

# W6100

## Hardwired Dual TCP/IP Stack Controller

V1.0.4



## W6100

W6100 은 WIZnet 의 Hardware TCP/IP 기술 기반 위에 새롭게 IPv6 를 추가하여 IPv4/IPv6 Dual Stack 을 지원하는 Hardwired Internet Controller 로, TCP, UDP, IPv6, IPv4, ICMPv6, ICMPv4, IGMP, ARP, PPPoE 등의 TCP/IP 뿐만 아니라 10 Base-T / 10 Base-Te / 100 Base-TX Ethernet PHY 와 Ethernet MAC Controller 를 모두 내장하여 IPv6 IoT Device 시장에 적합한 Embedded Internet One chip Controller 이다.

W6100 은 8 개의 독립적인 Hardware SOCKET 뿐만 아니라 다양한 SOCKET-less Command 를 지원하여 IPv6 Auto-configuration 을 손쉽게 구현할 수 있으며, ARP, PINGv4, PINGv6 을 통해 Network 상황을 관리할 수 있다.

W6100 은 Target HOST Interface 로 SPI 또는 Parallel System BUS 를 제공하고 송수신 Data 용으로 32KB 메모리를 가지고 있으며, 저전력/저발열로 설계되어 WOL(Wake On LAN)과 Ethernet PHY Power Down Mode 등을 제공한다.

또한 W6100 은 48 LQFP 와 48 QFN Lead-Free Package 를 제공하고 W5100S 와 완벽한 PIN-to-PIN 호환성을 제공한다.

## Features

- Support Hardwired TCP/IP Protocols  
: TCP, UDP, IPv6, IPv4, ICMPv6, ICMPv4, IGMP, MLDv1, ARP, PPPoE
- Support IPv4/IPv6 Dual Stack
- Support 8 independent SOCKETS simultaneously with 32KB Memory
- Support SOCKET-less Command  
: ARP, ICMPv6 (ARP, DAD, NA, RS) Command for IPv6 Auto-configuration & Network Monitoring (PING, PING6)
- Support Ethernet PHY Power Down Mode & System 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 32Kbytes Memory for TX/ RX Buffers
- 10BaseT/ 10BaseTe / 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 & Jumbo packet
- 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)
- W5100S PIN-to-PIN Compatible

## Target Applications

W6100 은 다음과 같은 Embedded application 에 적합하다.

