[ARP]='1'), Peer Hardware Address 로 설정된다.

Ex) SLPHAR = "11:22:33:AA:BB:CC"

SLPHAR0(0x0054)	SLPHAR1(0x0055)	SLPHAR2(0x0056)
0x11	0x22	0x33
SLPHAR3(0x0057)	SLPHAR4(0x0058)	SLPHAR5(0x0059)
0xAA	0xBB	0xCC

### 3.1.37 PINGSEQR (PING Sequence-number Register)

**[R=W] [0x005A-0x005B] [0x0000]**

PINGSEQR 은 PING Request Packet 의 Sequence Number 를 설정하며, 자동 증가하지 않는다.

Ex) PINGSEQR = 1000 (0x03E8)

PINGSEQR0(0x005A)	PINGSEQR1(0x005B)
0x03	0xE8

### 3.1.38 PINGIDR (PING ID Register)

**[R=W] [0x005C-0x005D] [0x0000]**

PINGIDR 은 PING Request Packet 의 ID 를 설정한다.

Ex) PINGIDR = 256 (0x0100)

PINGIDR0(0x005C)	PINGIDR1(0x005D)
0x01	0x00

### 3.1.39 SLIMR (SOCKET-less Interrupt Mask Register)

**[R=W] [0x005E] [0x00]**

SLIMR 은 SLIR 에 1:1 로 대응되는 Bit 를 Mask 한다.

7	6	5	4	3	2	1	0
-	-	-	-	-	TIMEOUT	ARP	PING
-	-	-	-	-	R=W	R=W	R=W

Bit	Symbol	Description
[7:3]	-	Reserved
2	TIMEOUT	TIMEOUT Interrupt Mask

		1 : Enable TIMEOUT Interrupt 0 : Disable TIMEOUT Interrupt
1	ARP	<b>ARP Interrupt Mask</b> 1 : Enable ARP Interrupt 0 : Disable ARP Interrupt
0	PING	<b>PING Interrupt Mask</b> 1 : Enable PING Interrupt 0 : Disable PING Interrupt

### 3.1.40 SLIR (SOCKET-less Interrupt Register)

**[RW] [0x005F] [0x00]**

SLIR 은 SLIR 에 해당하는 Event 가 발생했을 때 해당 Event Bit 가 1 로 설정된다.

SLIR 의 Event 가 발생하고 SLIMR 의 1:1 대응되는 Bit 가 '1'로 설정되고, 내부 Interrupt Pending Timer Count 가 0 인 경우, INTn 은 Low assert 되며, IR 의 Event 가 Clear 되거나 해당 Mask Bit 가 '0'으로 설정된 경우 High De-assert 된다.

7	6	5	4	3	2	1	0
-	-	-	-	-	TIMEOUT	ARP	PING
-	-	-	-	-	WC	WC	WC

Bit	Symbol	Description
[7:3]	-	Reserved
2	TIMEOUT	<b>TIMEOUT Interrupt</b> TIMEOUT 발생 시, '1'로 설정된다.
1	ARP	<b>ARP Interrupt</b> ARP Reply 수신 시, '1'로 설정된다.
0	PING	<b>PING Interrupt</b> PING Reply 수신 시, '1'로 설정된다.

### 3.1.41 CLKLOCKR (Clock Lock Register)

**[WO] [0x0070] [0x00]**

CLKLOCKR 의 초기값은 Lock 이며, Unlock(0xCE)으로 설정될 경우, MR2[CLKSEL]을 설정할 수 있다. 이후 별도의 Lock(Others) 설정전에는 Unlock 상태를 유지한다.

Unlock	Lock
0xCE	Others

### 3.1.42 NETLCKR (Network Lock Register)

**[WO] [0x0071] [0x00]**

NETLCKR 의 초기값은 Unlock 이며, Unlock 일 경우 GWR, SUBR, SHAR, 외 SIPR 을 설정할 수 있다. 별도의 Lock(0xC5) 설정 이전에는 Unlock 상태를 유지한다.

Unlock	Lock
0x3A	0xC5

### 3.1.43 PHYLCKR (PHY Lock Register)

**[WO] [0x0072] [0x00]**

PHYLCKR 의 초기값은 Lock 이며, Unlock(0x53)으로 설정될 경우, PHYCR0 와 PHYCR1 을 설정할 수 있다. 이후 별도의 Lock(Others) 설정전에는 Unlock 상태를 유지한다.

Unlock	Lock
0x53	Others

### 3.1.44 VERR (Version Register)

**[RO] [0x0080] [0x51]**

VERR 는 W5100S Version 이다.

### 3.1.45 TCNTR (Ticker Counter Register)

**[RO][0x0082-0x0083][0x0000]**

TCNTR 은 W5100S 내부 카운터이며 SYS\_CLK 이 동작할 때 자동으로 증가한다. 단위는 100us 이다.

### 3.1.46 TCNTCLR (Ticker Counter Clear Register)

**[WO][0x0088][0x00]**

TCNTCLR 의 Write Access 를 통해 TCNTR 을 Clear 시킬 수 있다.

## 3.2 SOCKET Register

### 3.2.1 Sn\_MR (SOCKET n Mode Register)

[R=W] [0x0000+0x0100\*(n+4)] [0x00]

Sn\_MR 은 SOCKET Mode 와 Option 을 설정한다.

Sn\_MR 은 SOCKET 생성(Sn\_CR[OPEN] = '1') 전에 설정한다.

7	6	5	4	3	2	1	0
MULTI	MF	ND / MC	-	P3	P2	P1	P0
R=W	R=W	R=W	-	R=W	R=W	R=W	R=W

Bit	Symbol	Description
7	MULTI	<b>UDP Multicast</b> UDP Mode 에서만 유효하다. (참조 <a href="#">4.4.3 UDP Multicast</a> )  1 : Enable UDP Multicast 0 : Disable UDP Multicast
6	MF	<b>MAC Filter Enable</b> MACRAW Mode 에서만 유효하다.  1 : Enable MAC Filter Multicast, Broadcast 와 Source MAC(SHAR) 주소 Packet 만 수신 0 : Disable MAC Filter 모든 Packet 을 수신
5	ND / MC	<b>No Delayed ACK(ND)</b> TCP Mode 에서만 유효하다.  1 : Enable No Delayed ACK, Packet 수신 시 ACK Packet 전송 0 : Disable No Delayed ACK, Sn_RTR(SOCKET n Retransmission Time Register) 시간 이 후 ACK Packet 전송  (참조, ND 설정과 무관하게 Sn_CR[RECV] 수행 이후, SOCKET n Window Size 가 MSS 보다 작은 경우는 ACK Packet 을 곧바로 전송한다.)  <b>Multicast IGMP Version(MC)</b> UDP Multicast Mode 에서만 유효하다. (Sn_MR [3:0] = 'UDP' & Sn_MR [MULTI] = '1')  0 : Using IGMP version 2



		1 : Using IGMP version 1
4	-	Reserved
[3:0]	P[3:0]	<b>Protocol Mode</b> SOCKET Protocol Mode 를 설정한다. MACRAW Mode 는 SOCKET 0 에서만 유효하다.

P3	P2	P1	P0	Protocol Mode
0	0	0	0	SOCKET Closed
0	0	0	1	TCP
0	0	1	0	UDP
0	0	1	1	IPRAW
0	1	0	0	MACRAW

### 3.2.2 Sn\_CR (SOCKET n Command Register)

[RW][AC] [0x0001+0x0100\*(n+4)] [0x00]

Sn\_CR 은 SOCKET Command 를 수행한다. Command 는 수행 완료 후 Auto Clear 되며, Auto Clear 전에는 다른 Command 를 수행할 수 없다.

Value	Symbol	Description												
0x01	OPEN	<b>SOCKET OPEN Command</b> SOCKET 을 OPEN 한다. Sn_MR[3:0]에 따른 Command 수행 결과는 Sn_SR(SOCKET n Status Register)로 확인한다.												
		<table><tr><th>Sn_MR (P[3:0])</th><th>Sn_SR</th></tr><tr><td>Sn_MR_CLOSE ('0000')</td><td>SOCK_CLOSED (0x00)</td></tr><tr><td>Sn_MR_TCP ('0001')</td><td>SOCK_INIT (0x13)</td></tr><tr><td>Sn_MR_UDP ('0010')</td><td>SOCK_UDP (0x22)</td></tr><tr><td>Sn_MR_IPRAW ('0011')</td><td>SOCK_IPRAW (0x32)</td></tr><tr><td>SO_MR_MACRAW ('0100')</td><td>SOCK_MACRAW (0x42)</td></tr></table>	Sn_MR (P[3:0])	Sn_SR	Sn_MR_CLOSE ('0000')	SOCK_CLOSED (0x00)	Sn_MR_TCP ('0001')	SOCK_INIT (0x13)	Sn_MR_UDP ('0010')	SOCK_UDP (0x22)	Sn_MR_IPRAW ('0011')	SOCK_IPRAW (0x32)	SO_MR_MACRAW ('0100')	SOCK_MACRAW (0x42)
		Sn_MR (P[3:0])	Sn_SR											
		Sn_MR_CLOSE ('0000')	SOCK_CLOSED (0x00)											
		Sn_MR_TCP ('0001')	SOCK_INIT (0x13)											
		Sn_MR_UDP ('0010')	SOCK_UDP (0x22)											
		Sn_MR_IPRAW ('0011')	SOCK_IPRAW (0x32)											
SO_MR_MACRAW ('0100')	SOCK_MACRAW (0x42)													
0x02	LISTEN	<b>TCP LISTEN Command</b> SOCK_INIT 상태에서 상대방의 접속을 대기한다. (참조 <a href="#">4.3.1 TCP Server</a> )												
0x04	CONNECT	<b>TCP CONNECT Command</b> SOCK_INIT 상태에서 상대방에게 접속을 요청한다. (참조 <a href="#">4.3.2 TCP Client</a> )												
0x08	DISCON	<b>TCP DISCON Command</b> SOCK_ESTABLISHED (Sn_SR = '0x17') 또는 SOCK_CLOSE_WAIT (Sn_SR = '0x1C')상태일 때, 상대방에게 접속 해제를 요청한다.												

		즉, 접속중인 상대방에게 Disconnect-Request (FIN Packet)를 전송한다.
0x10	CLOSE	<b>SOCKET CLOSE Command</b> SOCKET 을 CLOSE 한다. Sn_SR 은 이전 상태와 상관없이 무조건 SOCK_CLOSED 로 변경된다.  <i>* CAUTION : TCP 인 경우, FIN Packet 전송 없이 강제 CLOSE 된다.</i>
0x20	SEND	<b>SOCKET SEND Command</b> Sn_TX_WR (SOCKET n TX Write Point Register)과 Sn_TX_RD (SOCKETn Rx Read Pointer Register)로 계산된 Size 의 Data 를 전송하며, 계산된 전송 Size 는 Sn_TX_FSR (SOCKETn TX Free Buffer Size Register)를 초과할 수 없다. HOST 는 Sn_IR [SENDOK]='1'를 확인 후 그 다음 SEND Command 를 수행할 수 있다.  TCP 나 UDP Mode 에서, 계산된 Size 의 Data 가 MSS(Maximum Segment Size)를 초과할 경우 자동으로 MSS 단위로 나누어 전송한다.  IPRAW 나 MACRAW Mode 에서, 계산된 Size 의 Data 가 MSS 를 초과할 경우 HOST 는 MSS 단위로 직접 나누어 전송한다.  TCP Mode 에서, Data 를 상대방에게 성공적으로 전송한 경우 (상대방 으로부터 ACK 를 수신한 경우), Sn_TX_FSR 는 전송한 Data Size 만큼 증가한다. 그렇지 못한 경우(상대방으로부터 ACK 를 수신하지 못한 경우)는 Sn_IR [TIMEOUT] = '1'이 되고 Sn_SR 은 SOCK_CLOSED 로 변경된다.  UDP, IPRAW, MACRAW Mode 에서, Sn_IR[SENDOK]='1' 이후에 Sn_TX_FSR 은 전송한 Data Size 만큼 증가한다.
0x21	SEND_MAC	<b>SOCKET SEND_MAC Command</b> SEND_MAC 은 UDP Mode/IPRAW Mode 일 때만 사용된다. 기본동작은 SEND Command 와 같다. SEND Command 는 자동으로 ARP-Process 를 통해 Destination Hardware Address 를 얻은 후 Data 를 전송 하는 반면, SEND_MAC Command 는 HOST 가 설정한 Sn_DHAR (SOCKET n Destination Hardware Address Register)을 Destination Hardware Address 로 설정하여 ARP-Process 없이 Data 를 전송한다.
0x22	SEND_KEEP	<b>TCP SEND_KEEP Command</b> SEND_KEEP Command 는 TCP Mode 일 때만 사용된다. 또한 SEND_KEEP Command 설정 이전에는 최소 1 Byte 이상의 Data 를 전송해야 한다. SEND_KEEP Command 는 상대방에게 Keep alive(KA) Packet 을 송신하여 Connection 이 유효한지 확인한다. 만약 상대방이 응답이 없는 경우

		<p>설정된 Retransmission Time 이후 Sn_IR [TIMEOUT] = '1' &amp; Sn_SR = SOCK_CLOSED 이 된다. (참조 4.3.3.2 Keep )</p>
0x40	RECV	<p><b>SOCKET RECV Command</b> HOST 는 SOCKET n RX Buffer Block 에서 수신된 Data 를 읽고 RECV Command 를 통해 Data 를 읽은 만큼 Sn_RX_RD (SOCKET n Read Pointer Register)를 증가시켜야 한다.</p> <p>(참조 3.2.18 Sn_RX_RSR (SOCKET n RX Received Size Register), 3.2.20 Sn_RX_WR (SOCKET n RX Write Pointer Register), 3.2.19 Sn_RX_RD (SOCKET n RX Read Pointer Register))</p>

### 3.2.3 Sn\_IR (SOCKET n Interrupt Register)

[RW] [0x0002+0x0100\*(n+4)] [0x00]

Sn\_IR 은 SOCKET 의 상태변화나 Command 수행 결과를 알려준다.

Sn\_IR 의 Event 가 발생하고, Sn\_IMR 의 1:1 대응되는 Bit 가 설정되어 있을 경우, IR[Sn\_INT] = '1'로 설정된다.

7	6	5	4	3	2	1	0
-	-	-	SENDOK	TIMEOUT	RECV	DISCON	CON
			WC	WC	WC	WC	WC

Bit	Symbol	Description
[7:5]	-	Reserved
4	SENDOK	<b>SEND OK Interrupt</b> 1 : Sn_CR [SEND]을 완료했을 경우 0 : -
3	TIMEOUT	<b>TIMEOUT Interrupt</b> 1 : ARP 또는 TCP 통신과정에서 Sn_RCR (SOCKET Retransmission Count Register)이상의 재전송이 발생한 경우 0 : -
2	RECV	<b>RECEIVED Interrupt</b> 1 : SOCKET n RX Memory 에 DATA 를 수신하거나 Sn_CR [RECV] 수행 후 DATA 가 남아있는 경우 0 : -
1	DISCON	<b>DISCONNECTED Interrupt</b> 1 : 상대방에게 접속해제를 요청하고 접속해제를 완료한 경우, 상대방으로부터 접속해체 요청을 수신한 경우, 혹은 상대방으로부터 RST packet 을 수신한 경우

		0 : -
0	CON	<b>CONNECTED Interrupt</b> 1 : TCP mode 에서 접속이 완료된 경우 0 : -

### 3.2.4 Sn\_SR (SOCKET n Status Register)

**[RO] [0x0003+0x0100\*(n+4)] [0x00]**

Sn\_SR 은 SOCKET n 의 상태를 알려준다. Sn\_SR 는 Sn\_CR 의 Command 수행 또는 DATA 송수신에 의해 변경된다.

Value	Symbol	Description
0x00	SOCK_CLOSED	SOCKET 이 Close 된 상태.
0x13	SOCK_INIT	SOCKET 이 TCP mode 로 Open 된 상태.
0x14	SOCK_LISTEN	SOCKET 이 TCP mode 이고 상대방의 접속을 기다리는 상태.
0x17	SOCK_ESTABLISHED	SOCKET 이 TCP mode 이고 상대방과 접속된 상태.
0x1C	SOCK_CLOSE_WAIT	SOCKET 이 TCP mode 이고 접속해제 요청을 수신한 상태.
0x22	SOCK_UDP	SOCKET 이 UDP mode 로 Open 된 상태.
0x32	SOCK_IPRAW	SOCKET 이 IPRAW Mode 로 Open 된 상태.
0x42	SOCK_MACRAW	SOCKET 이 MACRAW Mode 로 Open 된 상태.

아래 SOCKET status 은 Sn\_SR 의 전이 과정에서 관찰될 수 있는 temporary Status 들이다.

Value	Symbol	Description
0x15	SOCK_SYSENT	Connect-request 를 전송한 상태.
0x16	SOCK_SYNRCV	Connect-request 를 수신한 상태.
0x18	SOCK_FIN_WAIT	SOCKET 이 Closing 되는 상태.
0x1B	SOCK_TIME_WAIT	
0x1D	SOCK_LAST_ACK	

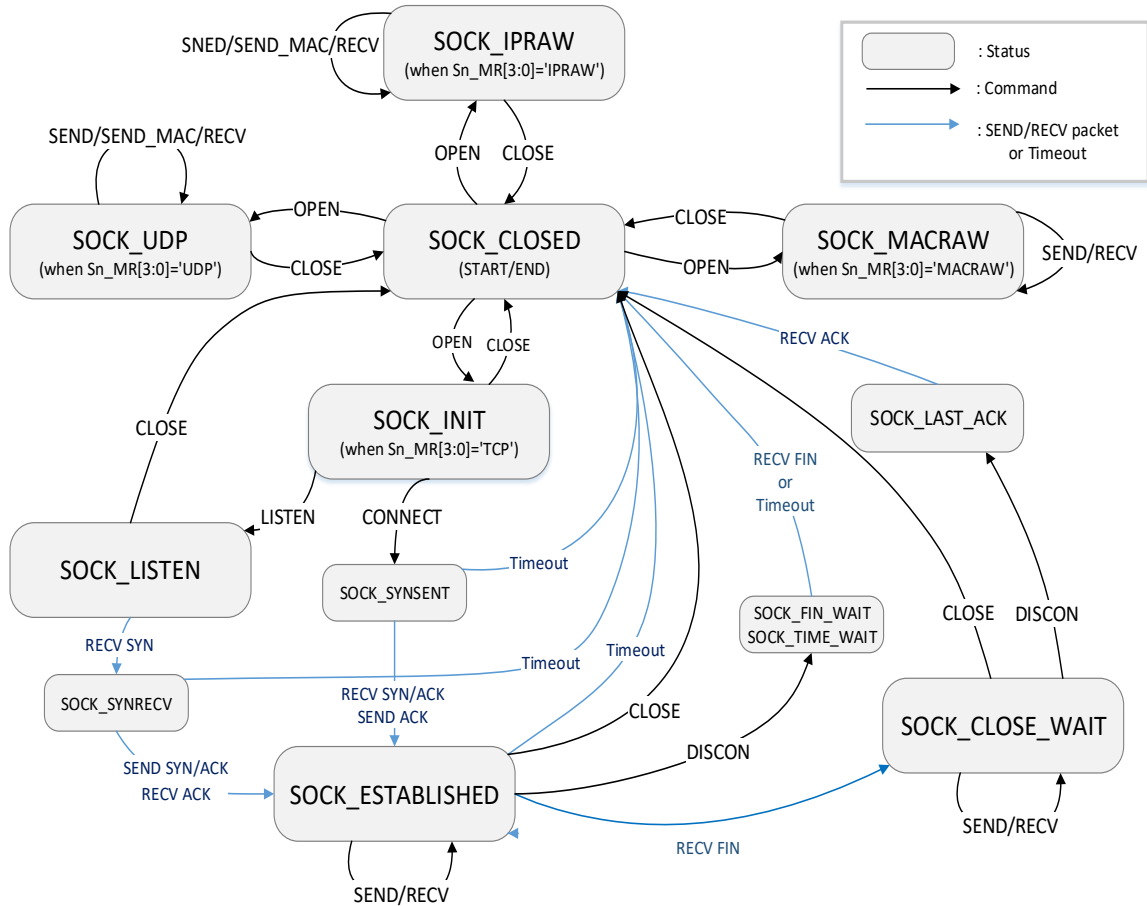


Figure 4 State Diagram

### 3.2.5 Sn\_PORTR (SOCKET n Source Port Register)

**[R=W] [0x0004+0x0100\*(n+4), 0x0005+0x0100\*(n+4)] [0x0000]**

Sn\_PORTR 은 SOCKET n 의 Source Port Number 를 설정한다.