- 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

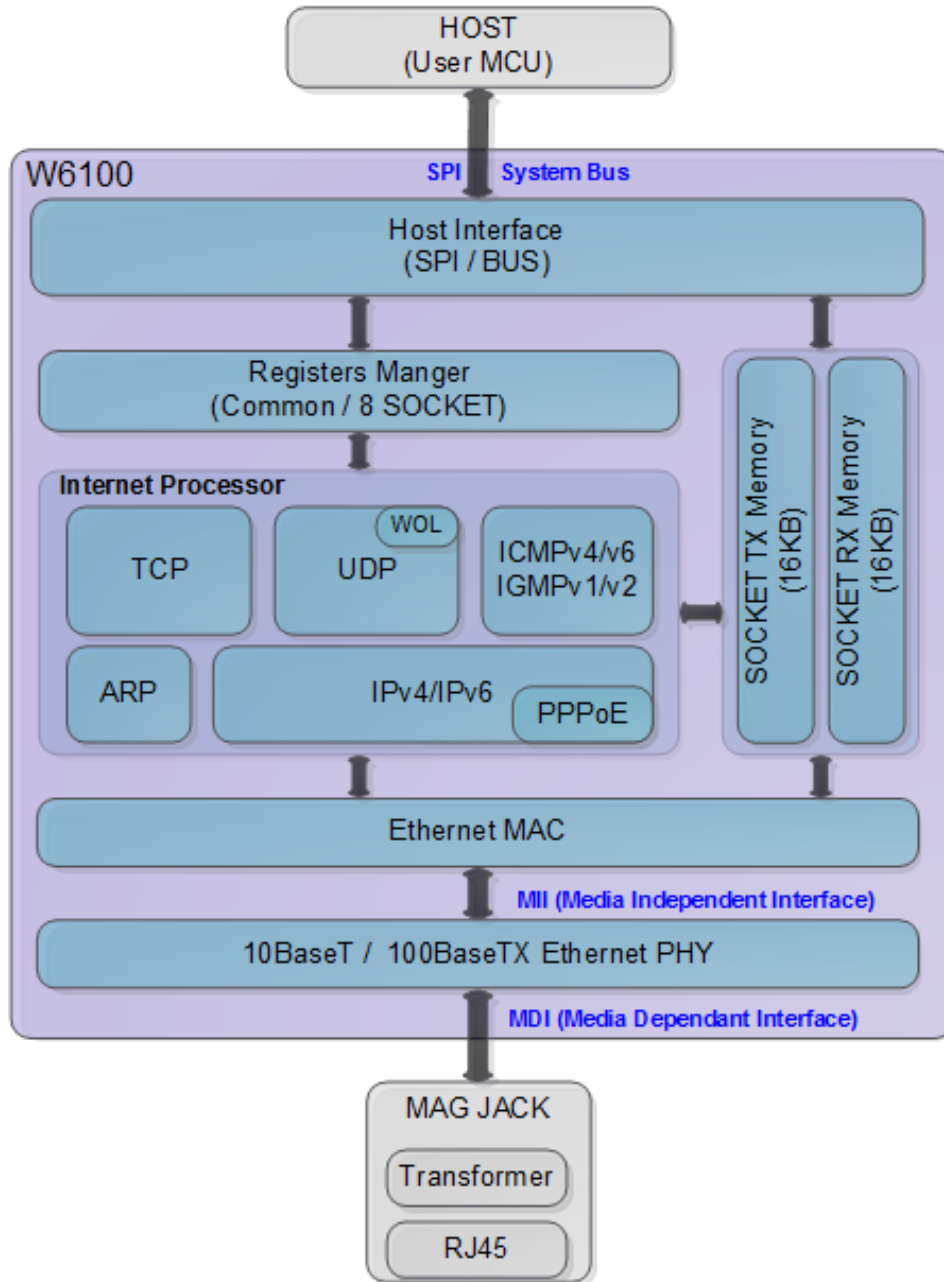


Figure 1 Block Diagram

---

# Contents

<b>1</b>	<b>PIN Description .....</b>	<b>12</b>
1.1	PIN Description .....	13
<b>2</b>	<b>Memory Map .....</b>	<b>17</b>
<b>3</b>	<b>W6100 Registers .....</b>	<b>19</b>
3.1	Common Register .....	19
3.2	SOCKET Register .....	26
<b>4</b>	<b>Register Descriptions.....</b>	<b>28</b>
4.1	Common Registers .....	29
4.1.1	CIDR (Chip Identification Register) .....	29
4.1.2	VER (Version Register).....	29
4.1.3	SYSR (System Status Register).....	29
4.1.4	SYCR0 (System Config Register 0).....	30
4.1.5	SYCR1 (System Config Register 1).....	30
4.1.6	TCNTR (Tick Counter Register).....	31
4.1.7	TCNTRCLR (TCNTR Clear Register) .....	31
4.1.8	IR (Interrupt Register) .....	31
4.1.9	SIR (SOCKET Interrupt Register).....	32
4.1.10	SLIR (SOCKET-less Interrupt Register).....	32
4.1.11	IMR (Interrupt Mask Register) .....	33
4.1.12	IRCLR (IR Clear Register) .....	34
4.1.13	SIMR (SOCKET Interrupt Mask Register) .....	34
4.1.14	SLIMR (SOCKET-less Interrupt Mask Register) .....	35
4.1.15	SLIRCLR (SLIR Clear Register).....	35
4.1.16	SLPSR (SOCKET-less Prefer Source IPv6 Address Register) .....	36
4.1.17	SLCR (SOCKET-less Command Register) .....	36
4.1.18	PHYSR (PHY Status Register) .....	37
4.1.19	PHYRAR (PHY Register Address Register) .....	37
4.1.20	PHYDIR (PHY Data Input Register) .....	38
4.1.21	PHYDOR (PHY Data Output Register) .....	38
4.1.22	PHYACR (PHY Access Control Register) .....	38
4.1.23	PHYDIVR (PHY Division Register) .....	38
4.1.24	PHYCR0 (PHY Control Register 0) .....	39
4.1.25	PHYCR1 (PHY Control Register 1) .....	39
4.1.26	NET4MR (Network IPv4 Mode Register).....	40
4.1.27	NET6MR (Network IPv6 Mode Register).....	41
4.1.28	NETMR (Network Mode Register).....	42
4.1.29	NETMR2 (Network Mode Register 2) .....	43
4.1.30	PTMR (PPP Link Control Protocol Request Timer Register) .....	43
4.1.31	PMNR (PPP Link Control Protocol Magic number Register) .....	43

---

4.1.32	PHAR (PPPoE Server Hardware Address Register on PPPoE) .....	44
4.1.33	PSIDR (PPPoE Session ID Register on PPPoE) .....	44
4.1.34	PMRUR (PPPoE Maximum Receive Unit Register) .....	44
4.1.35	SHAR (Source Hardware Address Register) .....	44
4.1.36	GAR (Gateway IP Address Register).....	45
4.1.37	SUBR (Subnet Mask Register) .....	45
4.1.38	SIPR (IPv4 Source Address Register) .....	45
4.1.39	LLAR (Link Local Address Register) .....	45
4.1.40	GUAR (Global Unicast Address Register) .....	46
4.1.41	SUB6R (IPv6 Subnet Prefix Register).....	46
4.1.42	GA6R (IPv6 Gateway Address Register) .....	47
4.1.43	SLDIP6R (SOCKET-less Destination IPv6 Address Register).....	47
4.1.44	SLDIPR (SOCKET-less Destination IPv4 Address Register) .....	47
4.1.45	SLDHAR (SOCKET-less Destination Hardware Address Register) .....	48
4.1.46	PINGIDR (PING ID Register) .....	48
4.1.47	PINGSEQR (PING Sequence-number Register) .....	48
4.1.48	UIPR (Unreachable IP Address Register) .....	48
4.1.49	UPORTR (Unreachable Port Register) .....	48
4.1.50	UIP6R (Unreachable IPv6 Address Register) .....	49
4.1.51	UPORT6R (Unreachable IPv6 Port Register) .....	49
4.1.52	INTPTMR (Interrupt Pending Time Register).....	49
4.1.53	PLR (Prefix Length Register).....	50
4.1.54	PFR (Prefix Flag Register).....	50
4.1.55	VLTR (Valid Life Time Register).....	50
4.1.56	PLTR (Preferred Life Time Register) .....	50
4.1.57	PAR (Prefix Address Register) .....	51
4.1.58	ICMP6BLKR (ICMPv6 Block Register) .....	51
4.1.59	CHPLCKR (Chip Lock Register).....	52
4.1.60	NETLCKR (Network Lock Register) .....	52
4.1.61	PHYLCKR (PHY Lock Register) .....	52
4.1.62	RTR (Retransmission Time Register).....	52
4.1.63	RCR (Retransmission Count Register).....	52
4.1.64	SLRTR (SOCKET-less Retransmission Time Register) .....	53
4.1.65	SLRCR (SOCKET-less Retransmission Count Register) .....	53
4.1.66	SLHOPR (Hop limit Register).....	53
4.2	SOCKET Register.....	54
4.2.1	Sn_MR (SOCKET n Mode Register) .....	54
4.2.2	Sn_PSR (SOCKET n Prefer Source IPv6 Address Register) .....	56
4.2.3	Sn_CR (SOCKET n Command Register).....	56
4.2.4	Sn_IR (SOCKET n Interrupt Register) .....	58
4.2.5	Sn_IMR (SOCKET n Interrupt Mask Register) .....	59