Ex) S0\_PORTR = 5000 (0x1388)

S0_PORTR0(0x0404)	S0_PORTR1(0x0405)
0x013	0x88

### 3.2.6 Sn\_DHAR (SOCKET n Destination Hardware Address Register)

**[RW] [0x0006+0x0100\*(n+4), 0x0007+0x0100\*(n+4), 0x0008+0x0100\*(n+4), 0x0009+0x0100\*(n+4), 0x000A+0x0100\*(n+4), 0x000B+0x0100\*(n+4)] [0x0000000000000000]**

Sn\_DHAR 는 상대방의 MAC Address 이다.

TCP 인 경우, 상대방과 접속되었을 때(Sn\_SR = 'SOCK\_ESTABLISHED') 상대방의 MAC Address 가 설정된다.

UDP, IPRAW 인 경우, Sn\_CR[SEND\_MAC] 사용시 상대방의 MAC Address 를 설정한다.

UDP Multicast 인 경우, Multicast Group 의 MAC Address 를 설정한다.

(참조 4.4.3 UDP Multicast)

Ex) SO\_DHAR = “11:22:33:AA:BB:CC”

SO_DHAR0(0x0406)	SO_DHAR1(0x0407)	SO_DHAR2(0x0408)
0x11	0x22	0x33
SO_DHAR3(0x0409)	SO_DHAR4(0x040A)	SO_DHAR5(0x040B)
0xAA	0xBB	0xCC

### 3.2.7 Sn\_DIPR (SOCKET n Destination IP Address Register)

**[RW] [0x000C+0x0100\*(n+4), 0x000D+0x0100\*(n+4), 0x000E+0x0100\*(n+4), 0x000F+0x0100\*(n+4)] [0x00000000]**

Sn\_DIPR 은 상대방의 IP Address 이다.

TCP 인 경우, 접속할 상대방의 IP Address 를 설정하거나, 접속된 상대방의 IP Address 를 확인한다.

UDP, IPRAW 인 경우, 전송할 상대방의 IP Address 를 설정한다.

UDP Multicast 인 경우, Multicast Group 의 IP Address 를 설정한다.(참조 4.4.3 UDP Multicast)

UDP, UDP Multicast, IPRAW 인 경우, 수신된 상대방의 IP Address 는 SOCKET n RX Buffer 에서 확인한다.

Ex) SO\_DIPR = “192.168.0.11”

SO_DIPR0(0x040C)	SO_DIPR1(0x040D)	SO_DIPR2(0x040E)	SO_DIPR3(0x040F)
192 (0xC0)	168 (0xA8)	0 (0x00)	11 (0x0B)

### 3.2.8 Sn\_DPORTR (SOCKET n Destination Port Register)

**[R=W] [0x0010+0x0100\*(n+4), 0x0011+0x0100\*(n+4)] [0x0000]**

Sn\_DPORTR 은 상대방의 Port 이다.

TCP 인 경우, 접속할 상대방의 Port 를 설정하거나, 접속된 상대방의 Port 를 확인한다.

UDP 인 경우, 전송할 상대방의 Port 를 설정한다.

UDP Multicast 인 경우, Multicast Group 의 Port 를 설정한다.(참조 4.4.3 UDP Multicast)

UDP, UDP Multicast 인 경우, 수신된 상대방의 Port 는 SOCKET n RX Buffer 에서 확인한다.

Ex) SO\_DPORTR = 5000 (0x1388),

SO_DPORTR0(0x0410)	SO_DPORTR1(0x0411)
0x13	0x88

### 3.2.9 Sn\_MSS (SOCKET n Maximum Segment Size Register)

**[RW] [0x0012+0x0100\*(n+4), 0x0013+0x0100\*(n+4)] [0x0000]**

Sn\_MSS 는 SOCKET 의 MSS 를 설정한다.

SOCKET 의 MSS 는 Sn\_CR[OPEN] 이전에 설정해야 한다.

최대 MSS 를 초과 설정 할 경우, 내부적으로 최대 MSS 을 사용한다.

Mode	Normal(MR[PPPoE]='0') Range	PPPoE(MR[PPPoE]='1') Range
TCP	1~1460	1~1452
UDP	1~1472	1~1464
IPRAW	1480	1472
MACRAW	1514	

Ex) SO\_MSS = 1460 (0x05B4),

SO_MSS0(0x0412)	SO_MSS1(0x0413)
0x05	0xB4

### 3.2.10 Sn\_PROTOR (SOCKET n IP Protocol Register)

**[R=W] [0x0014+0x0100\*(n+4)] [0x0000]**

Sn\_PROTOR 은 IPRAW Mode 에서 IGMP(0x01), TCP(0x06), UDP(0x11)를 제외한 Protocol Number 를 설정한다. (참조 [IANA Protocol Numbers](#))

IPRAW 인 경우, Sn\_PROTR 에 설정된 Protocol 만 송수신한다.

Ex) ICMP(Internet Control Message Protocol) = 0x01

### 3.2.11 Sn\_TOS (SOCKET n IP Type Of Service Register)

**[R=W] [0x0015+0x0100\*(n+4)] [0x00]**

Sn\_TOS 는 IP Header 의 TOS field 를 설정한다. (참조 [IANA IP Parameters](#))

### 3.2.12 Sn\_TTL (SOCKET n IP Time To Live Register)

**[R=W] [0x0016+0x0100\*(n+4)] [0x80]**

Sn\_TTL 은 IP header 의 TTL field 를 설정한다. (참조 [IANA IP Parameters](#))

### 3.2.13 Sn\_RXBUF\_SIZE (SOCKET n RX Buffer Size Register)

**[RW] [0x001E+0x0100\*(n+4)] [0x02]**

Sn\_RXBUF\_SIZE 는 SOCKET n 의 RX Buffer Size 를 0, 1, 2, 4, 8 Kbytes 단위로 설정한다.

RX Memory 는 SOCKET 0 부터 SOCKET 3 까지 Sn\_RXBUF\_SIZE 값으로 순차적으로 할당된다.

만약 그 외의 값으로 설정되거나 Sn\_RXBUF\_SIZE 의 총합이 8 Kbytes 를 초과하는 경우 오작동할 수 있다.

Sn\_RXBUF\_SIZE 는 RMSR 을 통해서도 설정할 수 있다.

Value (Dec)	0	1	2	4	8
Buffer size	0KB	1KB	2KB	4KB	8KB

Ex) S0\_RXBUF\_SIZE = 8 Kbytes

S0_RXBUF_SIZE(0x041E)
0x08

### 3.2.14 Sn\_TXBUF\_SIZE (SOCKET n TX Buffer Size Register)

[RW] [0x001F+0x0100\*(n+4)] [0x02]

Sn\_TXBUF\_SIZE 는 SOCKET n 이 TX Buffer Size 를 0, 1, 2, 4, 8 Kbytes 단위로 설정한다.

TX Memory 는 SOCKET 0 부터 SOCKET 3 까지 Sn\_TXBUF\_SIZE 값으로 순차적으로 할당된다.

만약 그 외의 값으로 설정되거나 Sn\_TXBUF\_SIZE 의 총합이 8 Kbytes 를 초과하는 경우 오작동할 수 있다.

Sn\_TXBUF\_SIZE 는 TMSR 을 통해서도 설정할 수 있다.

Value (Dec)	0	1	2	4	8
Buffer size	0KB	1KB	2KB	4KB	8KB

Ex) S0\_TXBUF\_SIZE= 4 Kbytes

S0_TXBUF_SIZE(0x041F)
0x04

### 3.2.15 Sn\_TX\_FSR (SOCKET n TX Free Size Register)

[RO] [0x0020+0x0100\*(n+4), 0x0021+0x0100\*(n+4)] [0x0800]

Sn\_TX\_FSR 은 SOCKET n TX Buffer Block 의 전송 가능한 Buffer Size 를 확인한다.

이 값은 W5100S 에 의해 자동으로 계산된다.

In UDP, IPRAW and MACRAW mode,  
 $Sn\_TX\_FSR = | Sn\_TX\_WR^{(1)} - Sn\_TX\_RD^{(2)} | + 1$   
 In TCP mode,  
 $Sn\_TX\_FSR = | Sn\_TX\_WR - Internal\ Pointer^{(3)} | + 1$

(1) SOCKET n TX Write Pointer Register

(2) SOCKET n TX Read Pointer Register

(3) TCP ACK Pointer managed by W5100S

Sn\_TX\_FSR 보다 큰 DATA 를 SOCKET n TX Buffer Block 에 저장하지 않게 주의한다.

Ex) S0\_TX\_FSR = 1024 (0x0400)

S0_TX_FSR0(0x0420)	S0_TX_FSR1(0x0421)
0x04	0x00



### 3.2.16 Sn\_TX\_RD (SOCKET n TX Read Pointer Register)

**[RO] [0x0022+0x0100\*(n+4), 0x0023+0x0100\*(n+4)] [0x0000]**

Sn\_TX\_RD 는 SOCKET n TX Buffer block 에서 마지막으로 전송된 DATA 의 주소이다.

Sn\_TX\_RD 는 Sn\_CR[OPEN]에 의해 초기화된다. TCP 인 경우, TCP 접속과정에서 재설정 된다.

Sn\_CR[SEND]나 Sn\_CR[SEND\_MAC]에 의해 전송할 DATA Size 만큼 증가된 Sn\_TX\_WR 까지,

Sn\_TX\_RD 는 자동 증가한 후, Sn\_IR[SENDOK]를 발생시킨다.

Ex) S0\_TX\_RD = 0xD4B3

S0_TX_RD0(0x0422)	S0_TX_RD1(0x0423)
0xD4	0xB3

### 3.2.17 Sn\_TX\_WR (SOCKET n TX Write Pointer Register)

**[RW] [0x0024+0x0100\*(n+4), 0x0025+0x0100\*(n+4)] [0x0000]**

Sn\_TX\_WR 는 SOCKET n TX Buffer Block 에 마지막으로 저장한 DATA 의 주소이다.

Sn\_TX\_WR 은 Sn\_CR[OPEN]에 의해 초기화된다. TCP 인 경우, TCP 접속과정에서 재설정 된다.

전송할 DATA 를 SOCKET n TX Buffer Block 에 저장하고 해당 DATA Size 만큼 Sn\_TX\_WR 을 증가시킨 후, Sn\_CR[SEND]나 Sn\_CR[SEND\_MAC]를 수행한다.

Ex) S0\_TX\_WR = 0x0800

S0_TX_WR0(0x0424)	S0_TX_WR1(0x0425)
0x08	0x00

### 3.2.18 Sn\_RX\_RSR (SOCKET n RX Received Size Register)

**[RO] [0x0026+0x0100\*(n+4), 0x0027+0x0100\*(n+4)] [0x0000]**

Sn\_RX\_RSR 은 SOCKET n RX Buffer 에 수신된 Data Size 를 알려준다.

이 값은 Sn\_RXBUF\_SIZE 를 초과할 수 없으며 W5100S 에 의해 계산된다.

In TCP, UDP, IPRAW and MACRAW mode,

$$\text{Sn\_RX\_RSR} = | \text{Sn\_RX\_WR}^{(1)} - \text{Sn\_RX\_RD}^{(2)} |$$

(1) SOCKET n RX Write Pointer Register

(2) SOCKET n RX Read Pointer Register

Ex) S0\_RX\_RSR = 2048 (0x0800)

S0_RX_RSR0(0x0426)	S0_RX_RSR1(0x0427)
0x08	0x00

### 3.2.19 Sn\_RX\_RD (SOCKET n RX Read Pointer Register)

**[RW] [0x0028+0x0100\*(n+4), 0x0029+0x0100\*(n+4)] [0x0000]**

Sn\_RX\_RD 는 HOST 가 마지막으로 읽은 SOCKET n RX Buffer Block 의 주소이다.

HOST 는 SOCKET n RX Buffer 에서 Sn\_RX\_RD 부터 Sn\_RX\_WR 까지 저장된 Data 를 Read 할 수 있으며, Read 한 Size 만큼 Sn\_RX\_RD 를 증가시킨다.

Ex) S0\_RX\_RD = 1536(0x0600)

S0_RX_RD0(0x0428)	S0_RX_RD1(0x0429)
0x06	0x00

### 3.2.20 Sn\_RX\_WR (SOCKET n RX Write Pointer Register)

**[RO] [0x002A+0x0100\*(n+4), 0x002B+0x0100\*(n+4)] [0x0000]**

Sn\_RX\_WR 은 SOCKET n RX Buffer Block 에 마지막으로 수신된 Data 의 주소이다.

수신할 Data Size 가 Sn\_RX\_RSR 과 같거나 작은 경우 SOCKET n RX Buffer Block 에 저장되고, Sn\_RX\_WR 는 저장된 Data Size 만큼 증가한다.

Ex) S0\_RX\_WR = 1536(0x0600)

S0_RX_WR0(0x042A)	S0_RX_WR1(0x042B)
0x06	0x00

### 3.2.21 Sn\_IMR (SOCKET n Interrupt Mask Register)

**[R=W] [0x002C+0x0100\*(n+4)] [0xFF]**

Sn\_IMR 은 Sn\_IR 의 1:1 대응되는 Bit 를 Mask 한다.

7	6	5	4	3	2	1	0
-	-	-	SENDOK	TIMEOUT	RCV	DISCON	CON
-	-	-	R=W	R=W	R=W	R=W	R=W

Bit	Symbol	Description
[7:5]	-	Reserved
4	SENDOK	Sn_IR[SENDOK] Interrupt Mask
3	TIMEOUT	Sn_IR[TIMEOUT] Interrupt Mask
2	RCV	Sn_IR[RCV] Interrupt Mask
1	DISCON	Sn_IR[DISCON] Interrupt Mask
0	CON	Sn_IR[CON] Interrupt Mask

### 3.2.22 Sn\_FRAGR (SOCKET n Fragment Offset in IP Header Register)

**[R=W] [0x002D+0x0100\*(n+4), 0x002E+0x0100\*(n+4)] [0x4000]**

Sn\_FRAGR 은 IP Header 의 Fragment field 를 설정한다.

Ex) S0\_FRAG0 = 0x0000 (Don't Fragment)

S0_FRAGR0(0x042D)	S0_FRAGR1(0x042E)
0x00	0x00

### 3.2.23 Sn\_MR2 (SOCKET n Mode register 2)

[R=W] [0x002F+0x0100\*(n+4)] [0x00]

Sn\_MR2 는 Sn\_MR 과 함께 SOCKET n 의 Option 을 설정한다. SOCKET n Option 은 Sn\_CR [OPEN] = '1' 이전에 설정한다.

7	6	5	4	3	2	1	0
-	MBBLK	MMBLK	IPV6BLK	-	-	BRDB	UNIB
-	R=W	R=W	R=W	-	-	R=W	R=W

Bit	Symbol	Description
7	-	Reserved
6	MBBLK	<b>Broadcast Blocking on MACRAW Mode</b> MACRAW Mode 인 경우, Broadcast Packet 의 수신 여부를 설정한다.  0 : Disable Broadcast Blocking 1 : Enable Broadcast Blocking
5	MMBLK	<b>Multicast Blocking on MACRAW Mode</b> MACRAW Mode 의 Multicast Packet 의 수신 여부를 설정한다.  0 : Disable Multicast Blocking 1 : Enable Multicast Blocking
4	IPV6BLK	<b>IPv6 Packet Blocking on MACRAW Mode</b> MACRAW Mode 인 경우, IPv6 Packet 의 수신 여부를 설정한다.  0 : Disable IPv6 Blocking 1 : Enable IPv6 Blocking
[3:2]	-	Reserved
1	BRDB	<b>Broadcast Blocking on UDP Mode/ Force PSH on TCP Mode</b> <b>*Broadcast Blocking on UDP Mode</b> UDP Broadcast Packet 의 수신 여부를 설정한다.  0 : Disable Broadcast Blocking 1 : Enable Broadcast Blocking  <b>* Force PSH flag on TCP Mode</b>

		모든 DATA Packet 에 PSH Flag 를 설정한다.  1: Force PSH Flag 0: No Force PSH Flag
0	UNIB	<b>Unicast Blocking on UDP Multicast Mode</b> Unicast Packet 의 수신 여부를 결정한다.  0 : Disable Unicast Blocking 1 : Enable Unicast Blocking

### 3.2.24 Sn\_KPALVTR (SOCKET n Keep Alive Timer Register)

**[RO] [0x0030+0x0100\*(n+4)] [0x00]**

Sn\_KPALVTR 는 SOCKET n 의 TCP Keep Alive(KA) Packet 의 전송주기를 설정한다.

단위는 5 sec 이다.

KA 전송은 Sn\_SR = ESTABLISHED 이고 Data 를 한번 이상 전송한 경우만 가능하다.

KA Packet 은 설정된 Sn\_KPALVTR 주기 후에 전송되거나, 주기가 설정되지 않은 경우 (Sn\_KPALVRT = '0'), Sn\_CR [SENDKEEP]에 의해 전송된다.

Ex) S0\_KPALVTR = 10 (0x0A),

$$10 * 5s = 50s$$

S0\_KPALVRT(0x0430)

0x0A

### 3.2.25 Sn\_RTR (SOCKET n Retransmission Time Register)

**[R=W] [0x0032+0x0100\*(n+4), 0x0033+0x0100\*(n+4)] [0x0000]**

Sn\_RTR 은 SOCKET n 의 Retransmission Time 을 설정한다.

Sn\_RTR 의 값이 '0'인 경우, Sn\_CR[OPEN]='1'에 의해 RTR 값으로 초기화된다.

(참조 [4.8 Retransmission](#))

Ex) S0\_RTR = 5000 (0x1388),

$$5000 * 100\mu s = 0.5s$$

S0\_RTR0(x0432)

0x013

S0\_RTR1(0x0433)

0x88

### 3.2.26 Sn\_RCR (SOCKET n Retransmission Count Register)

**[R=W] [0x0034+0x0100\*(n+4)] [0x00]**

Sn\_RCR 은 SOCKET n 의 Retransmission Counter 를 한다. Sn\_RCR 의 값이 '0'인 경우, Sn\_CR [OPEN] = '1'에 의해 RCR 값으로 초기화된다. (참조 [4.8 Retransmission](#))

## 4 Functional Description

W5100S 는 간단한 Register 조작만으로 Internet Connectivity 를 제공한다. 이 Chapter 에서는 W5100S 의 초기화와 각 Protocol(TCP, UDP, IPRAW, MACRAW) 및 추가 기능에 따른 Data 통신방법에 대하여 각 단계별로 Pseudo Code 를 기반으로 살펴본다.

### 4.1 W5100S RESET

- Reset 이전에 Hardware 를 설정한다. (참조 [7.4.1 Reset Timing](#))
- MOD[3:0] 설정 : HOST Interface Mode 를 결정한다.
- Hardware Reset : RSTn 에 500ns 이상의 Reset 신호를 인가한다.
- T<sub>STA</sub> 시간 동안 대기한다. : W5100S Stable Time

### 4.2 Initialization

W5100S 초기화는 Network 정보, SOCKET n TX/RX Buffer Block 등을 설정한다.

#### 4.2.1 Basic Setting

W5100S 의 동작을 위하여 아래의 Register 들을 사용자의 Application 에 맞게 설정한다.

- Mode Register (MR)
- Interrupt Mask Register (IMR)
- Retransmission Time Register (RTR)
- Retransmission Count Register (RCR)

위 Register 들의 자세한 내용은 Register Description 을 통해 알 수 있다.

#### 4.2.2 Network Information Setting

통신을 위한 기본 Network Information 설정한다.

```
NETWORK SETTING:
{
    /* W5100S MAC Address, 11:22:33:AA:BB:CC */
    SHAR[0:5] = { 0x11, 0x22, 0x33, 0xAA, 0xBB, 0xCC };

    /* W5100S Gateway IP Address, 192.168.0.1 */
    GAR[0:3] = { 0xC0, 0xA8, 0x00, 0x01 };

    /* W5100S Subnet MASK Address, 255.255.255.0 */
    SUBR[0:3] = { 0xFF, 0xFF, 0xFF, 0x00 };
}
```

```

/* W5100S IP Address, 192.168.0.100 */
SIPR[0:3] = {0xC0, 0xA8, 0x00, 0x64};
}

```

### 4.2.3 SOCKET TX/RX Buffer Setting

TMSR/RMSR 이나, 혹은 Sn\_TXBUF\_SIZE/Sn\_RXBUF\_SIZE 를 이용하여 SOCKET n TX/RX Buffer Size 를 결정한다. SOCKET n TX/RX Buffer 는 RING-Buffer 구조이기 때문에, SOCKET n TX/RX Buffer 제어를 위한 Base Address 와 MASK 를 계산한다. SOCKET n TX/RX Buffer Size 의 각 총합이 TX/RX Memory Block 8 Kbytes 를 초과하지 않도록 주의한다.

다음은 SOCKET n RX/TX Buffer 를 설정하는 Pseudo Code 를 예로 나타낸 것이다.

```

In case of, assign 2Kbytes RX/TX Memory per SOCKET
{
    // set Base Address of TX/RX Memory for SOCKET n
    gSO_RX_BASE = 0x8000; // TX Memory Block Base Address
    gSO_RX_BASE = 0xC000; // RX Memory Block Base Address
    TxTotalSize = 0;      // for check the total size of SOCKET n TX Buffer
    RxTotalSize = 0;      // for check the total size of SOCKET n RX Buffer

    for (n=0; n<3; n++) {
        Sn_TXBUF_SIZE = 2; // assign 2 Kbytes TX Memory per SOCKET
        Sn_RXBUF_SIZE = 2; // assign 2 Kbytes RX Memory per SOCKET
        // 0x07FF, for getting offset address within assigned SOCKET n TX/RX Memory
        gSn_TX_MASK = (1024 * Sn_TXBUF_SIZE) - 1;
        gSn_RX_MASK = (1024 * Sn_RXBUF_SIZE) - 1;
        if( n != 0 ) {
            gSn_TX_BASE = gSn-1_TX_BASE + (1024 * Sn-1_TXBUF_SIZE);
            gSn_RX_BASE = gSn-1_RX_BASE + (1024 * Sn-1_RXBUF_SIZE);
        } // end if
        TxTotalSize = TxTotalSize + Sn_TXBUF_SIZE;
        RxTotalSize = RxTotalSize + Sn_RXBUF_SIZE;
        If( TxTotalSize > 8 or RxTotalSize > 8 ) goto ERROR; // invalid Total Size
    } // end for
}

```

## 4.3 TCP

TCP(Transmission Control Protocol)는 IP Layer 위의 전송계층에 위치하며 1:1 연결 기반의 양방향 데이터 전송 프로토콜이다. 또한 Port Number 를 이용해 Application 간 통신을 제공한다.

TCP 는 1:1 연결 기반이므로 상대방과의 데이터 송수신을 위해서 상대방에게 연결을 요청하거나 상대방으로부터 연결요청을 받아야 한다. 이 과정에서 먼저 연결을 요청하는 쪽을 ‘TCP CLIENT’, 연결요청을 받은 쪽을 ‘TCP SERVER’로 구분한다. TCP 는 또한 신뢰성 있는 전송 프로토콜로써 연결이 완료된 이후의 데이터 송수신 과정에서 확인응답 메커니즘을 통해 전송 데이터의 수신여부를 확인하고 손실되거나 훼손된 데이터는 재전송한다. ‘TCP SERVER’와 ‘TCP CLIENT’는 TCP 연결종료가 이루어지기 전까지 연결을 유지하며 데이터를 송수신한다.

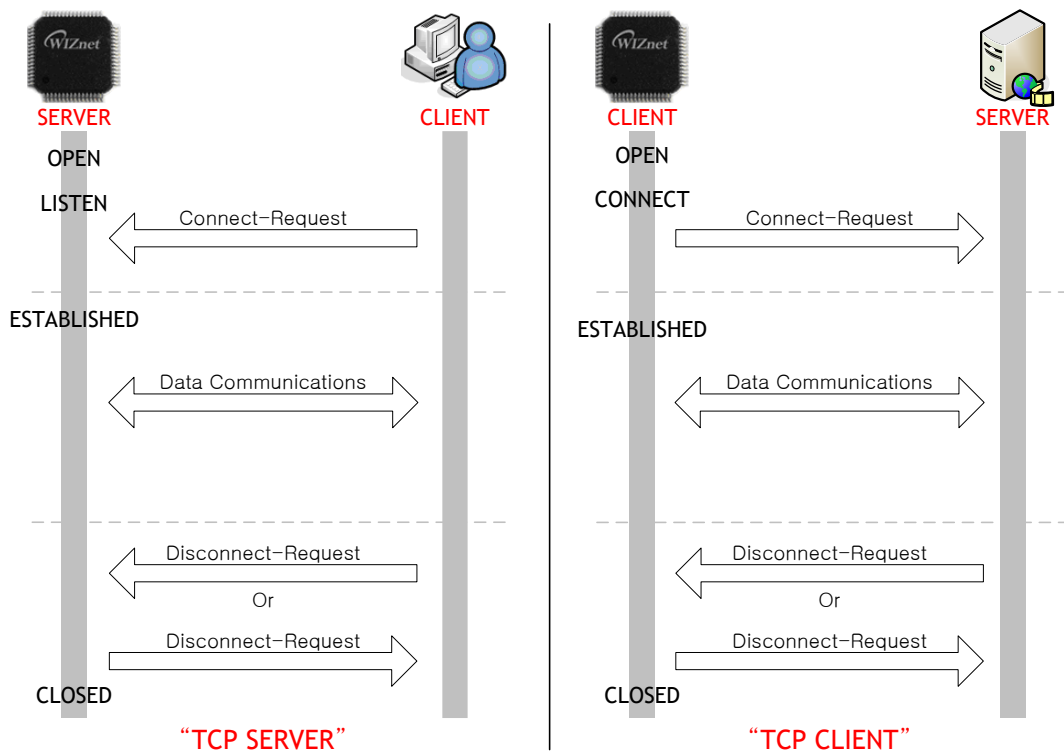


Figure 5 TCP SERVER and TCP CLIENT

### 4.3.1 TCP Server

Figure 6 는 'TCP SERVER'의 동작흐름을 도식화한다.

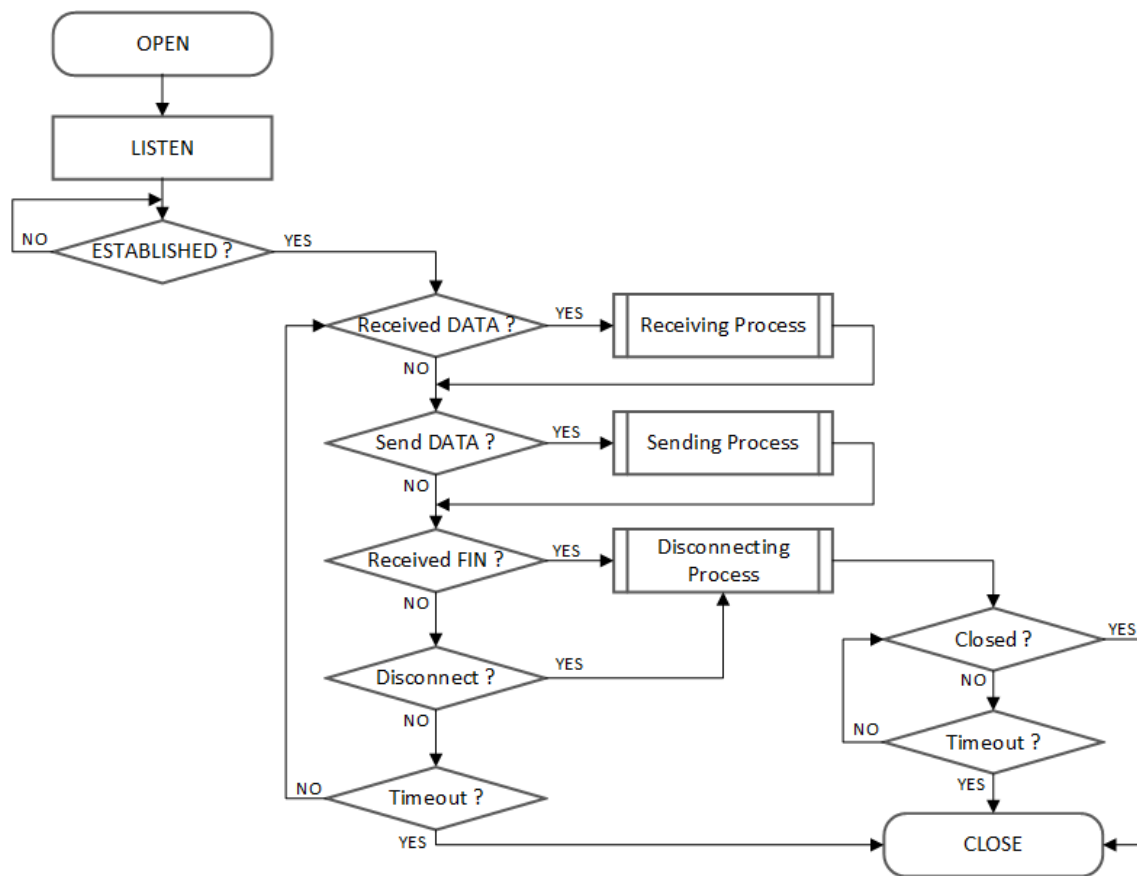


Figure 6 TCP Server Operation Flow

#### ○ OPEN

SOCKET n 을 TCP Mode 로 Open 한다.

```

{
START :
  Sn_MR[3:0] = '0001'; /* set TCP Mode */
  Sn_PORTR[0:1] = {0x13,0x88}; /* set PORT Number, 5000(0x1388) */

  /* Configure SOCKET Option when you need it. */
  // Sn_MR[ND] = '1'; /* set No Delay ACK */

  Sn_CR[OPEN] = '1'; /* set OPEN Command */
  while(Sn_CR != 0x00); /* wait until OPEN Command is cleared*/

  if(Sn_SR != SOCK_INIT) goto START; /* check SOCKET Status */

```



```
}
```

## ○ LISTEN

SOCKETn 을 ‘TCP SERVER’로 동작시킨다. 상대방의 연결요청(SYN packet)을 대기한다.

```
{
    Sn_CR = LISTEN; /* set LISTEN Command */
    while(Sn_CR != 0x00); /* wait until LISTEN Command is cleared*/
    if(Sn_SR != SOCK_LISTEN) goto OPEN; /* check SOCKET Status */
}
```

## ○ ESTABLISHED?

‘TCP SERVER’는 SYN Packet 수신 전까지 대기상태(Sn\_SR=SOCK\_LISTEN)를 유지하며, ‘TCP CLIENT’로부터 SYN Packet 을 수신하면 SYN/ACK Packet 을 전송하고, 연결을 완료한다. 접속이 완료될 경우, Sn\_IR[CON]=‘1’ 이거나 Sn\_SR=SOCK\_ESTABLISHED 로 확인할 수 있다.

First method :

```
{
    /* check SOCKET Interrupt */
    if (Sn_IR[CON] == '1')
    {
        /* clear SOCKET Interrupt */
        Sn_IR[CON] = '1';
        goto Received DATA?; /* or goto Send DATA?; */
    }
    else if(Sn_IR[TIMEOUT] == '1') goto Timeout?;
}
```

Second method :

```
{
    if (Sn_SR == SOCK_ESTABLISHED)
    {
        /* clear SOCKET Interrupt */
        Sn_IR[CON] = '1';
        goto Received DATA? /* or goto Send DATA?; */
    }
    else if(Sn_IR[TIMEOUT] == '1') goto Timeout?;
}
```

## □ Receive DATA?

SOCKETn 의 Data 수신여부는 Sn\_RX\_RSR > 0 또는 Sn\_IR[RECV]='1'을 통해 확인한다.

First method :

```
{
    /* check SOCKET RX Memory Received Size */
    if (Sn_RX_RSR > 0) goto Receiving Process;
}
```

Second method :

```
{
    if (Sn_IR[RECV] == '1')
    {
        /* check SOCKET Interrupt */
        Sn_IR[RECV] = '1'; /* clear SOCKET Interrupt */
        goto Receiving Process;
    }
}
```

## ○ Receiving Process

SOCKET RX Buffer로부터 수신된 Data를 Read한다.

RX Memory Block에서 수신 Data의 Read Offset Address를 [4.2.3 SOCKET TX/RX Buffer Setting](#)에서 계산된 gSn\_RX\_BASE, gSn\_RX\_MASK와 Sn\_RX\_RD를 이용하여 계산하고, Read Offset Address부터 Data를 Read한다. 수신 Data를 Read 후, 그 Size만큼 Sn\_RX\_RD를 증가시키고, Sn\_CR[RECV]='1'를 수행한다. 만약, Sn\_CR[RECV]='1'수행 이후 SOCKETn RX Buffer에 DATA가 여전히 남아있는 경우 Sn\_IR[RECV]='1'이 다시 발생한다.

Read Offset Address 계산시 SOCKETn RX Buffer Block의 경계주소 (n=0,1,2 : gSn\_RX\_BASE ~ gSn+1\_RX\_BASE, n=3 : gS3\_RX\_BASE ~ 0xFFFF)를 벗어나지 않도록 주의한다.

```
{
    /* get Received size */
    get_size = Sn_RX_RSR;

    /* calculate SOCKET n RX Buffer Size & Offset Address */
    gSn_RX_MAX = Sn_RXBUF_SIZE * 1024;
    get_offset = Sn_RX_RD & gSn_RX_MASK;

    /* calculate Read Offset Address */
    get_start_address = gSn_RX_BASE + get_offset;
```

```

/* if overflow the upper boundary of SOCKET n RX Buffer */
If( (get_offset + get_size) > gSn_RX_MAX )
{
    /* copy upper_size Bytes of get_start_address to destination_address
    - destination_address is user data memory address */
    upper_size = gSn_RX_MAX - get_offset;
    memcpy(get_start_address, destination_address, upper_size);
    destination_address += upper_size;
    /* copy the remained size Bytes of gSn_RX_BASE to destination_address */
    remained_size = get_size - upper_size;
    memcpy(gSn_RX_BASE, destination_address, remained_size);
}
else
{
    /* copy get_size of get_start_address to destination_address */
    memcpy(get_start_address, destination_address, get_size);
}

/* increase Sn_RX_RD as get_size */
Sn_RX_RD += get_size;

/* set RECV Command */
Sn_CR[RECV] = '1';
while(Sn_CR != 0x00); /* wait until RECV Command is cleared*/
}

```

#### □ Send DATA? / Sending Process

SOCKETn TX Buffer 에 Write 된 Data 를 전송한다.

TX Memory Block 에서 Data 가 Write Offset Address 를 [4.2.3 SOCKET TX/RX Buffer Setting](#) 에서 계산된 gSn\_TX\_BASE, gSn\_TX\_MASK 와 Sn\_TX\_WD 를 이용하여 계산하고, Write Offset Address 부터 Data 를 Write 한다. Data Write 후, 그 Size 만큼 Sn\_TX\_WD 를 증가시키고, Sn\_CR[SEND]='1' 를 수행한다.

다음 DATA 를 전송준비가 완료(Sn\_IR[SENDOK] = '1')될 때까지 다음 “Sending Process”를 수행할 수 없으며, 전송 TIMEOUT(Sn\_IR[TIMEOUT] = '1', 참조 [4.8.2 TCP Retransmission](#))이 발생할 수 있다. Sn\_IR[SENDOK]='1'까지의 시간은 사용된 SOCKET Count, DATA Size 와 Network Traffic 등에 의존적이다.

Write Offset Address 계산시 SOCKET n TX Buffer Block 의 경계주소(n=0,1,2 : gSn\_TX\_BASE ~ gSn+1\_TX\_BASE, n=3 : gS3\_TX\_BASE ~ 0xC000)를 벗어나지 않도록 주의한다.

전송될 Data Size 는 SOCKET n TX Buffer Size 를 초과할 수 없고, MSS 보다 큰 Data 는 MSS 단위로 나뉘어 전송된다.

```
{
    /* calculate SOCKET n TX Buffer Size & Offset Address */
    gSn_TX_MAX = Sn_TXBUF_SIZE * 1024;
    get_offset = Sn_TX_WR & gSn_TX_MASK;

    /* check the Max Size of Data(send_size) & Free Size of SOCKET n TX Buffer(Sn_TX_FSR)
    */
    if( send_size > gSn_TX_MAX ) send_size = gSn_TX_MAX;
    while(send <= Sn_TX_FSR); // wait until SOCKET n TX Buffer is free */
    /* If you don't want to wait TX Buffer Free
    send_size = Sn_TX_FSR; // write Data as Size of Free Buffer
    */

    /* calculate Write Offset Address */
    get_start_address = gSn_TX_BASE + get_offset;

    /* if overflow the upper boundary of SOCKET n TX Buffer */
    If( (get_offset + send_size) > gSn_TX_MAX )
    {
        /* copy upper size bytes of source_address to get_start_address
        - source_address is the start address of user data */
        upper_size = gSn_TX_MAX - get_offset;
        memcpy(source_address, get_start_address, upper_size);

        /* copy the remained size bytes of source_address to gSn_TX_BASE */
        source_address += upper_size;
        remained_size = send_size - upper_size;
        memcpy(source_address, gSn_TX_BASE, remained_size);
    }
    else
    {
        /* copy send_size bytes of source_address to get_start_address
        - source_address is the start address of user data */
        memcpy(source_address, get_start_address, send_size);
    }
}
```

```

}

/* increase Sn_TX_WR as send_size */
Sn_TX_WR += send_size;

/* set SEND Command */
Sn_CR = SEND;
while(Sn_CR != 0x00); /* wait until SEND Command is cleared */

/* wait until SEND Command is completed or Timeout is occurred */
while(Sn_IR[SENDOK] == '0' and Sn_IR[TIMEOUT] = '0');

/* clear SOCKET Interrupt*/
if(Sn_IR[SENDOK] == '1') Sn_IR[SENDOK] = '1';
else
    goto Timeout?;
}

```

#### ☐ Received FIN (Passive Close)

상대방으로부터 연결종료 요청(FIN Packet)을 수신한 경우,

First Method:

```

{
    If(Sn_SR == SOCK_CLOSE_WAIT) goto Disconnecting Process;
}

```

Second Method:

```

{
    If(Sn_IR[DISCON] == '1') goto Disconnecting Process;
}

```

#### ☐ Disconnected (Active Close)

상대방에게 연결종료 요청(FIN Packet)을 전송할 경우

```

{
    /* send FIN Packet */
    Sn_CR[DISCON] = '1';

    while(Sn_CR != 0x00); /* wait until DISCON Command is cleared*/
    goto Disconnecting Process;
}

```

## □ Disconnecting Process

상대방으로부터 FIN Packet 을 수신한 경우(Passive Close), 더 이상 DATA 통신이 필요하지 않다면 상대방에게 FIN Packet 을 전송하고 SOCKET 을 Close 한다.

상대방에게 FIN Packet 을 전송한 경우(Active Close), 상대방의 FIN Packet 수신을 대기하고 수신 후 SOCKET 을 Close 된다. 이 Process 에서 Sn\_IR [TIMEOUT] = '1'이 발생할 수 있다.

```
Passive Close: /* received FIN Packet from Peer */
{
    /* send FIN Packet */
    Sn_CR = DISCON;
    while(Sn_CR != 0x00); /* wait until DISCON Command is cleared*/

    /* wait unit ACK Packet is received */
    while(Sn_IR[DISCON] == '0' and Sn_IR[TIMEOUT] == '0') ;
    if (Sn_IR[DISCON] == '1')
    {
        /* clear Interrupt */
        Sn_IR[DISCON] = '1';
        goto CLOSED;
    }
    else goto Timeout?;
}
```

```
Active Close : /* sent FIN Packet to Peer */
{
    /* wait until FIN Packet is received*/
    while(Sn_IR[DISCON] == '0' and Sn_IR[TIMEOUT] == '0') ;
    if (Sn_IR[DISCON] == '1')
    {
        /* clear Interrupt */
        Sn_IR[DISCON] = '1';
        goto CLOSED;
    }
    else goto Timeout?;
}
```

#### □ Timeout?

TCP 는 SYN/DATA/FIN Packet 전송에 대한 응답(ACK Packet)을 수신하지 못한 경우, 설정된 시간 동안 재전송을 수행한다. 재전송을 실패할 경우 Sn\_IR[TIMEOUT] = '1'이 발생한다.