4.2.6	Sn_IRCLR (Sn_IR Clear Register) .....	59
4.2.7	Sn_SR (SOCKET n Status Register) .....	60
4.2.8	Sn_ESR (SOCKET n Extension Status Register).....	61
4.2.9	Sn_PNR (SOCKET n IP Protocol Number Register) .....	62
4.2.10	Sn_TOSR (SOCKET n IP Type of Service Register).....	62
4.2.11	Sn_TTLR (SOCKET n IP Time To Live Register) .....	62
4.2.12	Sn_FRGR (SOCKET n Fragment Offset in IP Header Register) .....	62
4.2.13	Sn_MSSR (SOCKET n Maximum Segment Size Register) .....	62
4.2.14	Sn_PORTR (SOCKET n Source Port Register).....	63
4.2.15	Sn_DHAR (SOCKET n Destination Hardware Address Register) .....	63
4.2.16	Sn_DIPR (SOCKET n Destination IPv4 Address Register).....	63
4.2.17	Sn_DIP6R (SOCKET n Destination IPv6 Address Register) .....	64
4.2.18	Sn_DPORTR (SOCKET n Destination Port Register) .....	64
4.2.19	Sn_MR2 (SOCKET n Mode register 2).....	65
4.2.20	Sn_RTR (SOCKET n Retransmission Time Register) .....	66
4.2.21	Sn_RCR (SOCKET n Retransmission Count Register) .....	66
4.2.22	Sn_KPALVTR (SOCKET n Keep Alive Time Register) .....	66
4.2.23	Sn_TX_BSR (SOCKET n TX Buffer Size Register).....	67
4.2.24	Sn_TX_FSR (SOCKET n TX Free Buffer Size Register).....	67
4.2.25	Sn_TX_RD (SOCKET n TX Read Pointer Register).....	67
4.2.26	Sn_TX_WR (SOCKET n TX Write Pointer Register) .....	68
4.2.27	Sn_RX_BSR (SOCKET n RX Buffer Size Register) .....	68
4.2.28	Sn_RX_RSR (SOCKET n RX Received Size Register) .....	69
4.2.29	Sn_RX_RD (SOCKET n RX Read Pointer Register).....	69
4.2.30	Sn_RX_WR (SOCKET n RX Write Pointer Register) .....	69
<b>5</b>	<b>HOST Interface Mode .....</b>	<b>70</b>
5.1	SPI Mode .....	70
5.1.1	SPI Frame.....	71
5.1.2	Variable Length Data Mode (VDM) .....	74
5.1.3	Fixed Length Data Mode (FDM) .....	75
5.2	Parallel BUS Mode .....	77
5.2.1	Parallel BUS Mode Data Write .....	79
5.2.2	Parallel Bus Mode Data Read .....	79
<b>6</b>	<b>Functional Description .....</b>	<b>80</b>
6.1	Initialization .....	80
6.1.1	Network Information Setting .....	80
6.1.2	SOCKET TX/RX Buffer Size Setting.....	81
6.2	TCP.....	82
6.2.1	TCP SERVER .....	83
6.2.2	TCP CLIENT.....	88
6.2.3	TCP DUAL.....	89

---

6.2.4	Other Functions .....	91
6.3	UDP.....	93
6.3.1	UDP Unicast .....	93
6.3.2	UDP Broadcast .....	96
6.3.3	UDP Multicast .....	98
6.3.4	UDP DUAL .....	100
6.3.5	Other Functions .....	101
6.4	IPRAW .....	103
6.4.1	Other Functions .....	107
6.5	MACRAW .....	108
6.6	SOCKET-less Command (SLCR).....	110
6.6.1	ARP .....	110
6.6.2	PING.....	112
6.6.3	ARP6 (ND, Neighbor Discovery).....	113
6.6.4	PING6 (ICMPv6 Echo).....	114
6.6.5	DAD (Duplicate Address Detection).....	115
6.6.6	RS (Router Solicitation) .....	118
6.6.7	Unsolicited NA(Neighbor Advertisement) .....	119
6.7	Retransmission.....	121
6.7.1	ARP & PING & ND Retransmission .....	121
6.7.2	TCP Retransmission .....	122
6.8	Others Functions .....	123
6.8.1	System Clock(SYS_CLK) Switching .....	123
6.8.2	Ethernet PHY Operation Mode Configuration .....	123
6.8.3	Ethernet PHY Parallel Detection.....	124
6.8.4	Ethernet PHY Auto MDIX .....	124
6.8.5	Ethernet PHY Power Down Mode.....	124
6.8.6	Ethernet PHY's Registers Control .....	125
6.8.7	Ethernet PHY 10BASE-Te Mode .....	128
<b>7</b>	<b>Clock &amp; Transformer Requirements .....</b>	<b>129</b>
7.1	Quartz Crystal Requirements. ....	129
7.2	Oscillator requirements. ....	130
7.3	Transformer Characteristics .....	130
<b>8</b>	<b>Electrical Specification .....</b>	<b>131</b>
8.1	Absolute Maximum ratings .....	131
8.2	Absolute Maximum ratings (Electrical Sensitivity).....	131
8.3	DC Characteristics .....	132
8.4	AC Characteristics.....	133
8.4.1	Reset Timing .....	133
8.4.2	BUS ACCESS TIMING .....	134
8.4.3	SPI ACCESS TIMING .....	135



---

8.4.4	Transformer Characteristics .....	135
8.4.5	MDIX.....	136
8.5	POWER DISSIPATION .....	136
<b>9</b>	<b>Package Information.....</b>	<b>137</b>
9.1	LQFP48.....	137
9.2	QFN48.....	139
<b>10</b>	<b>Document Revision History .....</b>	<b>140</b>