(참조 [4.8.2 TCP Retransmission](#))

```
{  
    /* check TIMEOUT Interrupt */  
    if(Sn_IR[TIMEOUT] == '1')  
    {  
        /* clear Interrupt */  
        Sn_IR[TIMEOUT] = '1';  
        goto CLOSE;  
    }  
}
```

#### □ CLOSE

SOCKET n 은 Disconnecting Process, Sn\_IR[TIMEOUT]='1', Sn\_CR[CLOSE]='1'에 의해 CLOSE 된다.

```
{  
    /*Wait until SOCKET n is closed*/  
    while(Sn_SR != SOCK_CLOSED);  
}
```

### 4.3.2 TCP Client

Figure 7 는 ‘TCP CLIENT’의 동작흐름을 도식화한다.

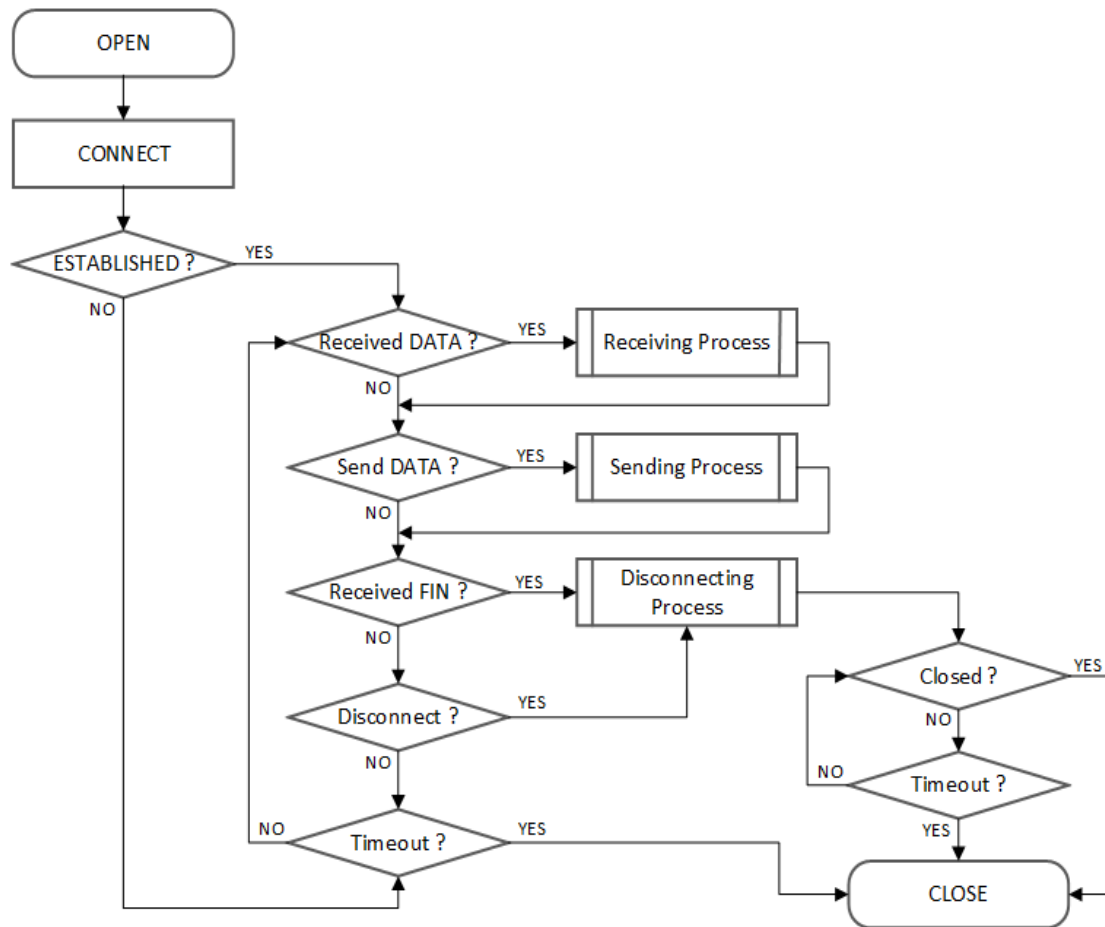


Figure 7 TCP Client Operation Flow

#### ○ OPEN

참조 [4.3.1 TCP Server : OPEN](#)

#### ○ CONNECT

SOCKETn 을 ‘TCP CLIENT’로 동작시킨다.

Sn\_CR[CONNECT] = ‘1’을 통해 ‘TCP SERVER’로 SYN Packet 을 전송한다.

```

{
  /* set destination IP address, 192.168.0.11 */
  Sn_DIPR[0:3] = { 0xC0, 0xA8, 0x00, 0x0B};

  /* set destination PORT number, 5000(0x1388) */
  Sn_DPORTR[0:1] = {0x13, 0x88};

```



```
/* set CONNECT command */  
Sn_CR = CONNECT;  
while(Sn_CR != 0x00); /* wait until command is cleared*/  
goto ESTABLISHED?;  
}
```

#### ○ ESTABLISHED?

‘TCP CLIENT’은 ‘TCP SERVER’로 전송한 SYN Packet 에 대한 SYN/ACK Packet 을 수신 전까지 접속요청상태 (Sn\_SR=SOCK\_SYNSENT)를 유지하며, ‘TCP SERVER’로부터 SYN/ACK Packet 를 수신하면 연결을 완료한다.

(참조 [4.3.1 TCP Server : Received DATA?](#))

#### ○ Others flow

참조 [4.3.1 TCP Server](#)

### 4.3.3 Other Functions

#### 4.3.3.1 TCP SOCKET Options

SOCKET 을 OPEN 하는 과정에서 Sn\_MR 과 Sn\_MR2 를 통해서 SOCKET Option 을 설정한다.

○ **No Delayed ACK : Sn\_MR[NDACK] = '1'**

No Delayed ACK Flag 는 SOCKET 이 TCP 통신 중에 상대방의 Data Packet 에 대한 ACK Packet 을 Delay 없이 즉시 전송하는 기능이다.

○ **Delayed ACK : Sn\_MR[NDACK] = '0'**

RTR 설정 시간 이후 수신한 Data Packet 에 대한 ACK Packet 을 전송하거나, Sn\_CR[RECV] = '1'에 의해 TCP Window Size 가 증가할 경우 ACK Packet 을 전송한다.

○ **Force PSH : Sn\_MR2[UBBLK]='1'**

TCP Force PSH 는 모든 Data Packet 의 PSH flag 를 설정한다.

○ **Auto PSH : Sn\_MR2[UBBLK]='0'**

상대방의 Window Size 가 0 이거나, MSS 단위로 나뉘어 전송된 마지막 Data packet 의 PSH flag 를 설정한다.

#### 4.3.3.2 Keep Alive

Keep Alive (KA) 는 상대방과의 연결이 유효한지 검사하기 위해 마지막 전송 Packet Data 의 마지막 1 Byte 를 전송한다. 따라서 이 기능은 1 Byte 이상의 Data 를 전송한 경우만 사용할 수 있다. KA Packet 전송에 대한 ACK Packet 을 수신하지 못할 경우, 설정된 Retransmission Time 이후 Sn\_IR[TIMEOUT] = '1' 발생한다.

KA Packet 은 설정한 주기(Sn\_KPALVTR > 0)마다 전송하거나, 설정 주기가 없는(Sn\_KPALVTR = 0) 경우 Sn\_CR[SEND\_KEEP] = '1' 로 전송한다.

## 4.4 UDP

UDP(User Datagram Protocol)는 IP Layer 위의 전송계층에 위치하며 신뢰성을 보장하지 않는 Datagram 통신을 하는 프로토콜이다. 또한 Port Number 를 이용해 Application 간 통신을 제공한다. UDP 는 연결과정이 필요 없으며 하나 이상의 상대방과 통신을 할 수 있는 이점이 있는 반면, 데이터 전송에 대한 신뢰성을 보장하지 않으므로 데이터 송수신 과정에서 데이터 손실이나 원하지 않는 상대방으로부터의 데이터 수신이 발생한다. UDP 전송방식은 데이터 송수신 범위에 따라 크게 Unicast, Broadcast, Multicast 로 구분한다.

아래의 Figure 8 는 UDP Mode SOCKET 의 동작흐름을 나타낸다.

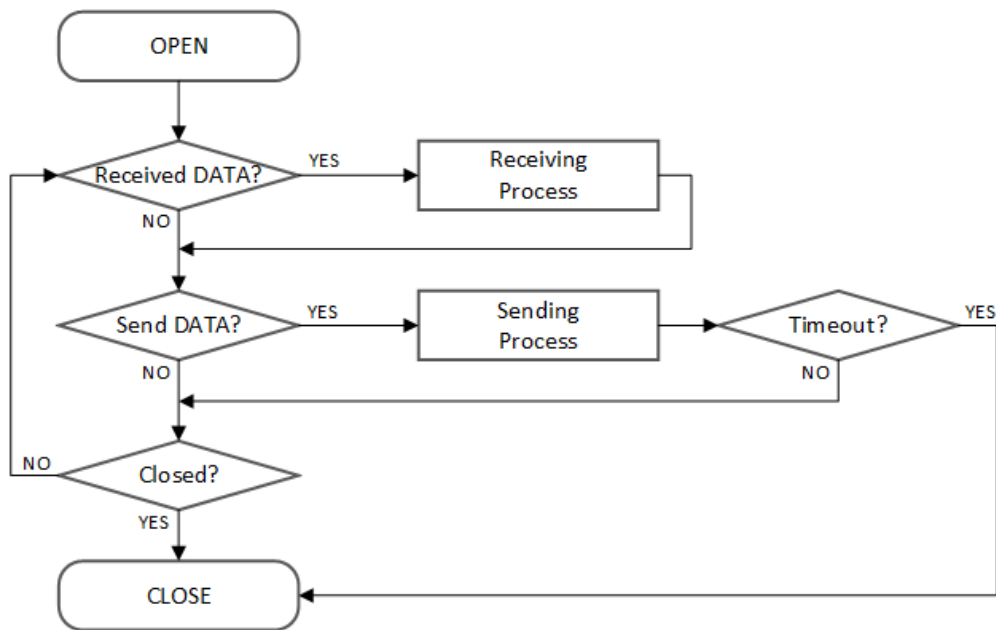


Figure 8 UDP Operation Flow

### 4.4.1 UDP Unicast

UDP Unicast 는 하나의 송신자가 하나의 목적지에 데이터를 전송하는 통신방식이다. 데이터 전송 시 SOCKET 은 먼저 상대방과 ARP 과정을 수행한다. ARP 과정에서 Sn\_IR[TIMEOUT] = '1' (참조 4.8.1 ARP & PING Retransmission)이 발생할 수 있다. 이후 동일한 목적지와 통신을 할 경우 ARP 과정은 생략된다. 또한, UDP Unicast 전송은 Sn\_DHAR 설정 이후 Sn\_CR [SEND\_MAC] = '1'를 수행할 경우 ARP 과정을 생략할 수 있다.

#### ○ OPEN

HOST 는 SOCKET n 을 UDP Mode 로 설정한다.

```

{
START :
  /* set UDP Mode */
  Sn_MR[3:0] = '0010';

```

```

/* set Source PORT Number, 5000(0x1388) */
Sn_PORTR[0:1] = {0x13, 0x88};

/* set SOCKET Option such as Broadcast Block. */
// 참조 3.2.23 Sn_MR2 (SOCKET n Mode register 2)
// Sn_MR2[BRDB] = '1';

/* set OPEN Command */
Sn_CR = OPEN;
while(Sn_CR != 0x00); /* wait until OPEN Command is cleared */

/* check SOCKET for UDP Mode */
if(Sn_SR != SOCK_UDP) goto START;
}

```

#### ○ Received DATA?

참조 [4.3.1 TCP Server : Received DATA?](#)

#### ○ Receiving Process

UDP 는 하나 이상의 목적지로부터 Data Packet 을 수신할 수 있으며, 각 목적지의 구분을 위해 Data Packet 은 Figure 9과 같이 “PACKET INFO”와 함께 SOCKET n RX Buffer 에 저장된다. HOST 는 반드시 Figure 9 단위로 Read 한다. SOCKET n RX Buffer 의 Free Size 보다 큰 Figure 9 UDP Packet 은 Discard 된다.

(참조 [4.3.1 TCP Server : Receiving Process](#))

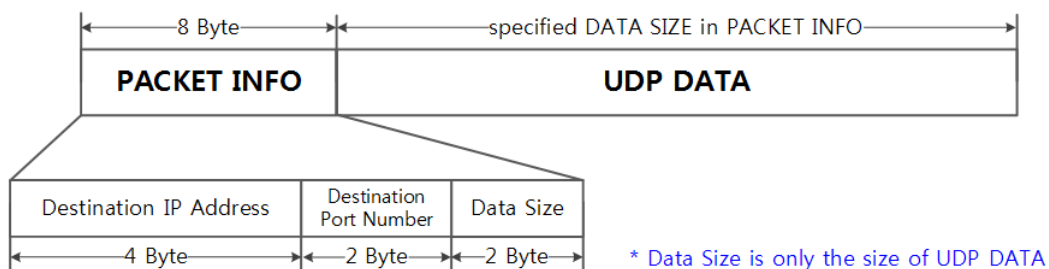


Figure 9 Received UDP DATA in SOCKET n RX Buffer Block

```

{
/* receive PACKINFO */
goto 4.3.1 TCP Server : Receiving Process with get_size = 8;

/* extract Destination IP, Port, Size in PACKET INFO*/
dest_ip[0:3] = destination_address[0:3];

```

```

dest_port = (destination_address[4] << 8) + destination_address[5];
data_size = (destination_address[6] << 8) + destination_address[7];

/* read UDP DATA */
goto 4.3.1 TCP Server : Receiving Process with get_size = data_size;
}

```

#### □ Send DATA? / Sending Process

참조 4.3.1 TCP Server : Send DATA? / Sending Process

```

{
    /* set destination IP address, 192.168.0.11 */
    Sn_DIPR[0:3] = {0xC0, 0xA8, 0x00, 0x0B};

    /* set destination PORT number, 5000(0x1388) */
    Sn_PORTR[0:1] = {0x13, 0x88};

    /* for using SEND_MAC Command : */
    // 참조 4.4.4.1 UDP MAC Send
    /* set destination MAC address, 11:22:33:AA:BB:CC
        Sn_DHAR[0:5] = {0x11, 0x22, 0x33, 0xAA, 0xBB, 0xCC};
    */
    goto 4.3.1 TCP Server : Sending Process;

    /* for using SEND_MAC command : */
    // 참조 4.4.4.1UDP MAC Send
    /*
        goto 4.3.1 TCP Server : Sending Process replaced Sn_CR[SEND] with
        Sn_CR[SEND_MAC];
    */
}

```

#### □ Timeout?

최초 목적지로의 전송이나, 이전 목적지와 다른 목적지로 Data Packet 를 전송할 경우, Data Packet 전송 이전에 ARP 과정을 수행하며, 이 과정에서 Sn\_IR[TIMEOUT] = '1'이 발생할 수 있으며(해당 Data packet 은 Discard 된다. UDP 는 TCP 와 달리 1:N 통신을 지원하므로, Sn\_IR [TIMEOUT] = '1' (참조 4.8.1 ARP & PING Retransmission)이 발생하더라도 SOCKET 은 CLOSE 되지 않는다.

```
{
    /* check TIMEOUT Interrupt */
    if(Sn_IR[TIMEOUT] == '1')
    {
        /* clear TIMEOUT Interrupt */
        Sn_IR[TIMEOUT] = '1';
        goto Received DATA? or Closed? /* or goto Received Data? or goto Closed? */
    }
}
```

#### □ CLOSE

Sn\_CR[CLOSE]에 의해 CLOSE 된다.

```
{
    /* set CLOSE Command */
    Sn_CR = CLOSE;
    while(Sn_CR != 0x00); /* wait until CLOSE Command is cleared*/

    /* wait until SOCKET n is closed */
    while(Sn_SR == SOCK_CLOSED);
}
```

### 4.4.2 UDP Broadcast

Broadcast 는 하나의 송신자가 같은 대역의 다수의 Node 들에게 데이터를 전송하는 통신 방식이다. Broadcast 는 Network 내의 모든 Node 에게 전송하는 방식(All Node Broadcasting) 과 Network 내의 동일 Subnet 을 갖는 Node 에게 전송하는 방식(Subnet Broadcasting) 이 있다.

#### □ Send DATA? / Sending Process

UDP Mode SOCKET 으로 Broadcast 를 사용해 Data 전송 시, Sn\_DIPR 은 같은 대역의 Broadcast Address 로 설정한다.

```
All Node Broadcasting :
{
    /* set broadcast address, 255.255.255.255 */
    Sn_DIPR[0:3] = {0xFF, 0xFF, 0xFF, 0xFF};

    /* set Destination PORT Number, 5000(0x1388) */
    Sn_PORTR[0:1] = {0x13, 0x88};
}
```

```

goto 4.3.1 TCP Server : Sending Process;
}

Subnet Broadcasting : Assume SIPR = "192.168.0.10" & SUBR = "255.255.255.0"
{
    /* set Broadcast Address, 192.168.0.255 */
    Sn_DIPR[0:3] = {0xC0, 0xA8, 0x00, 0xFF};

    /* set Destination PORT Number, 5000(0x1388) */
    Sn_PORTR[0:1] = {0x13, 0x88};

    goto 4.3.1 TCP Server : Sending Process;
}

```

### 4.4.3 UDP Multicast

UDP Multicast 는 하나의 송신자가 하나의 목적지 그룹에 데이터를 전송하는 통신방식이다. Multicast-Group Address 범위는 224.0.0.0 ~ 239.255.255.255 이며 (참조 [IANA Multicast Address](#)) 또한 이에 대응되는 MAC Address 는 01:00:5E:00:00:00 ~ 01:00:5E:FF:FF:FF 이다. Multicast MAC Address 설정 시 하위 23 Bits 는 Multicast-Group Address 와 동일해야 한다. (참조, [rfc1112](#))

#### □ OPEN

Sn\_CR[OPEN] = '1' 이전에 Multicast-Group 정보와 Sn\_MR[MULTI] = '1'를 설정하고, Sn\_CR[OPEN] = '1'을 수행하면 IGMPv1 또는 IGMPv2 를 통해 Multicast Group 에 Join 한다.

```

{
START :
    /* set Multicast-Group MAC Address, 01:00:5E:00:00:64 */
    Sn_DHAR[0:5] = {0x01, 0x00, 0x5E, 0x00, 0x00, 0x64};

    /* set Multicast-Group IP Address, 224.0.0.100 */
    Sn_DIPR[0:3] = {0xE0, 0x00, 0x00, 0x64};

    /* set Multicast-Group PORT Number, 3000(0x0BB8) */
    Sn_DPORTR[0:1] = {0x0B, 0xB8};

    /* set UDP Multicast */
    Sn_MR[MULTI] = '1';
}

```

```
/* set IGMP Version */
Sn_MR[MC] = '1'; /* Sn_MR[MC] = '1' : IGMPv1 , Sn_MR[MC] = '0' : IGMPv2 */

/* set SOCKET option such as Unicast Block or Broadcast Block.
참조 3.2.23 Sn_MR2 (SOCKET n Mode register 2) */
// Sn_MR2[UNIB] = '1';
// Sn_MR2[BRDB] = '1';

/* set UDP Mode */
Sn_MR[3:0] = 4'b0010;

/* set Source PORT Number, 3000(0x0BB8) */
Sn_PORTR[0:1] = {0x0B, 0xB8};

/* set OPEN Command */
Sn_CR = OPEN;

/* check SOCKET for UDP Mode */
if(Sn_SR != SOCK_UDP) goto START;
}
```

#### ○ Send DATA? / Sending Process

참조 4.3.1TCP Server : Sending Process



#### 4.4.4 Other Functions

##### 4.4.4.1 UDP MAC Send

목적지의 MAC 주소를 알고 있는 경우, 그 목적지 MAC 을 Sn\_DHAR 로 설정하고 Sn\_CR [SEND\_MAC] = '1'를 수행하여 ARP 과정 없이 UDP DATA 를 전송할 수 있다.

(참조 [4.4.1 UDP Unicast : Send DATA?/Sending Process](#))

##### 4.4.4.2 UDP SOCKET Options

Unicast UDP SOCKET n (Sn\_MR[MULTI] = '0')은 기본적으로 Unicasting, Broadcasting Packet 수신을 지원하지만, Sn\_MR2[BRDB] = '1' 인 경우 Broadcasting Packet 수신하지 않는다.

Multicast UDP SOCKET n (Sn\_MR[MULTI] = '1')은 기본적으로 Unicasting, Broadcasting, Multicasting Packet 수신을 지원하지만, Sn\_MR2[UNIB] = '1' 인 경우 Unicasting Packet 을 수신하지 않으며, Sn\_MR2[BRDB] = '1' 인 경우 Broadcasting Packet 을 수신하지 않는다.