---

## List of Figures

Figure 1 Block Diagram .....	4
Figure 2 W6100 Pin Layout .....	12
Figure 3 W6100 Memory Map .....	17
Figure 4 State Diagram .....	61
Figure 5 Variable Length Data Mode (CSn controlled by Host).....	70
Figure 6 Fixed Length Data Mode (CSn is always connected by Groud).....	70
Figure 7 SPI Mode 0 & Mode 3.....	71
Figure 8 SPI Frame Format .....	71
Figure 9 Write SPI Frame in VDM .....	74
Figure 10 Read SPI Frame in VDM .....	75
Figure 11 1 byte Data Write Access SPI Frame in FDM .....	76
Figure 12 2 bytes Data Write Access SPI Frame in FDM .....	76
Figure 13 4 bytes Data Write Access SPI Frame in FDM .....	76
Figure 14 1 byte Data Read Access SPI Frame in FDM .....	76
Figure 15 2 bytes Data Read Access SPI Frame in FDM.....	76
Figure 16 4 bytes Data Read Access SPI Frame in FDM.....	76
Figure 17 Host Interface in Parallel BUS Mode .....	77
Figure 18 Parallel Bus N-Bytes Data Write Access.....	79
Figure 19 Parallel Bus Mode Continuous Read Access .....	79
Figure 20 TCP SERVER and TCP CLIENT .....	82
Figure 21 TCP SERVER Operation Flow .....	83
Figure 22 TCP CLIENT Operation Flow .....	88
Figure 23 UDP Operation Flow .....	93
Figure 24 Received DATA in UDP Mode SOCKET RX Buffer Block.....	94
Figure 25 IPv6 Multicast-Group Address Format.....	98
Figure 26 IPRAW Operation Flow .....	104
Figure 27 Received DATA in IPRAW4 Mode SOCKET RX Buffer Block .....	105
Figure 28 Received DATA in IPRAW6 Mode SOCKET RX Buffer Block .....	105
Figure 29 MACRAW Operation Flow .....	108
Figure 30 Received DATA Format in MACRAW .....	109
Figure 31 SOCKET-less Command Operation Flow .....	110
Figure 32 DAD Operation Flow .....	116
Figure 33 RS Operation Flow .....	118
Figure 34 Unsolicited NA Operation Flow.....	120
Figure 35 MDC/MDIO Write Control Flow.....	126
Figure 36 MDC/MDIO Read Control Flow .....	127
Figure 37 Quartz Crystal Model.....	129
Figure 38 Transformer Type.....	130
Figure 39 Reset Timing.....	133

---

Figure 40 Bus Read Timing .....	134
Figure 41 BUS Write Timing .....	134
Figure 42 SPI Access Timing .....	135
Figure 43 Transformer Type.....	136

## List of Tables

Table 1 Pin Type Notation .....	12
Table 2 PIN Description.....	13
Table 3 Registers in Parallel BUS Mode.....	77
Table 4 Parameter Description in PACKET INFO .....	94
Table 5 Parameters of Flags in IPv6 Multicast Address.....	98
Table 6 Definition of Scope in IPv6 Multicast Address.....	98
Table 7 Internet Protocol Supported In IPRAW Mode .....	103
Table 8 parameters of ‘PACKET INFO’ in IPRAW4 Mode .....	105
Table 9 parameters of ‘PACKET INFO’ in IPRAW6 Mode .....	105
Table 10 Quartz Crystal .....	129
Table 11 Crystal Recommendation Characteristics .....	129
Table 12 Oscillator Characteristics.....	130
Table 13 Transformer Characteristics .....	130
Table 14 Absolute Maximum ratings .....	131
Table 15 Electro Static Discharge (ESD) .....	131
Table 16 Latch up Test .....	131
Table 17 DC Characteristics .....	132
Table 18 Reset Table .....	133
Table 19 BUS Read Timing .....	134
Table 20 BUS Write timing.....	135
Table 21 SPI Access Timing .....	135
Table 22 Transformer Characteristics .....	135
Table 23 Power Dissipation .....	136
Table 24 LQFP48 VARIATIONS (ALL DEMINSIONS SHOWN IN MM) .....	137
Table 25 QFN48 VARIATIONS (ALL DEMINSIONS SHOWN IN MM) .....	139

# 1 PIN Description

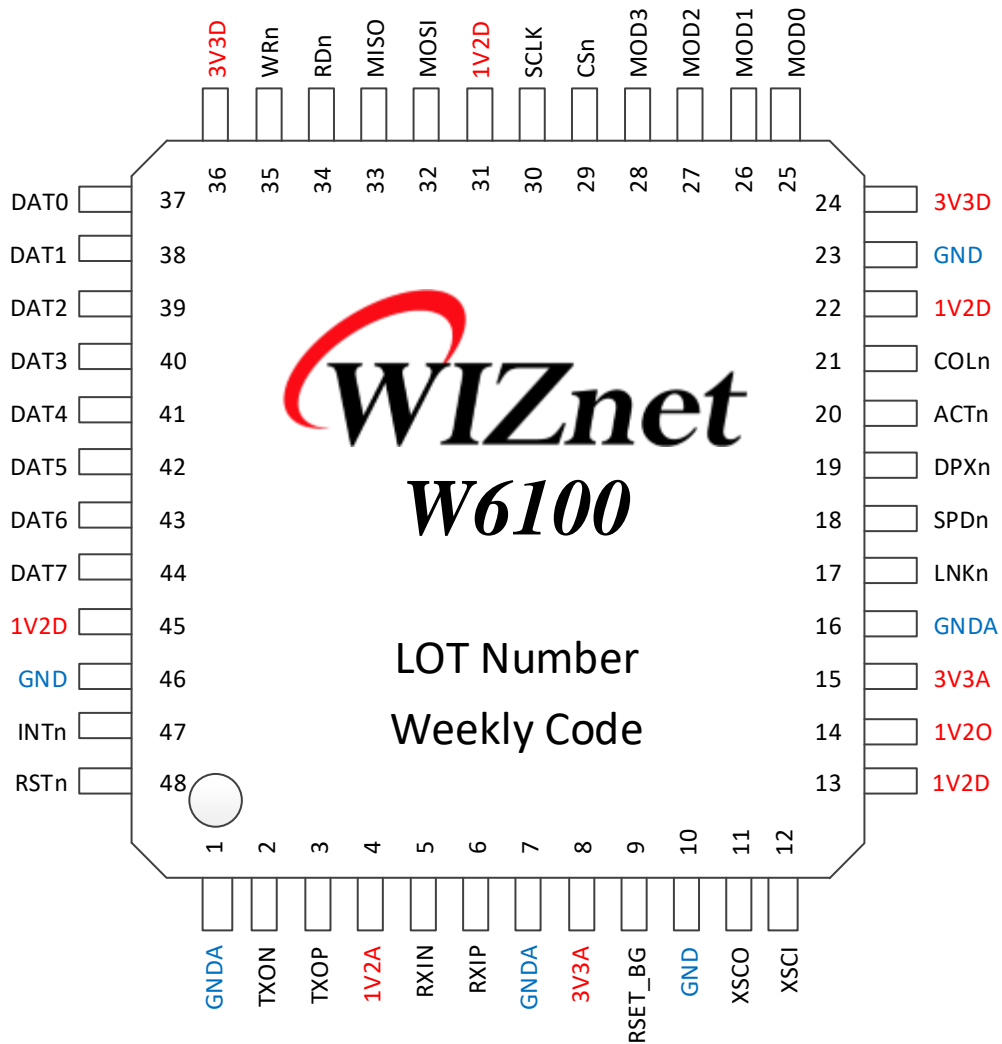


Figure 2 W6100 Pin Layout

Table 1 Pin Type Notation

Type	Description
I	Input
O	Output
M	Alternate (Multi-function) Signal
U	Internal pulled-up 75KΩ resistor
D	Internal pulled-down 75KΩ 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> Data 를 MDI Mode 에서 TXOP/TXON 차동 신호 쌍을 통해 미디어로 전송한다.
3	TXOP	AO	
4	1V2A	AP	<b>Analog 1.2V Power</b> 1V20 전압 소스로부터 공급받는다.
5	RXIN	AI	<b>Differential Received Signal Pair</b> Data 를 MDI 모드에서 RXIP/RXON 차동 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> 25MHz Crystal Oscillator(XTAL)나 Oscillator(OSC)로 연결한다.
12	XSCI	AI	W6100 은 외부 25MHz 의 Clock 을 25MHz(Low Frequency mode)나 그 4 배수인 100MHz(Normal mode)로 변환하고 이를 내부 Operation Clock(SYS_CLK) 으로 사용한다.  <b>* CAUTION</b> OSC 를 사용할 경우 25MHz@1.2V 를 사용하며 XSCI 만 연결하고 XSCO 는 반드시 Floating 시켜야한다.  참조) <a href="#">Clock Selection Guide</a> , same as W5100S
13	1V2D	P	<b>Digital 1.2V Power</b> 1V20 전압 소스로부터 공급받는다.
14	1V20	PO	<b>Internal Regulator 1.2V Power Output</b>  W6100 을 위한 내부 Regulator 의 1.2V Power Output 으로 Max 150mA 를 지원하며, 반드시 외부 안정화 Capacitor 3.3 $\mu$ F 를 통해 W6100 의 1V2D 와 1V2A 로 공급한다.  1V20 는 Ferrite Bead 를 사용하여, 1V2D 와 1V2A 분리 하여 공급한다.

			* <b>CAUTION</b> 이 Power 는 W6100 을 위한 것으로 다른 Device 의 Power 로 사용될 수 없다.
15	3V3A	AP	<b>Analog 3.3V Power</b>
16	GND A	AP	<b>Analog Ground</b>
17	LNKn	OU	<b>Link Status LED</b> SPI, Parallel Bus Mode 인 경우, 유효하다.  Low : Link up High : Link down
18	SPDn	OU	<b>Link Speed LED</b> SPI, Parallel Bus Mode 인 경우, 유효하다.  Low : 100Mbps High : 10Mbps
19	DPXn	OU	<b>Link Duplex LED</b> SPI, Parallel Bus Mode 인 경우, 유효하다.  Low : Full-Duplex High : Half-Duplex
20	ACTn	OU	<b>Link Activity LED</b> SPI, Parallel Bus Mode 인 경우, 유효하다.  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 인 경우, 유효하다. 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>W6100 Mode Selection</b> MOD[3:0]에 따라 아래와 같이 선택한다.
26	MOD[1]	ID	
27	MOD[2]	ID	

28	MOD[3]	ID	“010X” : Parallel Bus Mode Others : Reserved
29	CSn	IU	<b>W6100 Chip Select</b> Low : Select High : No Select
30	SCLK	ID	<b>SPI Clock</b> SPI Mode 인 경우, SPI Clock 을 입력한다. Parallel Bus Mode 인 경우, GND 와 연결하거나 Floating 한다.
31	1V2D	P	<b>Digital 1.2V Power</b> 1V20 로부터 공급받는다.
32	MOSI /ADDR0	IDM	<b>SPI Master Output Slave Input / Address 0</b> MOSI : SPI mode 인 경우, SPI Data 를 HOST 로부터 수신 ADDR0 : Parallel Bus Mode 인 경우, Address 0 을 입력
33	MISO /ADDR1	IOPM	<b>SPI Master Input Slave Output / Address 1</b> MISO : SPI mode 인 경우, SPI Data 를 HOST 로 송신 ADDR1 : Parallel Bus Mode 인 경우, 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, W6100 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	<p><b>Interrupt</b></p> <p>W6100 의 Ethernet 통신 처리시 Event 가 발생할 경우, HOST 에게 알린다.</p> <p>Low : Interrupt Occurred High : No Interrupt</p> <p><i>참조) SYCR1 의 IEN(Interrupt Enable), INTPTMR(Interrupt Pending Time Register), IR(Interrupt Register), SIR(SOCKET Interrupt Register), SLIR(SOCKET-less Interrupt Register)</i></p>
48	RSTn	IP	<p><b>Reset</b></p> <p>W6100 을 초기화 한다. RSTn 신호는 반드시 1.0us 이상 Low 를 유지해야 한다. W6100 는 RSTn 신호가 인가된 후 60.3ms 이후에 완전히 초기화가 된다.</p> <p><i>참조) 8.4.1 Reset Timing</i></p> <p>Low : W6100 를 초기화 시킨다. High : W6100 를 정상 동작 시킨다.</p>