UDP SOCKET Options 들은 Sn\_CR[OPEN] = '1' 이전에 설정될 수 있다.

Sn_MR[MULTI]	Sn_MR2[BRDB]	Sn_MR2[UNIB]	Unicast	Multicast	Broadcast
0	0	Don't Care	O	X	O
0	1	Don't Care	O	X	X
1	0	0	O	O	O
1	0	1	X	O	O
1	1	0	O	O	X
1	1	1	X	O	X

##### 4.4.4.3 Port Unreachable Block

상대방이 Open 되지 않은 Port 로 UDP Packet 을 전송할 경우, 일반적으로 Destination Port Unreachable Packet 을 전송한다. 이는 Port Scan 공격의 대상이 되며, 이를 방지하기 위해서 Port Unreachable Packet 전송을 Block 할 수 있다.

MR2[UDPURB] = '0'인 경우, 상대방에게 Port Unreachable 를 알려준다.

MR2[UDPURB] = '1'인 경우, 상대방에게 Port Unreachable 를 알려주지 않는다.

## 4.5 IPRAW

IPRAW SOCKET n 는 Sn\_PROTOR 로 설정된 상위 Transport Layer 의 Protocol Number (참조 [IANA Protocol Numbers](#))를 갖는 Internet Protocol(IPv4) Layer 통신을 지원한다. IPv6 의 상위 Protocol 이나, TCP, UDP 는 지원되지 않는다.

Table 5 Internet Protocol Supported In IPRAW Mode

Protocol	Number	Semantic	W5100S Support
ICMP	1	Internet Control Message Protocol	O
IGMP	2	Internet Group Management	O
IPv4	4	IPv4 encapsulation	O
TCP	6	Transmission Control	X
UDP	17	User Datagram	X
IPv6	-	Protocols over IPv6	X
others	-	Other Protocols	O

참고로, W5100S 는 Sn\_PROTOR = ICMP 로 SOCKET n 이 OPEN 되었을 경우, 상대방의 PING-Request 에 대한 Auto PING Reply 를 더 이상 지원하지 않으며, PING-Request Packet 은 IPRAW SOCKET n RX Buffer Block 으로 저장된다.

Figure 10 은 IPRAW SOCKET n 의 동작흐름을 도식화한다.

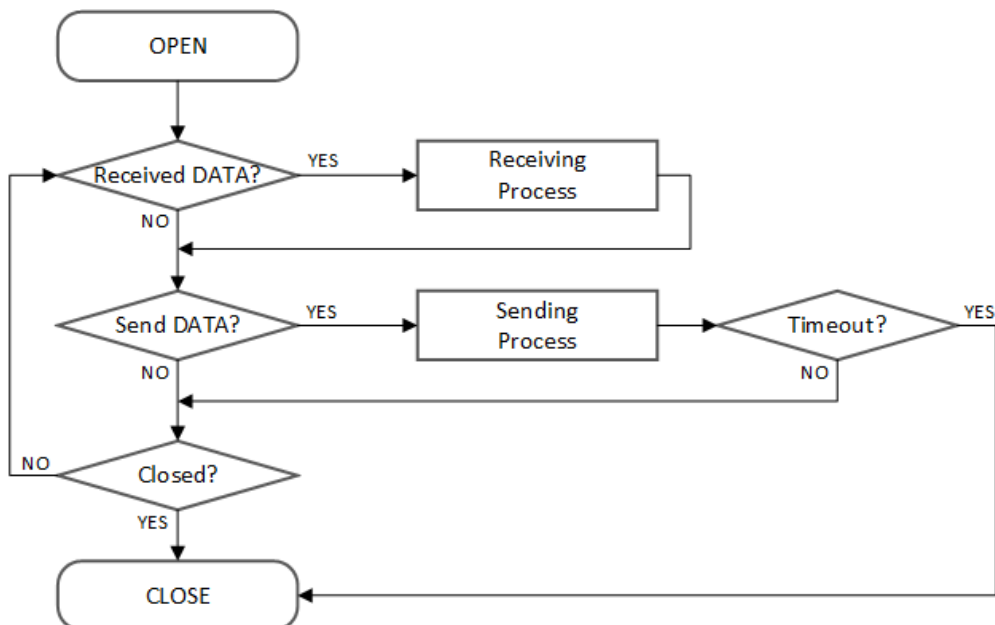


Figure 10 IPRAW Operation Flow

### O OPEN

SOCKET n 을 IPRAW Mode 로 설정한다.

```
{
START :
    /* set Protocol Number */
    Sn_PROTOR = protocol_num;

    /* set IPRAW Mode */
    Sn_MR[3:0] = '0011';

    /* set OPEN Command */
    Sn_CR[OPEN] = '1';
    while(Sn_CR != 0x00); /* wait until OPEN Command is cleared*/

    /* check SOCKET for IPRAW Mode */
    if(Sn_SR != SOCK_IPRAW) goto START;
}
```

#### □ Received DATA?

참조 [4.3.1 TCP Server : Received DATA?](#)

#### □ Receiving Process

IPRAW 는 하나 이상의 목적지로부터 Data Packet 을 수신할 수 있으며, 각 목적지의 구분을 위해 Data Packet 은 Figure 11 과 같이 “PACKET INFO”와 함께 SOCKET n RX Buffer Block 에 저장된다. HOST 는 반드시 Figure 11 단위로 Read 한다. SOCKET n RX Buffer 의 Free Size 보다 큰 Figure 11 IPRAW Packet 은 Discard 된다.

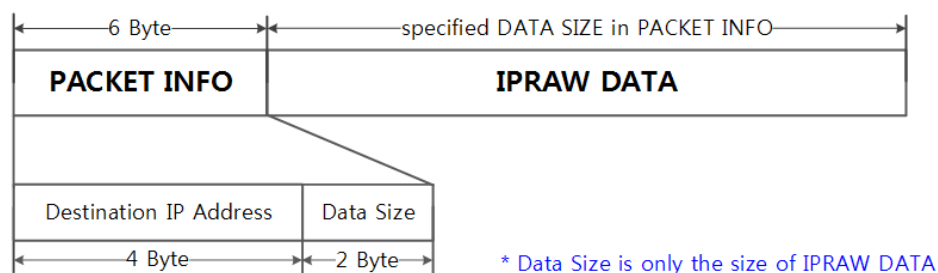


Figure 11 Received Data in IPRAW Mode SOCKET RX Buffer Block

```
{
    /* receive PACKINFO */
    goto 4.3.1TCP Server : Receiving Process with get_size = 6;

    /* extract Destination IP, Size in PACKET INFO*/
    dest_ip[0:3] = destination_address[0:3];
}
```

```

data_size = (destination_address[4] << 8) + destination_address[5];

/* read UDP DATA */
goto 4.3.1TCP Server : Receiving Process with get_size = data_size;
}

```

#### ○ Sending DATA? / Sending Process

전송될 DATA Size 는 SOCKET n TX Buffer Block Size 를 초과할 수 없고, 설정된 MSS 보다 큰 Data 는 MSS 단위(1480)로 직접 나누어 전송한다.

```

{
    /* set Destination IP Address, 192.168.0.11 */
    Sn_DIPR[0:3] = {0xC0, 0xA8, 0x00, 0x0B};

    /* for using SEND_MAC Command :
    참조 4.4.4.1 UDP MAC Send */
    /* set Destination MAC Address, 11:22:33:AA:BB:CC
    Sn_DHAR[0:5] = {0x11, 0x22, 0x33, 0xAA, 0xBB, 0xCC};
    */
    goto 4.3.1TCP Server : Sending Process;

    /* for using SEND_MAC Command :
    참조 4.4.4.1UDP MAC Send */
    /*
    goto 4.3.1 TCP Server : Sending Process replaced Sn_CR[SEND] with
    Sn_CR[SEND_MAC];
    */
}

```

#### □ Timeout?

참조 4.4.1 UDP Unicast : Timeout?

## 4.6 MACRAW

MACRAW Mode 는 Ethernet MAC 을 이용한 Data 통신을 제공하며 오직 SOCKET 0 에서만 사용 가능하다. Sn\_MR[MF] 설정에 따라, Ethernet PHY 로 수신된 모든 Ethernet Packet 을 수신하거나, Destination MAC Address 가 Broadcast, Multicast, Source(SHAR) MAC 인 경우 수신할 수 있다. Broadcast, Multicast Packet 은 IPv6 Packet 과 함께, Sn\_MR2 설정에 따라 선별적으로 수신될 수 있다.

MACRAW SOCKET 0 은 이미 OPEN 된 다른 SOCKET n 이 수신하는 TCP, UDP, IP DATA Packet 은 수신하지 않으나, ARP-Request 와 ICMP PING Request(IPRAW SOCKET n 이 ICMP 를 처리하지 않는 경우만) Packet 들을 수신하며, 이 Packet 에 대한 Auto-Reply Packet 을 전송한다.

Figure 12 은 MACRAW SOCKET 0 의 동작흐름을 도식화한다.

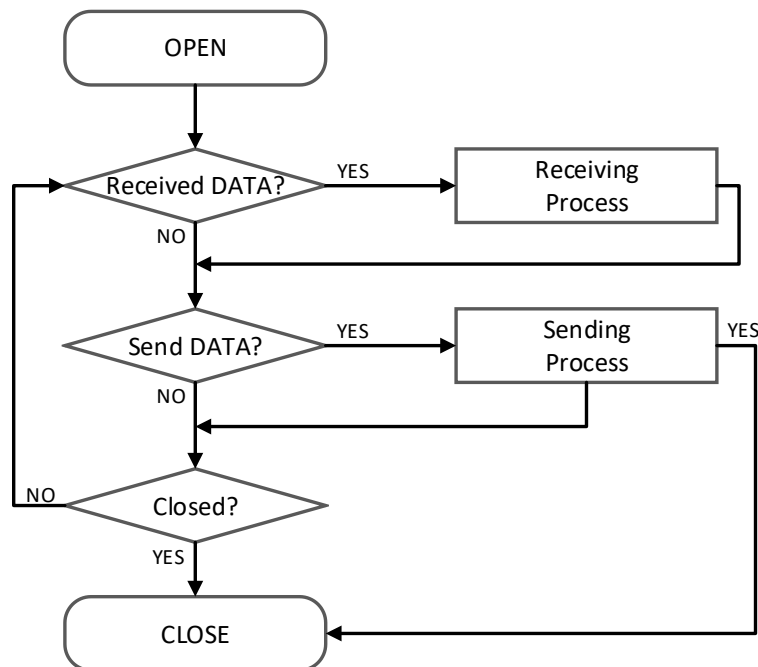


Figure 12 MACRAW Operation Flow

### ○ OPEN

SOCKET 0 을 MACRAW Mode 로 설정한다.

```

{
START :
  /* set MACRAW Mode */
  S0_MR = '0100';

  /* MACRAW SOCKET Options */
  /*
    S0_MR[MR] = '1';      // enable MAC Filter
  */
}
  
```

```

SO_MR2[MBBLK] = '1';    // Broadcast Packet Block
SO_MR2[MMLBK] = '1';    // Multicast Packet Block
SO_MR2[IPV6BLK] = '1';  // IPv6 Packet Block
*/

/* set OPEN Command */
SO_CR = OPEN;
while(Sn_CR != 0x00); /* wait until OPEN Command is cleared*/

/* check SOCKET 0 is MACRAW Mode */
if(SO_SR != SOCK_MACRAW) SO_CR = CLOSE; goto START;
}

```

#### □ Received DATA?

참조 [4.3.1 TCP Server : Received DATA?](#)

#### □ Receiving Process

MACRAW 는 하나 이상의 목적지로부터 Data Packet 을 수신할 수 있으며, 각 목적지의 구분을 위해 DATA Packet 은 Figure 13 과 같이 “PACKET INFO”와 함께 SOCKET 0 RX Buffer Block 에 저장된다. HOST 는 반드시 Figure 13 단위로 Read 한다

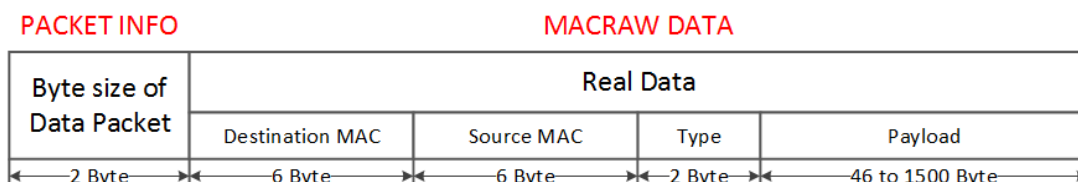


Figure 13 Received DATA Format in MACRAW

```

{
/* receive PACKINFO */
goto 4.3.1TCP Server : Receiving Process with get_size = 2;

/* extract Size in PACKET INFO*/
data_size = (destination_address[0] << 8) + destination_address[1];

/* read UDP Data */
goto 4.3.1TCP Server : Receiving Process with get_size = data_size;
}

```

#### ○ Sending DATA? / Sending Process

전송될 Data Size 는 SOCKET 0 TX Buffer Size 를 초과할 수 없고, MSS 보다 큰 Data 는 MSS 단위(1512)로 직접 나누어 전송해야 한다. 또한 60Byte 보다 작은 경우 Zero Padding 되어 전송된다.

(참조 [4.3.1 TCP Server : Send DATA? / Sending Process](#))

#### ☐ CLOSE

참조 [4.4.1 UDP Unicast : CLOSE](#)

## 4.7 SOCKET-less Command (SLCR)

SLCR 은 SOCKET 없이 ARP Request(SLCR[ARP] = '1'), PING Request(SLCR[PING] = '1') Packet 을 전송을 수행한다. 각 Request 전송에 대한 응답이 없는 경우 SLIR[TIMEOUT] = '1' (참조 4.8.1 ARP & PING Retransmission)이 발생한다.

SLCR[ARP] 와 SLCR[PING]은 SLIR[TIMEOUT] = '1'이 되거나, Command 수행을 완료(SLIR[ARP] = '1' 혹은 SLIR[PING] = '1') 할 때까지 동시에 수행할 수 없다.

Figure 14 는 SOCKET-less Commands 의 동작흐름을 나타낸다.

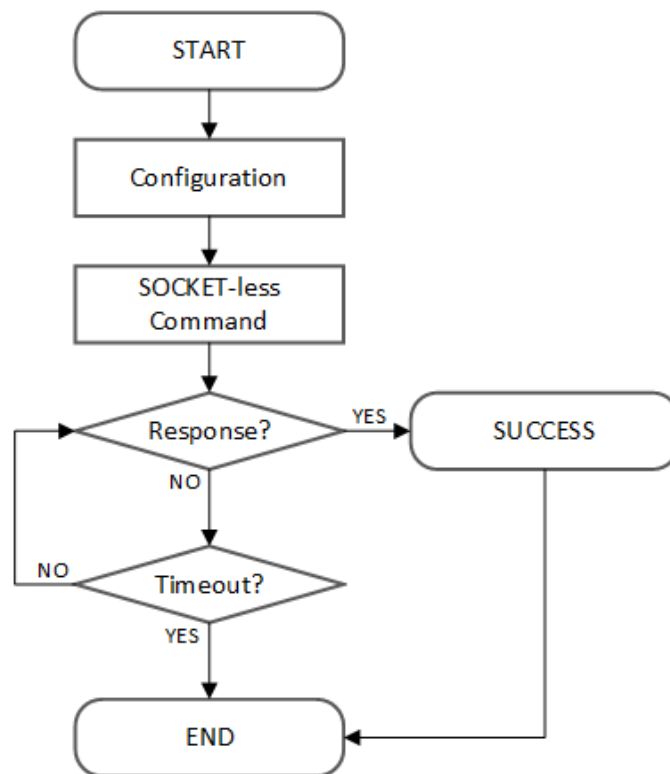


Figure 14 SOCKET-less Command Operation Flow

### 4.7.1 ARP Request (SLCR[ARP] = '1')

SLIPR 로 설정된 목적지로 ARP Request Packet 을 전송한다.

목적지로부터 ARP Reply 를 수신한 경우(SLIR[ARP]= '1') SLPHAR 로 목적지 MAC 주소를 확인할 수 있으며, 그렇지 못한 경우 SLIR[TIMEOUT] = '1' 이 발생한다.

(참조 4.8.1 ARP & PING Retransmission)

#### □ Configuration

SOCKET-less 의 Retransmission Time 과 ARP Interrupt Mask Bit, 상대방의 IP Address 를 설정한다.



```
{
START :

/* set SOCKET-less Retransmission Time, 100ms(0x03E8) (단위, 100us) */
SLRTR[0:1] = {0x03, 0xE8};

/* set SOCKET-less Retransmission Counter, 5 */
SLRCR = 0x05;

/* set Interrupt Mask Bit */
SLIMR[ARP] = '1';          // ARP Interrupt Mask Bit
SLIMR[TIMEOUT] = '1';     // TIMEOUT Interrupt Mask Bit

/* set Destination IP Address, 192.168.0.100 */
SLPIPR[0:3] = {0xC0, 0xA8, 0x00, 0x64};
}
```

#### ☐ SOCKET-less Command

ARP Command 를 통해 ARP Request Packet 을 전송한다.

```
{
/* set ARP Command */
SLCR[ARP] = '1';
while(SLCR != 0x00) ; /* Wait until ARP Command is completed*/
}
```

#### ☐ Response?

상대방으로부터 ARP Reply Packet 을 수신하면 ARP Interrupt 가 발생한다.

```
{
/* check ARP Interrupt */
if(SLIR[ARP] == '1') /* received ARP Reply Packet */
{
/* clear Interrupt */
SLIR[ARP] = '1'; goto SUCCESS;
}
else goto Timeout;
}
```

#### □ Timeout?

설정된 Retransmission Time 동안 상대방으로부터 ARP Reply Packet 을 수신하지 못하면 TIMEOUT Interrupt 가 발생한다.

```
{
    /* check TIMEOUT Interrupt */
    if(SLIR[TIMEOUT] == 1)
    {
        /* clear Interrupt */
        SLIR[TIMEOUT] = '1';
        goto END;
    }
    else goto Response;
}
```

#### □ SUCCESS

ARP Reply Packet 을 수신하면 상대방의 MAC Address 는 SLPHAR 에 저장된다.

```
{
    /* get Destination MAC Address */
    destination_mac[0:5] = SLPHAR[0:5];
    goto END;
}
```

### 4.7.2 PING Command (SLCR[PING] = '1')

SLIPR 로 설정된 목적지로 PING Request Packet 을 전송한다.

목적지로부터 PING Reply 를 수신한 경우(SLIR[PING]= '1') SLPHAR 로 목적지 MAC 주소를 확인할 수 있으며, ARP Reply 나 PING Reply Packet 을 수신하지 못한 경우 SLIR[TIMEOUT] = '1' 이 발생한다. (참조 [4.8.1 ARP & PING Retransmission](#))

Sn\_PROTOR = ICMP 인 IPRAW SOCKET n 이 OPEN 되어 있을 경우 PING Reply Packet 은 해당 IPRAW SOCKET n 의 RX Buffer Block 에 저장되고, SLIR[TIMEOUT] = '1'이 발생할 수 있다.

#### □ Configuration

SOCKET-less 의 Retransmission Time 과 PING Interrupt Mask Bit, 상대방의 IP Address 를 설정한다.

```
{
```

START :

```

/* set SOCKET-less Retransmission Time, 100ms(0x03E8)(단위, 100us) */
SLRTR[0:1] = { 0x03, 0xE8};

/* set SOCKET-less Retransmission counter, 5 */
SLRCR = 0x05;

/* set Interrupt Mask Bit */
SLIMR[PING] = '1';      // PING Interrupt Mask Bit
SLIMR[TIMEOUT] = '1';   // TIMEOUT Interrupt Mask Bit

/* set Destination IP Address, 192.168.0.100 */
SLPIPR[0:3] = {0xC0, 0xA8, 0x00, 0x64};

/* set PING Sequence Number, 1000(0x03E8) */
PINGSEQR[0:1] = {0x03, 0xE8};

/* set PING ID, 256(0x0100) */
PINGIDR[0:1] = {0x01, 0x00};
}

```

#### ☐ SOCKET-less Command

PING Command 를 통해 PING Request Packet 을 전송한다.

```

{
    /* set PING Command */
    SLCR[PING] = '1';
    while(SLCR != 0x00) ; /* Wait until PING Command is completed*/
}

```

#### ☐ Response?

상대방으로부터 PING Reply Packet 을 수신하면 PING Interrupt 가 발생한다.

```

{
    /* check PING Interrupt */
    if(SLIR[PING] == '1') /* received PING Reply Packet */
    {
        /* clear Interrupt */
        SLIR[PING] = '1';
    }
}

```

```
goto SUCCESS;  
}  
else goto Timeout;  
}
```

☐ Timeout? / SUCCESS

참조 [4.7.1 ARP Request \(SLCR\[ARP\] = '1'\) Timeout? / SUCCESS](#)

## 4.8 Retransmission

### 4.8.1 ARP & PING Retransmission

ARP & PING Retransmission 은 해당 Request Packet 전송에 대한 상대방의 응답이 없을 경우 해당 Packet 에 대한 Retransmission 이 발생한다. 이때 Retransmission 은 상대방의 응답을 수신할 때까지 지정된 RTR 시간마다 해당 Packet 을 전송하고, 지정된 RCR 횟수를 초과하여 전송한 경우 해당 Packet 에 대한 TIMEOUT 이 발생한다.

Retransmission TIMEOUT 시간( $ARP_{TO}$ ,  $PING_{TO}$ )은 다음과 같다.

$$ARP_{TO}, PING_{TO} = (TIMEOUT_{VAL} \times 0.1ms) \times (TIMEOUT_{CNT} + 1)$$

$$TIMEOUT_{VAL} = SLRTR \text{ or } Sn\_RTR$$

$$TIMEOUT_{CNT} = SLRCR \text{ or } Sn\_RCR$$

Ex)  $TIMEOUT_{VAL} = 2000(0x07D0)$ ,  $TIMEOUT_{CNT} = 7(0x0007)$

$$ARP_{TO} = 2000 \times 0.1ms \times (7+1) = 1.6s$$

$ARP_{TO}$  은  $SLCR[ARP]$ 이나  $Sn\_CR[SEND]$ ,  $Sn\_CR[CONNECT]$ 에 의해 발생하며,  $SLIR[TIMEOUT]$ 이나  $Sn\_IR[TIMEOUT]$ 으로 확인한다.

$PING_{TO}$ 은  $SLCR[PING]$ 에 의해 발생하며,  $SLCR[TIMEOUT]$ 으로 확인한다.

### 4.8.2 TCP Retransmission

TCP Retransmission 은 SYN, FIN, DATA Packet 등을 전송했을 때 상대방의 ACK Packet 을 수신하지 못한 경우 발생한다. 이때 Retransmission 은 상대방의 ACK 를 수신할 때까지 지정된 RTR 시간마다 해당 Packet 을 전송하고, 지정된 RCR 횟수를 초과하여 전송한 경우 해당 Packet 에 대한 TIMEOUT 이 발생한다.

TCP Retransmission Timeout( $TCP_{TO}$ )은 다음과 같다.

$$TCP_{TO} = \left( \sum_{N=0}^M (TIMEOUT_{VAL} \times 2^N) + ((TIMEOUT_{CNT} - M) \times TIMEOUT_{MAXVAL}) \right) \times 0.1ms$$

N : Retransmission Counter,  $0 \leq N \leq M$

M : Minimum value of  $TIMEOUT_{VAL} \times 2^{(M+1)} > 65535$  and  $0 \leq M \leq TIMEOUT_{CNT}$

$TIMEOUT_{VAL} = SLRTR \text{ or } Sn\_RTR$

$TIMEOUT_{CNT} = SLRCR \text{ or } Sn\_RCR$

$TIMEOUT_{MAXVAL} : TIMEOUT_{VAL} \times 2^M$

---

Ex) RTR = 2000(0x07D0), RCR = 7(0x0007)

$$\begin{aligned} TCP_{TO} &= (0x07D0+0x0FA0+0x1F40+0x3E80+0x7D00+0xFA00+0xFA00+0xFA00) \times 0.1\text{ms} \\ &= (2000 + 4000 + 8000 + 16000 + 32000 + ((7 - 5) \times 64000)) \times 0.1\text{ms} \\ &= 190000 \times 0.1\text{ms} = 19.0\text{s} \end{aligned}$$

$TCP_{TO}$ 은 Sn\_CR 의 CONNECT, SEND, DISCON Command 에 의해 발생하며, Sn\_IR[TIMEOUT]으로 확인한다.

## 4.9 Others Functions

### 4.9.1 System Clock(SYS\_CLK) Switching

SYS\_CLK 은 25MHz 나 100MHz 로 동작 할 수 있으며, MR2[CLKSEL]이나, PHYCR1[RST], PHYCR1[PWDN] 의 설정에 의해 Switching 된다. Clock Switching 이 발생할 경우 SYS\_CLK 이 안정화 될 때까지 대기한다. (참조 7.4.1 Reset Timing)

MR2[CLKSEL]	PHYCR1[RST]	PHYCR1[PWDN]	SYS_CLK(MHz)
0	1	X	25
0	0	0	100 (Default)
0	0	1	25
1	X	X	25

### 4.9.2 Ethernet PHY Operation Mode Configuration

PHYCR0 을 통해 PHY Operation Mode(Speed, Duplex)를 설정하고, Ethernet PHY HW Reset 이후 적용된다. 설정된 PHY Operation Mode 는 PHYSR[5:3]을 통해 확인할 수 있으며, Ethernet PHY Link Up 이후 Link 상태 정보는 PHYSR[2:0]을 통해 확인할 수 있다. PHYCR0 설정 이전에, PHYLCKR 을 Unlock 한다.

Ex) PHY Operation Mode 설정

```

PHY_10FDX :
{
    /* PHYCR0 & PHYCR1 Unlock */
    PHYLCKR = 0x53;

    /* Set PHYCR0 10Mbps/Full-Duplex */
    PHYCR0[DPX] = '0';    // 0 - FDX, 1 - HDX;
    PHYCR0[SPD] = '1';    // 0 - 100Mbps, 1 - 10Mbps;
    PHYCR0[AUTO] = '1';   // 0 - Auto-negotiation, 1 - Manual;

    /* PHY Reset Process */
    PHYCR1[RST] = '1';
    Wait  TPRST;    // refer to 7.4.1 Reset Timing

    /* wait until PHY Link is up */
    while(PHYSR[LINK] != '0');

    /* read PHYSR */
    If ( (PHYSR[DPX] == '0') & (PHYSR[SPD] == '1') & (PHYSR[AUTO] == '1') ) SUCCESS;
    else FAIL;
}

```

```

/* PHYCR0 & PHYCR1 Lock */
PHYLOCKR = 0x00;    // for Lock, write any value
}

```

### 4.9.3 Ethernet PHY Parallel Detection

Link Partner 가 Auto-negotiation 을 지원하지 않을 경우, 내장 Ethernet PHY 는 Parallel Detection 을 통해 Link 를 형성한다. 10F/10H 와 같이 Duplex Mode 가 일치하지 않는 경우 Network 성능 저하의 원인 될 수 있다.

Link Partner PHY	Auto	10H	10F	100H	100F
Auto	100F 100F	10H 10H	10F 10H	100H 100H	100F 100H
Manual 10H	10H 10H	10H 10H	10F 10H		
Manual 10F	10H 10F	10H 10F	10F 10F		
Manual 100H	100H 100H			100H 100H	100F 100F
Manual 100F	100H 100F			100H 100F	100F 100F

### 4.9.4 Ethernet PHY Auto MDIX

Ethernet PHY 를 Auto-negotiation(PHYCR0[AUTO] = '0')로 사용할 경우 Auto-MDIX 를 지원하며, Symmetric Transformer(Figure 30 Transformer Type)를 사용한다.

Auto-negotiation(PHYCR0[AUTO] = '1')를 사용하지 않는 경우는 Auto-MDIX 를 지원하지 않으므로 Cross UTP cable 를 사용한다.

(참고) Link 된 두 Node 간 어느 한 Node 라도 Auto-MDIX 를 지원할 경우, Straight 혹은 Cross UTP Cable 둘 다 사용 가능하다.

### 4.9.5 Ethernet PHY Power Down Mode

PHYCR1[PWDN] = '1' 로 설정할 경우 Ethernet PHY 는 Power Down Mode 로 진입하며, SYS\_CLK 은 25MHz 로 변경된다.

PHYCR1[PWDN] = '1'로 설정할 경우 Ethernet PHY 는 Normal Mode 로 진입하고, SYS\_CLK 은 MR2[CLKSEL] 설정에 따라 결정된다. (참조 3.1.19 MR2 (Mode Register 2))

Enter Power Down mode :



```
{
    /* PHYCR0 & PHYCR1 Unlock */
    PHYLCKR = 0x53;

    /* Enable Power Down Mode */
    PHYCR1[PWDN] = '1';

    /* PHYCR0 & PHYCR1 Lock */
    PHYLCKR = 0x00; // for Lock, write any value

    /* wait until clock is stable switched */
    Wait TPRST; // refer to 7.4.1 Reset Timing
}
```

Exit Power Down mode :

```
{
    /* PHYCR0 & PHYCR1 Unlock */
    PHYLCKR = 0x53;

    /* enable Power Down Mode */
    PHYCR1[PWDN] = '0';

    /* PHYCR0 & PHYCR1 Lock */
    PHYLCKR = 0x00; // for Lock, write any value

    /* wait until Clock is stable switched */
    Wait TPRST; // refer to 7.4.1 Reset Timing

    /* wait until Clock is switched 25 to 100MHz*/
    Wait TLF; // refer to 7.4.1 Reset Timing
}
```

#### 4.9.6 Ethernet PHY's Registers Control

Ethernet PHY 의 Register 들은 일반적으로 MDC/MDIO(Management Data Clock / Input Output) Interface 를 통해 Access 된다. W5100S 는 MDC/MDIO Controller 를 내장하고 있고, 이는 PHYDIVR, PHYRAR, PHYDOR, PHYDIR, PHYACR 로 Control 할 수 있다.

Ex) BMCR Write

```
{
    PHYRAR = 0x00; // BMCR Address 0x00
```

```
PHYDIR = 0x80; // BMCR[15] = '1' , PHY SW Reset

/* write */
PHYACR = 0x01;
while(PHYACR != 0); // wait until MDC/MDIO Control is completed
}
```

Ex) BMSR Read

```
{
    PHYRAR = 0x01; // BMSR Address 0x01

    /* read */
    PHYACR = 0x02;
    while(PHYACR != 0); // wait until MDC/MDIO Control is completed

    if( PHYDOR & 0x0004)
    {
        // LINK UP - BMSR[2] = '1'
    }
}
```

## 5 HOST Interface Mode

### 5.1 SPI Mode

MOD[3:0] = "0000" 일 경우 SPI Mode 를 지원하며, HOST 와 W5100S 는 Figure 15 와 같이 연결 된다.

W5100S 는 Figure 16 와 같이 SPI Mode 0 과 Mode 3 만을 지원하며, MOSI 는 SCLK 의 Rising edge 때 수신(Sampling)되고, MISO 는 Falling edge 일 때 송신(Toggling)된다.

MOSI 와 MISO 는 SCLK 마다 MSB 에서 LSB 까지 순차적으로 송수신된다.

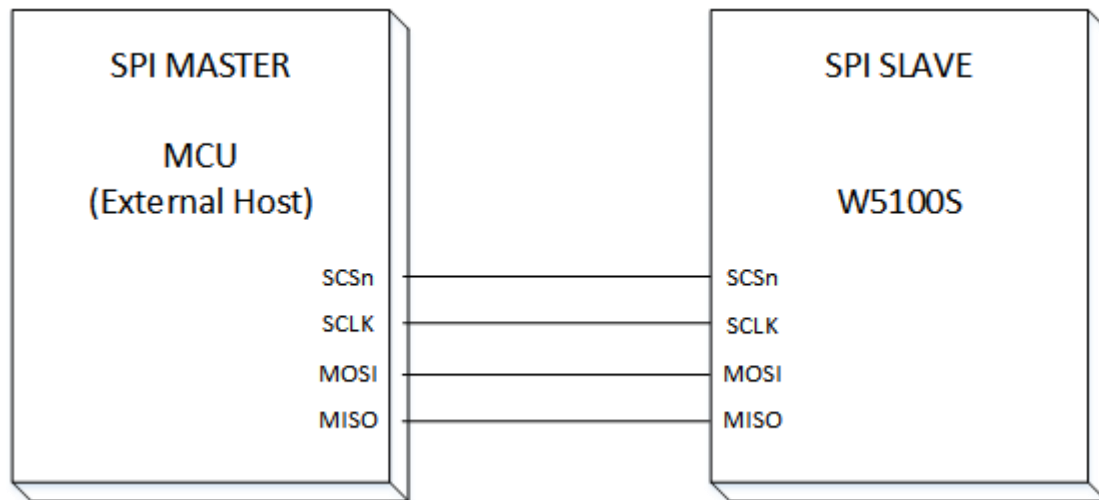


Figure 15 SCSn controlled by Host

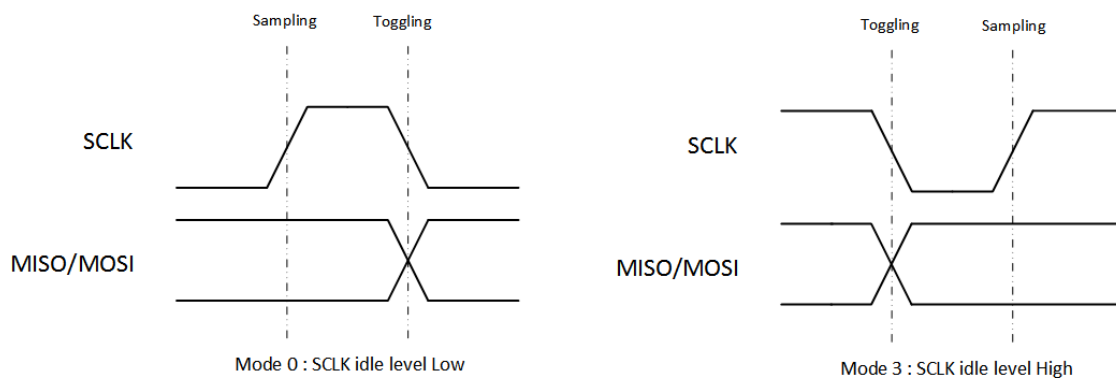


Figure 16 SPI Mode 0 & Mode 3

#### 5.1.1 SPI Frame

W5100S 는 HOST 가 송신하는 SPI Frame(Figure 17 SPI Frame)에 의해 제어된다. SPI Frame 은 Control Phase, Address Phase, Data Phase 로 구성되고, SPI Frame 단위로 CSn 를 제어해야 한다.

W5100 은 1-Byte Data Read/Write 만을 지원하는 반면, W5100S 는 Sequential N-Bytes(N = 1, 2, 3, ...) Data Read/Write 를 지원한다.

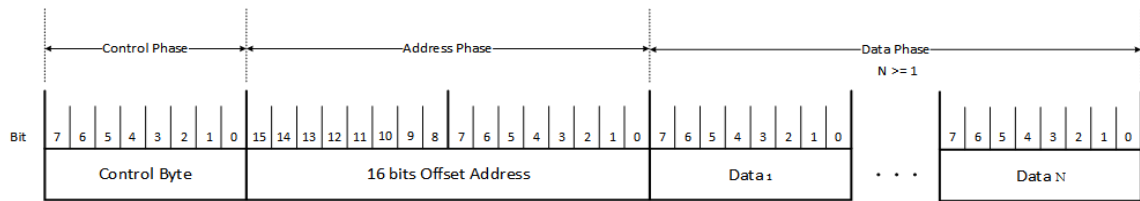


Figure 17 SPI Frame

#### □ Control Phase

Control Phase 는 8 Bits 로 구성되고, Read/Write Access Type 을 설정한다.

HOST 는 Control Phase 송신 2 SYS\_CLK 이전에, CSn 을 High 에서 Low 로 Assert 한다.

Table 6 W5100 Mode SPI Command

Access Type	Value
Write	0xF0
Read	0x0F

#### □ Address Phase

W5100S 의 Registers Block 이나, TX/RX Memory Block 의 16bits Start Offset Address 를 설정한다. Start Offset Address 다음 Data Access 를 위해 자동으로 1 씩 증가한다.

#### □ Data Phase

N-Bytes Data 를 Read/Write 한다. Data Access 를 완료하고 2 SYS\_CLK 이후에 CSn 을 Low 에서 High 로 De-Assert 한다.

## 5.1.2 SPI Write

Figure 18 는 N-Bytes Data Write SPI Frame 을 보여준다.

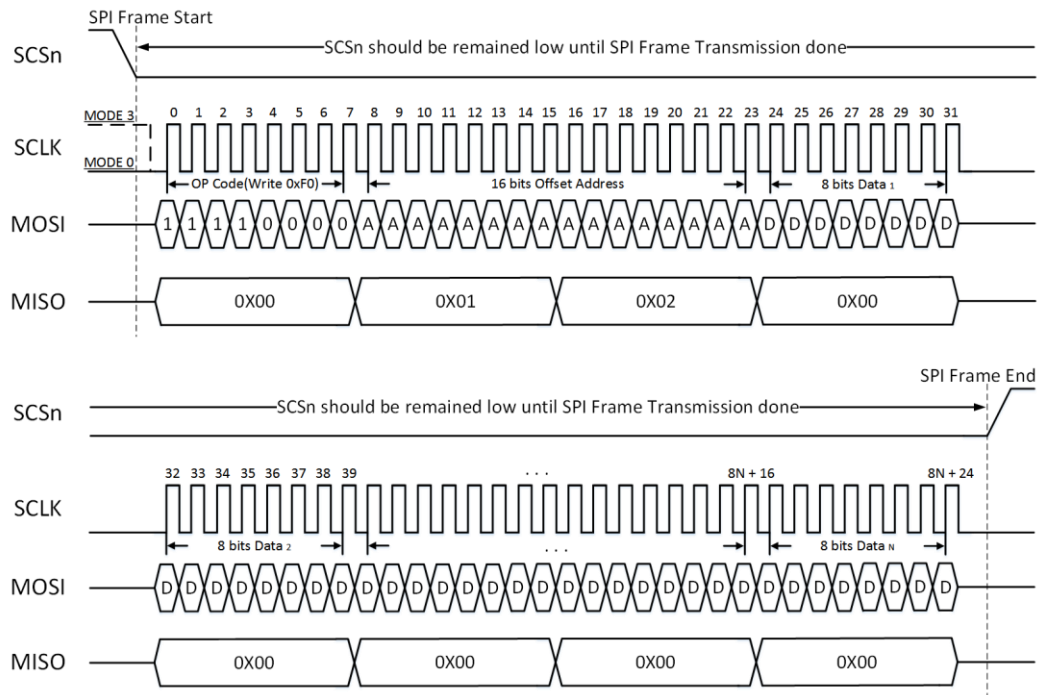


Figure 18 W5100 Mode Write SPI Frame

## 5.1.3 SPI Read

Figure 19 은 N-Bytes Data Read SPI Frame 을 보여준다.

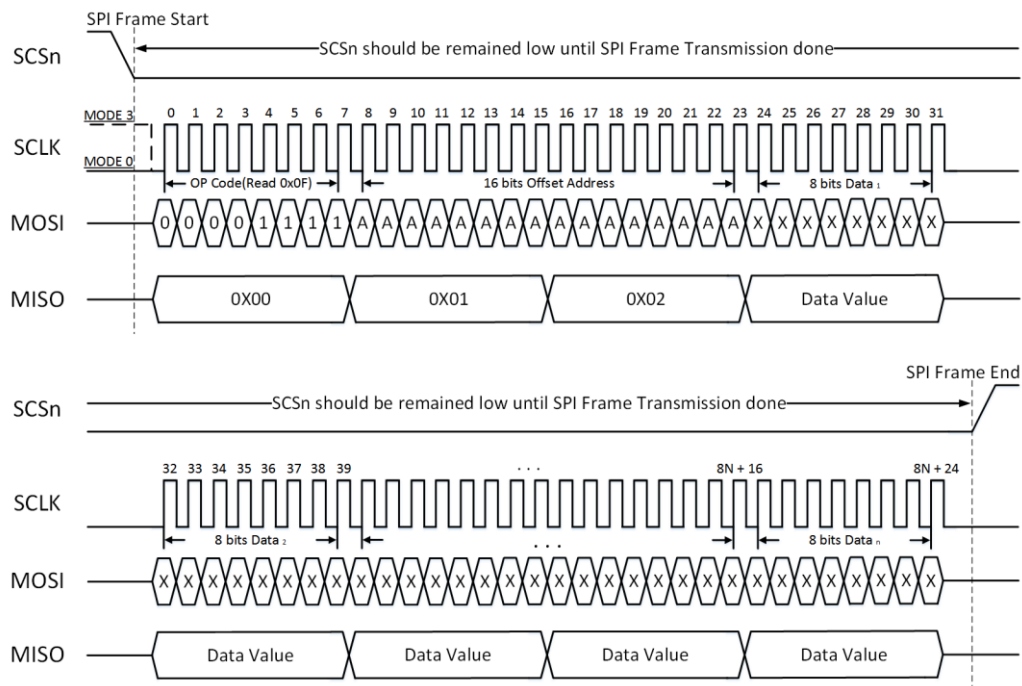


Figure 19 W5100 Mode Read SPI Frame

## 5.2 Parallel Bus Mode

MOD[3:0] = "010X" 일 경우 Parallel Bus Mode 를 지원하며, HOST 와 W5100S 는 Figure 20 과 같이 연결된다.