## 2 Memory Map

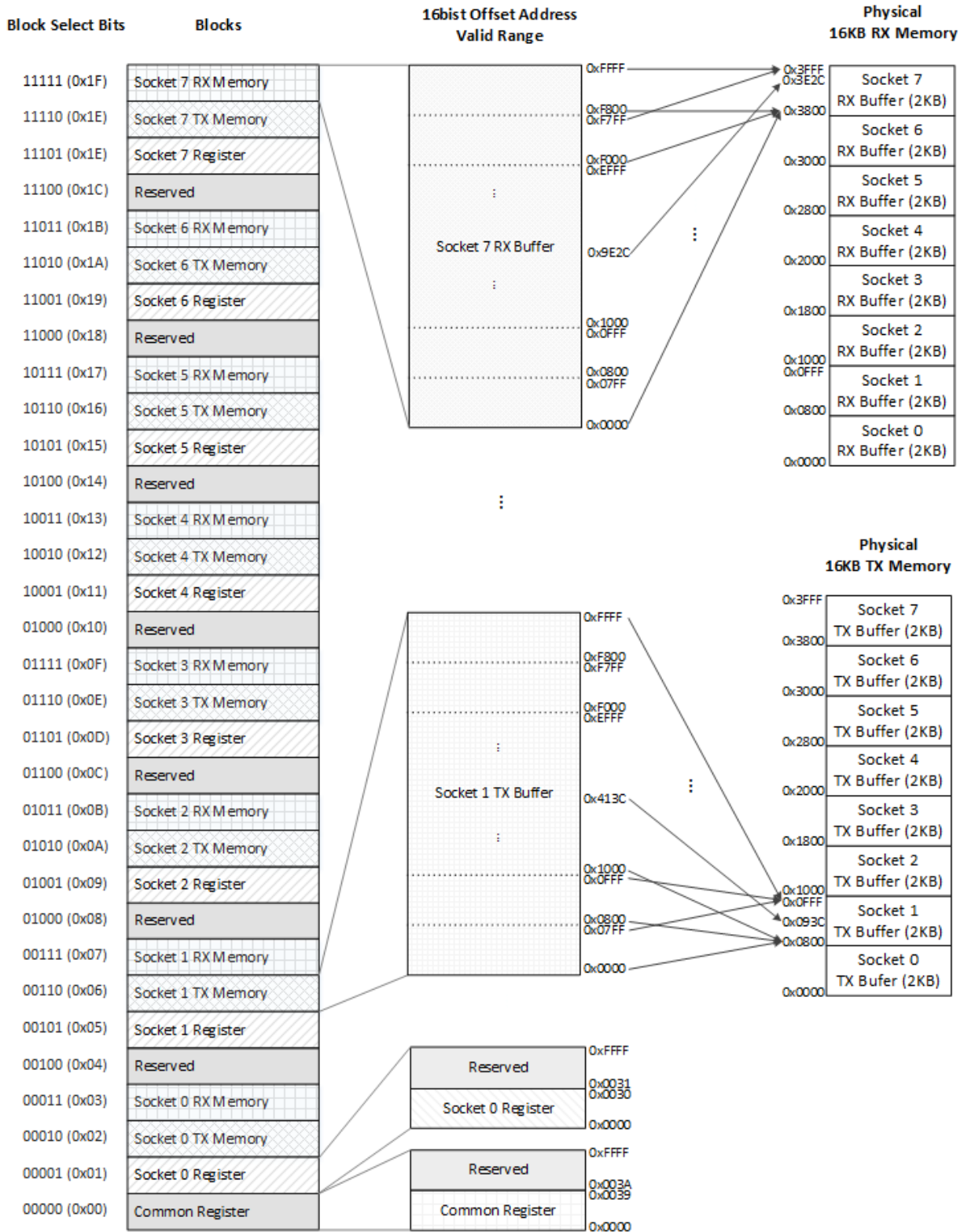


Figure 3 W6100 Memory Map

Figure 3 에서, W6100 은 다음과 같은 Block 으로 구성된다.

- 1 x Common Register Block, 7 x Reserved Block
- 8 x SOCKET n Register Block
- 8 x SOCKET n TX Buffer Block
- 8 x SOCKET n RX Buffer Block

이들 Block 은 5 bits 의 Block Select Bits 로 구분되며, 각 Block 내의 영역들은 Byte 단위의 16bits Offset Address 로 Access 된다.

8 개의 SOCKET n TX buffer Block 은 16KB TX Memory 영역 내에서 초기 2KB 씩 각각 할당되며, Sn\_TX\_BSR(4.2.23)을 통해 0, 1, 2, 4, 8 또는 16KB 크기로 재설정될 수 있다. 할당된 SOCKET n TX Buffer Block 의 총합이 16KB 를 초과하지 않도록 주의한다.

8 개의 SOCKET n RX buffer Block 은 16KB RX Memory 영역 내에서 초기 2KB 씩 각각 할당되며, Sn\_RX\_BSR(4.2.27)을 통해 0, 1, 2, 4, 8, 또는 16KB 크기로 재설정될 수 있다. 할당된 SOCKET n RX Buffer Block 의 총합이 16KB 를 초과하지 않도록 주의한다.

## 3 W6100 Registers

### 3.1 Common Register

Offset	Register	Type <sup>1</sup>	Reset
0x0000	CIDR0 (Chip Identification Register)	RO	0x61
0x0001	CIDR1	RO	0x00
0x0002	VER0 (Chip Version Register)	RO	0x46
0x0003	VER1	RO	0x61
0x2000	SYSR (System Status Register)	RO	0xEU
0x2004	SYCR0 (System Config Register 0)	WO	0x80
0x2005	SYCR1	R=W	0x80
0x2016	TCNTR0 (Tick Counter Register)	RO	0x00
0x2017	TCNTR1	RO	0x00
0x2020	TCNTCLR (TCNTR Clear Register)	WO	0x00
0x2100	IR (Interrupt Register)	RO	0x00
0x2101	SIR (SOCKET Interrupt Register)	RO	0x00
0x2102	SLIR (SOCKET-less Interrupt Register)	RO	0x00
0x2104	IMR (Interrupt Mask Register)	R=W	0x00
0x2108	IRCLR (IR Clear Register)	WO	0x00
0x2114	SIMR (SOCKET Interrupt Mask Register)	R=W	0x00
0x2124	SLIMR (SOCKET-less Interrupt Mask Register)	R=W	0x00
0x2128	SLIRCLR (SLIR Clear Register)	WO	0x00
0x212C	SLPSR (SOCKET-less Prefer Source IPv6 Address Register)	R=W	0x00
0x2130	SLCR (SOCKET-less Command Register)	RW,AC	0x00
0x3000	PHYSR (PHY Status Register)	RO	0x00
0x3008	PHYRAR (PHY Register Address Register)	R=W	0x00
0x300C	PHYDIR0 (PHY Data Input Register)	R=W	0x00
0x300D	PHYDIR1	R=W	0x00
0x3010	PHYDOR0 (PHY Data Output Register)	RO	0x00
0x3011	PHYDOR1	RO	0x00
0x3014	PHYACR (PHY Access Control Register)	RW,AC	0x00
0x3018	PHYDIVR (PHY Division Register)	R=W	0x01
0x301C	PHYCR0 (PHY Control Register 0)	WO	0x00