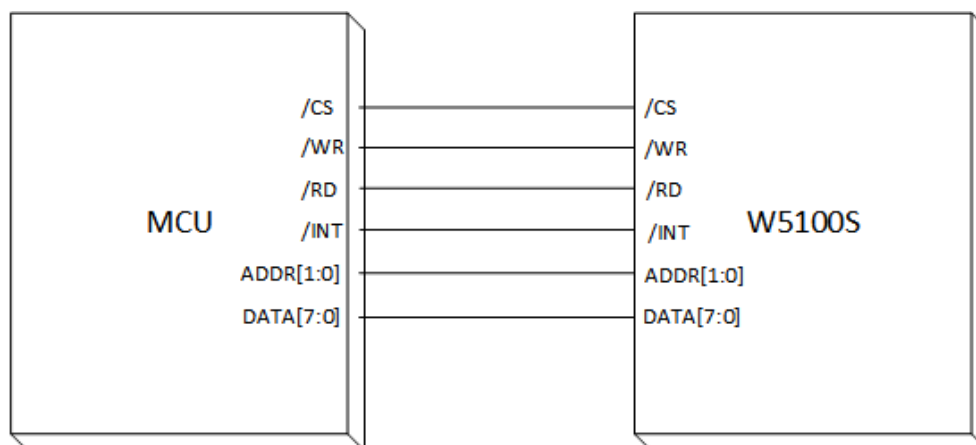


Figure 20 Direct & Indirect Mode Control by Host

Parallel Bus Mode 는 Table 7 의 Register 를 통해 W5100S 의 Common / SOCKET n Register / TX/RX Memory Block 을 Access 하며, N-Bytes Sequential Data Read/Write 를 지원한다.

Table 7 Indirect Mode Address Value

ADDR[1:0]	Symbol	Description
00	MR	Common Register 의 MR(Mode Register) Address
01	IDM_ARH	상위 8 bits Offset Address Register
10	IDM_ARL	하위 8 bits Offset Address Register
11	IDM_DR	8 Bits Data Register

### 5.2.1 Parallel Bus Data Write

Figure 21 은 N-Bytes Data Write 를 보여준다.

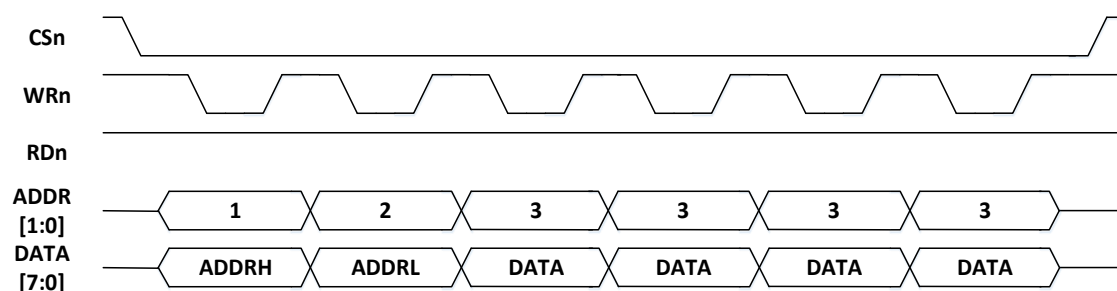


Figure 21 Parallel Bus N-Bytes Data Write

## 5.2.2 Parallel Bus Data Read

Figure 22 은 N-Bytes Data Read 를 보여준다.

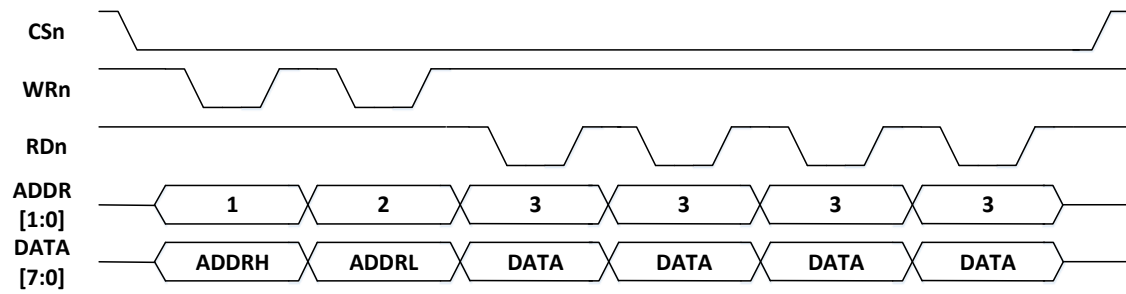


Figure 22 Indirect Mode Continuous Read Access

## 6 Clock & Transformer Requirements

### 6.1 Quartz Crystal requirements.

Table 8 Quartz Crystal

Parameter	Condition / Description	Min	Typ	Max	Unit
Frequency(F)			25		MHz
Frequency Tolerance	At 25°C	-50		+50	ppm
Frequency Stability	1 Year aging.	-50		+50	ppm
Load capacitance( $C_L$ )	ESR = 30 $\Omega$		8		pF
Feedback Resistor( $R_F$ )	External resistor		1M		$\Omega$
Startup time	W5100S Reset			60	ms
Trans-conductance( $g_m$ )			16.7		mA/V
gain margin (gain <sub>margin</sub> )	gain <sub>margin</sub> = $g_m / g_{m_{crit}}$	6.99			dB

$C_o^{(1)}$  : The Packaging Parasitic Shunt Capacitance.

$C_L^{(1)}$  : Load Capacitance. eq)  $C_L = (C_{L1} \times C_{L2}) / (C_{L1} \times C_{L2}) + C_s$

$C_{L1}, C_{L2}$  : External Capacitances of the circuit connected to the crystal (Typically,  $C_{L1} = C_{L2}$ )

$C_s$  : Stray Capacitance of printed circuit board and connections.

$g_{m_{crit}}$  : Oscillator loop critical gain. eq)  $g_{m_{crit}} = 4 \times (ESR + R_{Ext}) \times (2\pi F)^2 \times (C_o + C_L)^2$

$ESR^{(1)}$  : Maximal equivalent series resistance. eq)  $ESR = R_m \times (1 + C_o / C_L)^2$

$R_{Ext}$  : Resistor for limiting the drive level(DL) of the crystal.

$DL^{(1)}$  : The power dissipated in the crystal. Excess power can destroy the crystal.

$R_F^{(2)}$  : Feedback resistor.

- $C_o, C_L, ESR$  and  $DL$  are provided by the crystal manufacturer.
- The W5100S has no feedback resistor. Therefore, it must be inserted outside.

\* Crystal 회로는 아래와 같은 형식으로 모델링된다.

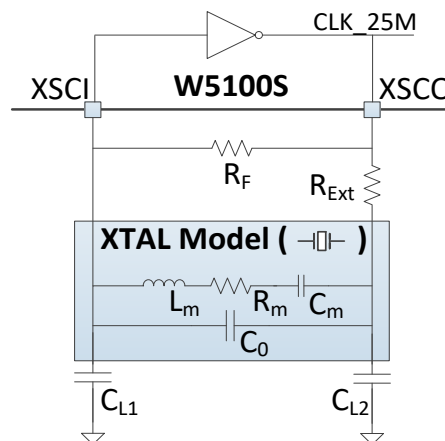


Figure 23 Quartz Crystal Model

Table 9 Crystal Recommendation Characteristics

Parameter	Range
Frequency	25 MHz



Frequency Tolerance (at 25°C)	±30 ppm
Shunt Capacitance	7pF Max
Drive Level	500uW
Load Capacitance	12pF
Aging (at 25°C)	±3ppm / year Max

## 6.2 Oscillator requirements.

Table 10 Oscillator Characteristics

Parameter	Condition / Description	Min	Typ	Max	Unit
Frequency		25			MHz
Frequency Tolerance	At 25°C	-50		+50	ppm
Frequency Stability	1 Year aging. 25°C	-50		+50	ppm
Clock duty	50% of waveform	45	50	55	%
Input high voltage		-	0.97	-	V
Input low voltage		-	0.13	-	V
Rise/Fall Time	10% to 90% of waveform			8ns	
Start up Time		-	-	10ms	
Operating volatage		1.08V	1.2V	1.32V	
Aging (at 25°C)		±3 / year Max			ppm

## 6.3 Transformer Characteristics

Table 11 Transformer Characteristics

Parameter	Transmit End	Receive End
Turn Ratio	1:1	1:1
Inductance	350 uH	350 uH

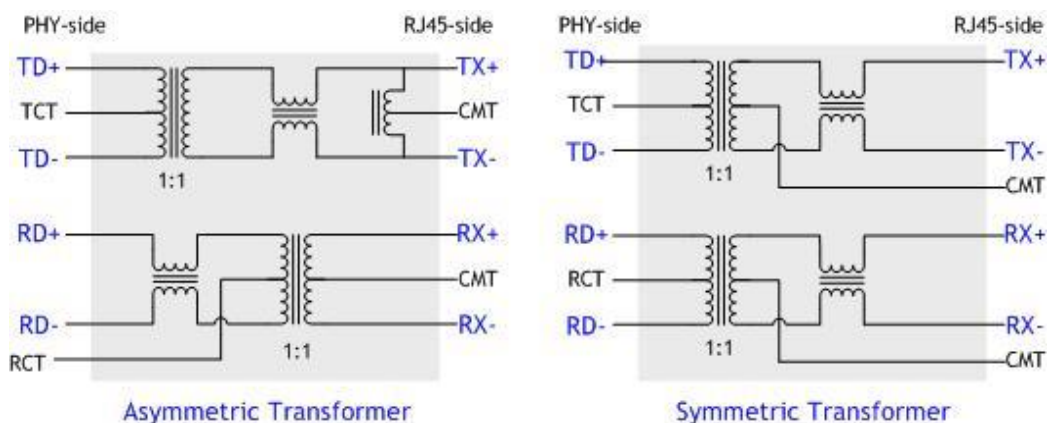


Figure 24 Transformer Type

## 7 Electrical Specification

### 7.1 Absolute Maximum ratings

Table 12 Absolute Maximum ratings

Symbol	Parameter	Rating	Unit
V <sub>DD</sub>	DC Supply voltage	-0.5 to 4.6	V
V <sub>IN</sub>	DC input voltage	-0.5 to 4.6	V
V <sub>OUT</sub>	DC output voltage	-0.5 to 3.63	V
I <sub>IN</sub>	DC input current	20	mA
T <sub>OP</sub>	Operating temperature	-40 to +85	°C
T <sub>JMAX</sub>	Maximum junction temperature	125	°C
T <sub>STG</sub>	Storage temperature	-65 to +150	°C

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

### 7.2 Absolute Maximum ratings (Electrical Sensitivity)

Table 13 Electro Static Discharge (ESD)

Symbol	Parameter	Test Condition	Class	Maximum value(1)	Unit
V <sub>ESD</sub> HBM	Electrostatic discharge voltage (human body model)	TA = +25 °C conforming to MIL-STD 883F Method 3015.7	2	2000	V
V <sub>ESD</sub> MM	Electrostatic discharge voltage (man machine model)	TA = +25 °C conforming to JEDEC EIA/JESD22 A115-A	B	200	V
V <sub>ESD</sub> CDM	Electrostatic discharge voltage (charge device model)	TA = +25 °C conforming to JEDEC JESD22 C101-C	III	500	V

Table 14 Latch up Test

Test Condition	Class	Maximum value	Unit
TA = +25 °C conforming to JESD78	Current	≥ ±100	mA
	Voltage	≥ 1.5*V <sub>DD</sub>	V

## 7.3 DC Characteristics

Table 15 DC Characteristics

(Test Condition: Ta = -40 to 85 °C)

Symbol	Parameter	Test Condition	Min	Typ	Max	Unit
V <sub>DD</sub>	Supply voltage	Apply VDD, AVDD	2.97	3.3	3.63	V
V <sub>IH</sub>	High level input voltage		2.0	-	-	V
V <sub>IL</sub>	Low level input voltage		-		0.8	V
V <sub>T+</sub>	Schmitt trig Low to High Threshold point	All inputs except Analog PINs	0.8	1.1	-	V
V <sub>T-</sub>	Schmitt trig High to Low Threshold point	All inputs except Analog PINs	-	1.6	2.0	V
T <sub>J</sub>	Junction temperature		-40	25	125	°C
I <sub>L</sub>	Input Leakage Current			±1	±10	μA
R <sub>PU</sub>	Pull-up Resistor		40	75	190	Kohm
R <sub>PD</sub>	Pull-down Resistor	RSVD(Pin 23, Pin 38 ~ Pin 42)	30	75	190	Kohm
V <sub>OL</sub>	Low level output voltage	IOL = 2.0mA ~ 8.0mA All outputs except XSCO			0.4	V
V <sub>OH</sub>	High level output voltage	IOH = = 2.0mA ~ 8.0mA, All outputs except XSCO	2.4			V

## 7.4 AC Characteristics

### 7.4.1 Reset Timing

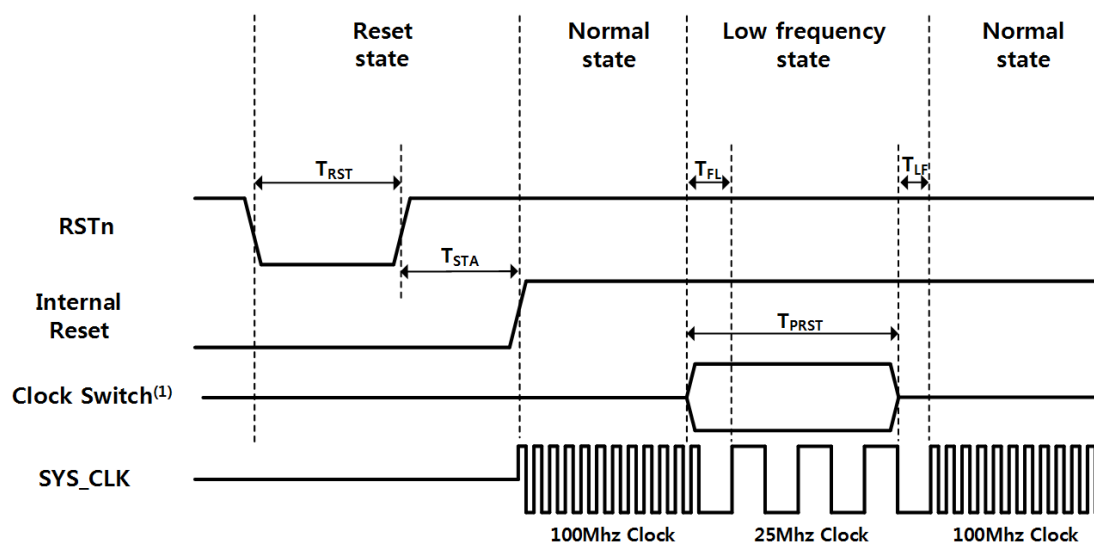


Figure 25 Reset Timing

Table 16 Reset Table

Symbol	Description	Min	Typ	Max
$T_{RST}$	Reset Time	210 ns	330 ns	560 ns
$T_{STA}$	Stable Time	-		60.3 ms
$T_{FI}$	Fast to Low Time by MR2[CLKSEL]	100 ns		-
$T_{FI}$	Fast to Low Time by PHYCR1[Reset] or PHYCR1[PWDN]	300 ns		
$T_{PRST}$	PHY Auto Reset Time	0.6 ms		-
$T_{PRST}$	PHY Power Down Time	200 us		
$T_{PRST}$	Clock Switch Time	200 ns		
$T_{LF}$	Low to Fast Time by MR2[CLKSEL]	100 ns		-
$T_{LF}$	Low to Fast Time by PHYCR1[Reset] or PHYCR1[PWDN]	100 ns		

**\*COMMENT:** PHY Power-down Mode has  $T_{FI}$  and  $T_{LF}$  (In PHY Power-down Mode, SYS\_CLK switches to Low Clock. After  $T_{FI}$ , User can be disable PHY Power-down Mode.)

**\*CAUTION:** User must not set PHY Auto Reset and PHY Power-down Mode at the same time

## 7.4.2 BUS ACCESS TIMING

### 7.4.2.1 READ TIMING

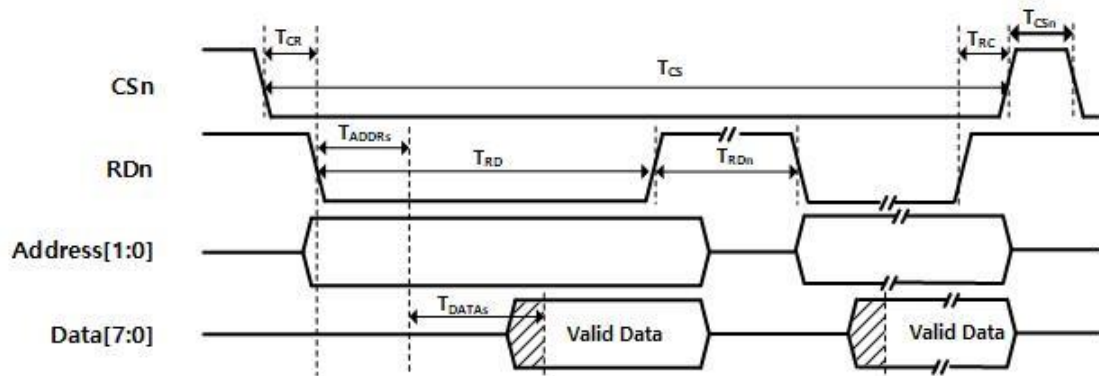


Figure 26 Bus Read Timing

Table 17 BUS Read Timing

Symbol	Description	Min	Max
$T_{ADDRs}$	Address Setup Time	SYS_CLK	
$T_{CR}$	CSn Low to RDn Low Time	0 ns	
$T_{CS}$	CSn Low Time	4 SYS_CLK	
$T_{RC}$	RDn High to CSn High Time	0 ns	
$T_{CSn}$	CSn Next Assert Time	3 SYS_CLK	
$T_{RD}$	RDn Low Time	4 SYS_CLK	
$T_{RDn}$	RDn Next Assert Time	3 SYS_CLK	
$T_{DATAs}$	DATA Setup Time		3 SYS_CLK+5ns

### 7.4.2.2 WRITE TIMING

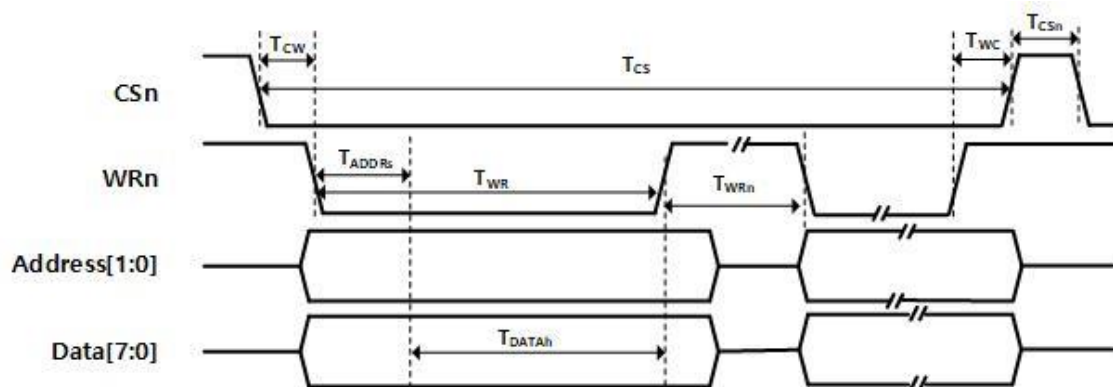


Figure 27 BUS Write Timing

Table 18 BUS Write timing

Symbol	Description	Min	Max
$T_{ADDRs}$	Address Setup Time	SYS_CLK	
$T_{CW}$	CSn Low to WRn Low Time	0 ns	
$T_{CS}$	CSn Low Time	4 SYS_CLK	
$T_{WC}$	WRn High to CSn High Time	0 ns	
$T_{CSn}$	CSn Next Assert Time	3 SYS_CLK	
$T_{WR}$	WRn Low Time	4 SYS_CLK	
$T_{WRn}$	WRn Next Assert Time	3 SYS_CLK	
$T_{DATAs}$	DATA Setup Time	2 SYS_CLK	

## 7.4.3 SPI ACCESS TIMING

### 7.4.3.1 SPI READ TIMING

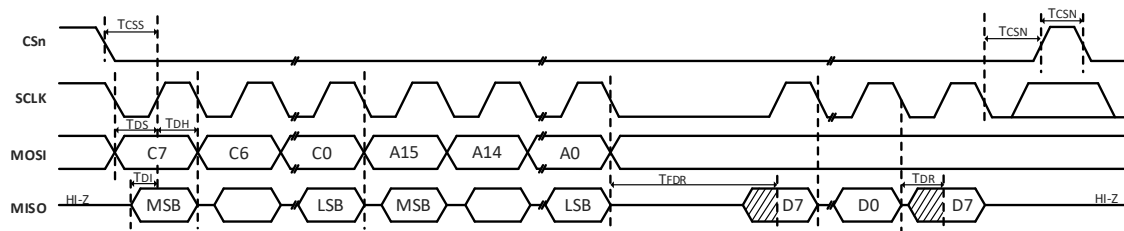


Figure 28 SPI Read Timing

Table 19 SPI Read Timing

Symbol	Description	Min	Max	Units
$F_{SCK}$	SCLK Clock Frequency		70/33.3 <sup>1</sup>	MHz
$T_{CSS}$	SCSn Setup Time	3 SYS_CLK		ns
$T_{CSN}$	SCSn Next Time	2 SYS_CLK		ns
$T_{DS}$	Data In Setup Time	3		ns
$T_{DH}$	Data In Hold Time	3		ns
$T_{DI}$	Data Invalid Time	7		ns
$T_{FDR}^2$	1 <sup>st</sup> Data Ready Time	6 SYS_CLK + 30		ns
$T_{DR}^3$	Data Ready Time	3 SYS_CLK		ns

<sup>1</sup> Theoretical Guaranteed Speed

Even though theoretical design speed is 70MHz, the signal in the high speed may be distorted because of the circuit crosstalk and the length of the signal line. The typical speed of the SCLK is 33.3 MHz which was tested and measured with the stable waveform.