<sup>1</sup> 참조 4 Register Descriptions

0x301D	PHYCR1	R=W	0x40
0x4000	NET4MR (Network IPv4 Mode Register)	R=W	0x00
0x4004	NET6MR (Network IPv6 Mode Register)	R=W	0x00
0x4008	NETMR (Network Mode Register)	R=W	0x00
0x4009	NETMR2 (Network Mode Register 2)	R=W	0x00
0x4100	PTMR (PPP Link Control Protocol Request Timer Register)	R=W	0x28
0x4104	PMNR (PPP Link Control Protocol Magic number Register)	R=W	0x00
0x4108	PHAR0 (PPPoE Hardware Address Register on PPPoE)	R=W	0x00
0x4109	PHAR1	R=W	0x00
0x410A	PHAR2	R=W	0x00
0x410B	PHAR3	R=W	0x00
0x410C	PHAR4	R=W	0x00
0x410D	PHAR5	R=W	0x00
0x4110	PSIDR0 (PPPoE Session ID Register)	R=W	0x00
0x4111	PSIDR1	R=W	0x00
0x4114	PMUR0 (PPPoE Maximum Receive Unit Register)	R=W	0xFF
0x4115	PMUR1	R=W	0xFF
0x4120	SHAR0 (Source Hardware Address Register)	R=W	0x00
0x4121	SHAR1	R=W	0x00
0x4122	SHAR2	R=W	0x00
0x4123	SHAR3	R=W	0x00
0x4124	SHAR4	R=W	0x00
0x4125	SHAR5	R=W	0x00
0x4130	GAR0 (Gateway IP Address Register)	R=W	0x00
0x4131	GAR1	R=W	0x00
0x4132	GAR2	R=W	0x00
0x4133	GAR3	R=W	0x00
0x4134	SUBR0 (Subnet Mask Register)	R=W	0x00
0x4135	SUBR1	R=W	0x00
0x4136	SUBR2	R=W	0x00
0x4137	SUBR3	R=W	0x00
0x4138	SIPR0 (IPv4 Source Address Register)	R=W	0x00
0x4139	SIPR1	R=W	0x00
0x413A	SIPR2	R=W	0x00
0x413B	SIPR3	R=W	0x00

0x4140	LLAR0 (Link Local Address Register)	R=W	0x00
0x4141	LLAR1	R=W	0x00
0x4142	LLAR2	R=W	0x00
0x4143	LLAR3	R=W	0x00
0x4144	LLAR4	R=W	0x00
0x4145	LLAR5	R=W	0x00
0x4146	LLAR6	R=W	0x00
0x4147	LLAR7	R=W	0x00
0x4148	LLAR8	R=W	0x00
0x4149	LLAR9	R=W	0x00
0x414A	LLAR10	R=W	0x00
0x414B	LLAR11	R=W	0x00
0x414C	LLAR12	R=W	0x00
0x414D	LLAR13	R=W	0x00
0x414E	LLAR14	R=W	0x00
0x414F	LLAR15	R=W	0x00
0x4150	GUAR0 (Global Unicast Address Register)	R=W	0x00
0x4151	GUAR1	R=W	0x00
0x4152	GUAR2	R=W	0x00
0x4153	GUAR3	R=W	0x00
0x4154	GUAR4	R=W	0x00
0x4155	GUAR5	R=W	0x00
0x4156	GUAR6	R=W	0x00
0x4157	GUAR7	R=W	0x00
0x4158	GUAR8	R=W	0x00
0x4159	GUAR9	R=W	0x00
0x415A	GUAR10	R=W	0x00
0x415B	GUAR11	R=W	0x00
0x415C	GUAR12	R=W	0x00
0x415D	GUAR13	R=W	0x00
0x415E	GUAR14	R=W	0x00
0x415F	GUAR15	R=W	0x00
0x4160	SUB6R0 (IPv6 Subnet Prefix Register)	R=W	0x00
0x4161	SUB6R1	R=W	0x00
0x4162	SUB6R2	R=W	0x00

0x4163	SUB6R3	R=W	0x00
0x4164	SUB6R4	R=W	0x00
0x4165	SUB6R5	R=W	0x00
0x4166	SUB6R6	R=W	0x00
0x4167	SUB6R7	R=W	0x00
0x4168	SUB6R8	R=W	0x00
0x4169	SUB6R9	R=W	0x00
0x416A	SUB6R10	R=W	0x00
0x416B	SUB6R11	R=W	0x00
0x416C	SUB6R12	R=W	0x00
0x416D	SUB6R13	R=W	0x00
0x416E	SUB6R14	R=W	0x00
0x416F	SUB6R15	R=W	0x00
0x4170	GA6R0 (IPv6 Gateway Address Register)	R=W	0x00
0x4171	GA6R1	R=W	0x00
0x4172	GA6R2	R=W	0x00
0x4173	GA6R3	R=W	0x00
0x4174	GA6R4	R=W	0x00
0x4175	GA6R5	R=W	0x00
0x4176	GA6R6	R=W	0x00
0x4177	GA6R7	R=W	0x00
0x4178	GA6R8	R=W	0x00
0x4179	GA6R9	R=W	0x00
0x417A	GA6R10	R=W	0x00
0x417B	GA6R11	R=W	0x00
0x417C	GA6R12	R=W	0x00
0x417D	GA6R13	R=W	0x00
0x417E	GA6R14	R=W	0x00
0x417F	GA6R15	R=W	0x00
0x4180	SLDIP6R0 (SOCKET-less Destination IPv6 Address Register)	R=W	0x00
0x4181	SLDIP6R1	R=W	0x00
0x4182	SLDIP6R2	R=W	0x00
0x4183	SLDIP6R3	R=W	0x00
0x4184	SLDIP6R4	R=W	0x00
0x4185	SLDIP6R5	R=W	0x00