<sup>2</sup> 1<sup>st</sup> Data Ready Time

1<sup>st</sup> Data phase in each SPI read process has  $T_{FDR}$  (Data ready time).  $T_{FDR}$  includes  $T_{DR}$  (Data Ready Time).

<sup>3</sup> Data Ready Time

$T_{DR}$  (Data Ready Time) must be respected for the validity of the next Byte Data.

### 7.4.3.2 SPI WRITE TIMING

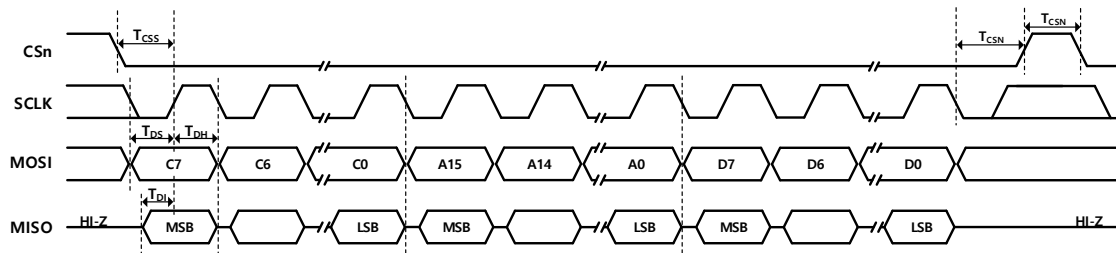


Figure 29 SPI Write Timing

Table 20 SPI Write Timing

Symbol	Description	Min	Max	Units
$F_{SCK}$	SCLK Clock Frequency		70/33.3	MHz
$T_{CSS}$	SCSn Setup Time	3 SYS_CLK		ns
$T_{CSN}$	SCSn Next Time	2 SYS_CLK		ns
$T_{DS}$	Data In Setup Time	3		ns
$T_{DH}$	Data In Hold Time	3		ns
$T_{DI}$	Data Invalid Time	7		ns

### 7.4.4 Transformer Characteristics

Table 21 Transformer Characteristics

Parameter	Transmit End	Receive End
Turn Ratio	1:1	1:1
Inductance	350 uH	350 uH

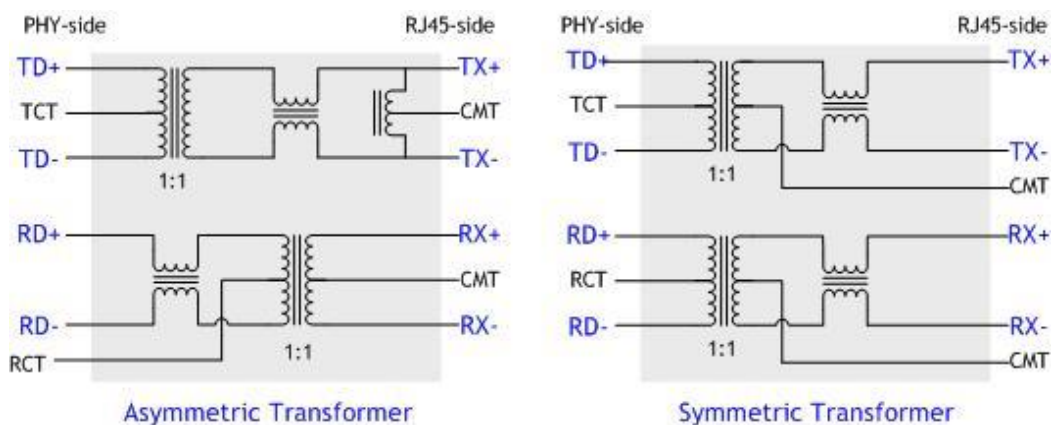


Figure 30 Transformer Type

### 7.4.5 MDIX

W5100S 는 Auto-Negotiation Mode 일 때, Auto-MDIX 를 지원한다.

## 7.5 POWER DISSIPATION

Table 22 Power Dissipation

(Test Condition: VDD=3.3V, AVDD=3.3V, Ta = 25° C)

Condition	Min	Typ	Max	Unit
100M Link	-	93	110	mA
10M Link	-	93	210	mA
100M Unlink(실측)	-	40	170	mA
10M Unlink(실측)	-	25	150	mA
Un-Link (Auto-negotiation mode)(실측)	-	43	170	mA
Power Down mode	-	10	20	mA



## 8 Package Information

### 8.1 LQFP48

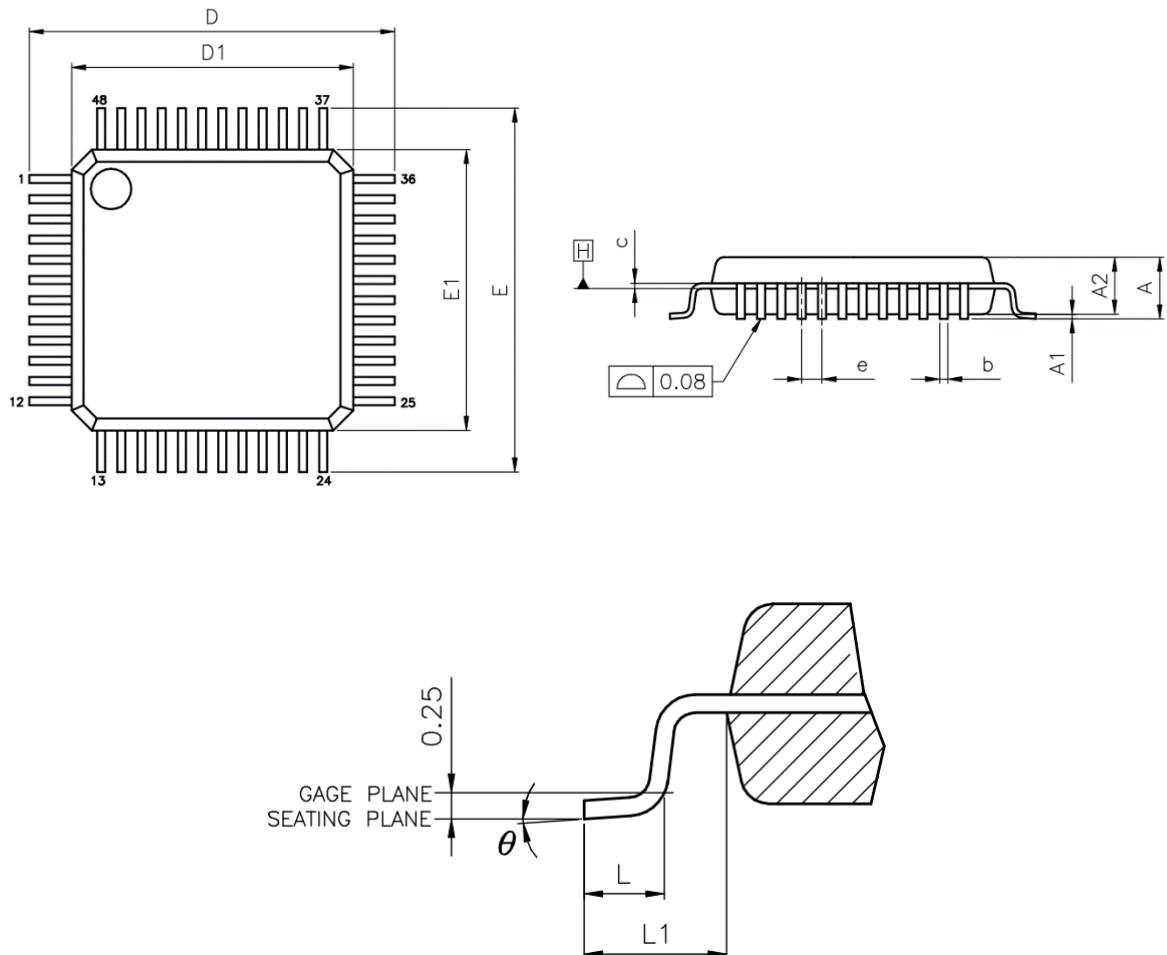


Table 23 LQFP48 VARIATIONS (ALL DEMINSIONS SHOWN IN MM)

SYMBOL	MIN	NOM	MAX
A	--	--	1.60
A1	0.05	--	0.15
A2	1.35	1.40	1.45
b	0.17	0.22	0.27
c	0.09	--	0.20
D	9.00 BSC		
D1	7.00 BSC		
E	9.00 BSC		
E1	7.00 BSC		
e	0.50 BSC		
L	0.45	0.60	0.75

L1	1.00 REF		
$\theta$	0°	3.5°	7°

#### NOTES:

1. JEDEC OUTLINE:  
MS-026 BBC  
MS-026 BBC-HD (THERMALLY ENHANCED VARIATIONS ONLY)
2. DATUM PLANE  $\square$  IS LOCATED AT THE BOTTOM OF THE MOLD PARTING LINE COINCIDENT WITH WHERE THE LEAD EXITS THE BODY
3. DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 mm PER SIDE. DIMENSIONS D1 AND E1 DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE  $\square$ .
4. DIMENSION b DOES NOT INCLUDE DAMBAR PROTRUSION.

## 8.2 QFN48

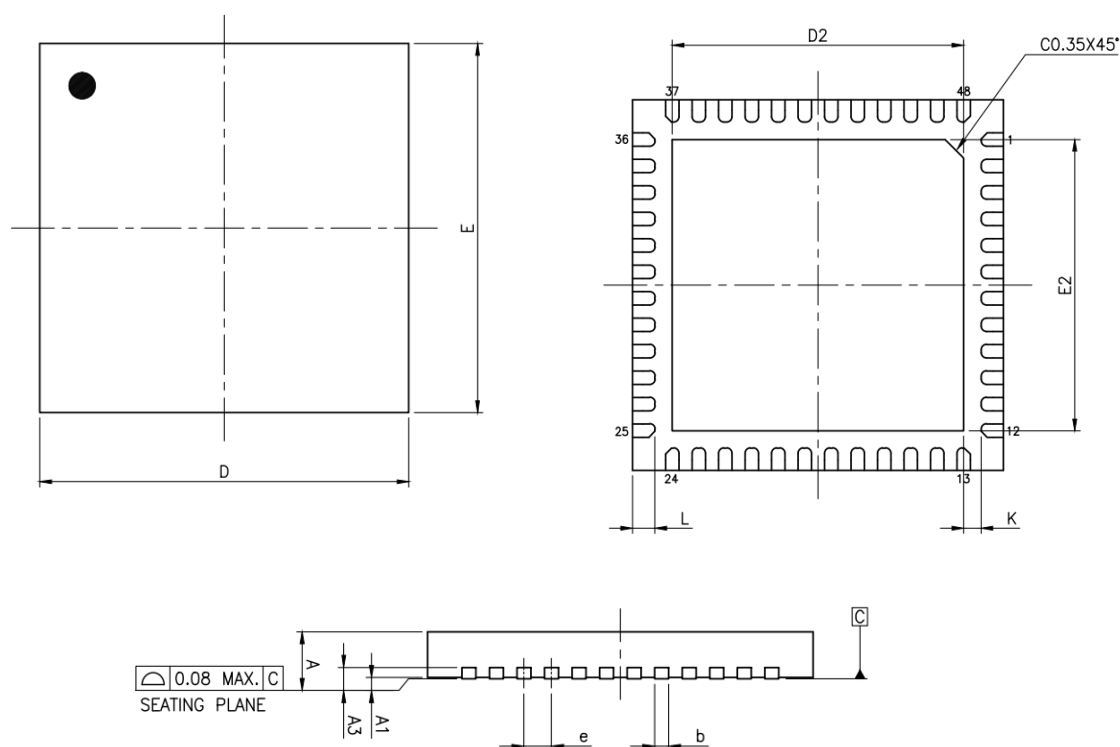


Table 24 QFN48 VARIATIONS (ALL DEMINSIONS SHOWN IN MM)

SYMBOL	MIN	NOM	MAX
A	0.70	0.75	0.80
A1	0.00	0.02	0.05
A3	0.203 REF		
b	0.20	0.25	0.30
D	7.00 BSC		
E	7.00 BSC		
e	0.50 BSC		

D2	5.25	5.30	5.35	
E2	5.25	5.30	5.35	
L	0.35	0.40	0.45	
K	0.20	--	--	
LEAD FINISH	Pure Tin	V	PPF	X
JEDEC CODE	N/A			

**NOTES:**

1. ALL DIMENSIONS ARE IN MILLIMETERS
2. DEMENSION B APPLIES TO METALLIZED TERMINAL AND IS MEASURED BETWEEN 0.15mm AND 0.30mm FROM THE TERMINAL TIP. IF THE TERMINAL HAS THE OPTIONAL RADIUS ON THE OTHER END OF THE TERMINAL, THE DIMENSION b SHOULD NOT BE MEASURED IN THAT RADIUS AREA.
3. BILATERAL COPLANARITY ZONE APPLIES TO THE EXPOSED HEAT SINK SLUG AS WELL AS THE TERMINALS.

## 9 Document Revision History

Version	Date	Descriptions
Ver. 1.0.0	1APR2018	Initial Release
Ver. 1.0.1	2ARP2018	1. 오타 및 잘못된 링크 수정
Ver. 1.0.2	5ARP2018	1. Retry -> Retransmission (in <a href="#">3.1.33SLRTR (SOCKET-less Retransmission Time Register)</a> )
Ver. 1.1.0	5SEP2018	1. RST 동작 (Active Low -> High) ( <a href="#">3.1.31 PHYCR1 (PHY Control Register 1)</a> ) 2. RST 완료 시, PHYSR1[ACT]='1'을 통해 확인 ( <a href="#">3.1.31 PHYCR1 (PHY Control Register 1)</a> ) 3. WOL(Wake On LAN) 비트 추가 ( <a href="#">3.1.31 PHYCR1 (PHY Control Register 1)</a> ) 4. RCR 설명 중 Sn_RCR에 대한 설명 추가 ( <a href="#">3.1.10 RCR (Retransmission Count Register)</a> ) 5. PHYSR1 레지스터 추가 ( <a href="#">3.1.24 PHYSR1 (PHY Status Register 1)</a> ) 6. UDP DATA Size 표기 변경(0 to 1420 -> DATA Size in Packet Info) ( <a href="#">Figure 9</a> ) 7. IPRAW DATA Size 표기 변경(0 to 1420 -> DATA Size in Packet Info) ( <a href="#">Figure 11</a> ) 8. NETLCKR에서 Lock(0x3A -> 0xC5), Unlock(0xC5 -> 0x3A) 값 수정 ( <a href="#">3.1.42 NETLCKR (Network Lock Register)</a> ) 9. INTPTMR 레지스터 타입 변경 (RW -> R=W) ( <a href="#">3.1.6 INTPTMR (Interrupt Pending Time Register)</a> ) 10. RSTn의 Reset Timing 정보 수정 (50ns -> 560ns) ( <a href="#">1.1 PIN Description</a> ) 11. 그 외의 오타 및 링크 수정
Ver. 1.2.0	13NOV2018	1. BUS access timing 변경 (in <a href="#">7.4.2 BUS ACCESS TIMING</a> ) TWR <sub>n</sub> , TRD <sub>n</sub> 추가
Ver. 1.2.1	14NOV2018	1. BUS access timing 변경 (in <a href="#">7.4.2 BUS ACCESS TIMING</a> ) Burst Mode로 변경
Ver. 1.2.2	23NOV2018	1. Sn_MSS reset value, RW 변경 (in <a href="#">3.2.9 Sn_MSS (SOCKET n</a>

		<p><a href="#">Maximum Segment Size Register</a>)</p> <p>2. Modified Load capacitance value(12p -&gt; 8p) (in <a href="#">6.1 Quartz Crystal requirements.</a>)</p>
Ver. 1.2.3	7JAN2019	1. Modified DC Characteristics (in <a href="#">7.3 DC Characteristics</a> )
Ver. 1.2.4	7MAR2019	<p>1. Modified Trans-conductance(<math>g_m</math>)(8.43 -&gt; 16.7) (in <a href="#">6.1 Quartz Crystal requirements.</a></p> <p>2. Modified pin name of SPI/BUS Timing figure (in <a href="#">7.4.2 BUS ACCESS TIMING</a> , <a href="#">7.4.3 7.4.3SPI ACCESS TIMING</a> )</p> <p>3. Edit typo PHYCR0[RST] -&gt; PHYCR1[RST] Modify PHYCR1[RST] value (in <a href="#">4.9.1 System Clock(SYS_CLK) Switching</a>)</p> <p>4. Modified Power Dissipation (in <a href="#">7.5 POWER DISSIPATION</a>)</p>
Ver. 1.2.5	15MAY2019	1. Added Maximum junction temperature (in <a href="#">7.1 Absolute Maximum ratings</a> )
Ver. 1.2.6	21FEB2022	1. Modified register description ( <a href="#">3.1.41 CLKLOCKR (Clock Lock Register)</a> , <a href="#">3.1.42 NETLOCKR (Network Lock Register)</a> , <a href="#">3.1.43 PHYLOCKR (PHY Lock Register)</a> )
Ver. 1.2.7	17MAY2022	1. Corrected RCR reset value '0x08 -> 0x07' (in <a href="#">3.1.10 RCR (Retransmission Count Register)</a> ) and related examples ( <a href="#">4.8.1 ARP &amp; PING Retransmission</a> , <a href="#">4.8.2 TCP Retransmission</a> )
Ver. 1.2.8	16MAY2023	<p>1. Added 1<sup>st</sup> Data Ready Time and corrected Data Ready Time in SPI Read Timing (<a href="#">7.4.3.1SPI READ TIMING</a>)</p> <p>2. Corrected Max SPI SCLK Clock Frequency on typical performance (<a href="#">7.4.3.1SPI READ TIMING</a>) (<a href="#">7.4.2.2WRITE TIMING</a>)</p>

## Copyright Notice

Copyright 2018 WIZnet Co., Ltd. All Rights Reserved.

Technical Support: <https://forum.wiznet.io/>

Sales & Distribution: <mailto:sales@wiznet.io>

For more information, visit our website at <https://www.wiznet.io/>