0x4186	SLDIP6R6	R=W	0x00
0x4187	SLDIP6R7	R=W	0x00
0x4188	SLDIP6R8	R=W	0x00
0x4189	SLDIP6R9	R=W	0x00
0x418A	SLDIP6R10	R=W	0x00
0x418B	SLDIP6R11	R=W	0x00
0x418C	SLDIP6R12 / SLDIRO (SOCKET-less Destination IPv4 Address Register)	R=W	0x00
0x418D	SLDIP6R13 / SLDIR1	R=W	0x00
0x418E	SLDIP6R14 / SLDIR2	R=W	0x00
0x418F	SLDIP6R15 / SLDIR3	R=W	0x00
0x4190	SLDHAR0 (SOCKET-less Destination Hardware Address Register)	RO	0x00
0x4191	SLDHAR1	RO	0x00
0x4192	SLDHAR2	RO	0x00
0x4193	SLDHAR3	RO	0x00
0x4194	SLDHAR4	RO	0x00
0x4195	SLDHAR5	RO	0x00
0x4198	PINGIDR0 (PING ID Register)	R=W	0x00
0x4199	PINGIDR1	R=W	0x00
0x419C	PINGSEQR0 (PING Sequence-number Register)	R=W	0x00
0x419D	PINGSEQR1	R=W	0x00
0x41A0	UIPR0 (Unreachable IP Address Register)	RO	0x00
0x41A1	UIPR1	RO	0x00
0x41A2	UIPR2	RO	0x00
0x41A3	UIPR3	RO	0x00
0x41A4	UPORTR0 (Unreachable Port Register)	RO	0x00
0x41A5	UPORTR1	RO	0x00
0x41B0	UIP6R0 (Unreachable IPv6 Address Register)	RO	0x00
0x41B1	UIP6R1	RO	0x00
0x41B2	UIP6R2	RO	0x00
0x41B3	UIP6R3	RO	0x00
0x41B4	UIP6R4	RO	0x00
0x41B5	UIP6R5	RO	0x00
0x41B6	UIP6R6	RO	0x00
0x41B7	UIP6R7	RO	0x00
0x41B8	UIP6R8	RO	0x00

0x41B9	UIP6R9	RO	0x00
0x41BA	UIP6R10	RO	0x00
0x41BB	UIP6R11	RO	0x00
0x41BC	UIP6R12	RO	0x00
0x41BD	UIP6R13	RO	0x00
0x41BE	UIP6R14	RO	0x00
0x41BF	UIP6R15	RO	0x00
0x41C0	UPOINT6R0 (Unreachable IPv6 Port Register)	RO	0x00
0x41C1	UPOINT6R1	RO	0x00
0x41C5	INTPTMR0 (Interrupt Pending Time Register)	R=W	0x00
0x41C6	INTPTMR1	R=W	0x00
0x41D0	PLR (Prefix Length Register)	RO	0x00
0x41D4	PFR (Prefix Flag Register)	RO	0x00
0x41D8	VLTR0 (Valid Life Time Register)	RO	0x00
0x41D9	VLTR1	RO	0x00
0x41DA	VLTR2	RO	0x00
0x41DB	VLTR3	RO	0x00
0x41DC	PLTR0 (Preferred Life Time Register)	RO	0x00
0x41DD	PLTR1	RO	0x00
0x41DE	PLTR2	RO	0x00
0x41DF	PLTR3	RO	0x00
0x41E0	PAR0 (Prefix Address Register)	RO	0x00
0x41E1	PAR1	RO	0x00
0x41E2	PAR2	RO	0x00
0x41E3	PAR3	RO	0x00
0x41E4	PAR4	RO	0x00
0x41E5	PAR5	RO	0x00
0x41E6	PAR6	RO	0x00
0x41E7	PAR7	RO	0x00
0x41E8	PAR8	RO	0x00
0x41E9	PAR9	RO	0x00
0x41EA	PAR10	RO	0x00
0x41EB	PAR11	RO	0x00
0x41EC	PAR12	RO	0x00
0x41ED	PAR13	RO	0x00



---

0x41EE	PAR14	RO	0x00
0x41EF	PAR15	RO	0x00
0x41F0	ICMP6BLKR (ICMPv6 Block Register)	R=W	0x00
0x41F4	CHPLCKR (Chip Lock Register)	WO	0x00
0x41F5	NETLCKR (Network Lock Register)	WO	0x00
0x41F6	PHYLCKR (PHY Lock Register)	WO	0x00
0x4200	RTR0 (Retransmission Time Register)	R=W	0x07
0x4201	RTR1	R=W	0xD0
0x4204	RCR (Retransmission Count Register)	R=W	0x08
0x4208	SLRTR0 (SOCKET-less Retransmission Time Register)	R=W	0x07
0x4209	SLRTR1	R=W	0xD0
0x420C	SLRCR (SOCKET-less Retransmission Count Register)	R=W	0x00
0x420F	SLHOPR (Hop limit Register)	R=W	0x80

### 3.2 SOCKET Register

Offset	Register	Type	Reset
0x0000	Sn_MR (SOCKET n Mode Register)	R=W	0x00
0x0004	Sn_PSR (SOCKET n Prefer Source IPv6 Address Register)	R=W	0x00
0x0010	Sn_CR (SOCKET n Command Register)	RW,AC	0x00
0x0020	Sn_IR (SOCKET n Interrupt Register)	WO	0x00
0x0024	Sn_IMR (SOCKET n Interrupt Mask Register)	R=W	0xFF
0x0028	Sn_IRCLR (Sn_IR Clear Register)	WO	0x00
0x0030	Sn_SR (SOCKET n Status Register)	RO	0x00
0x0031	Sn_ESR (SOCKET n Extension Status Register)	RO	0x00
0x0100	Sn_PNR (SOCKET n IP Protocol Number Register)	R=W	0x00
0x0104	Sn_TOSR (SOCKET n IP Type Of Service Register)	R=W	0x00
0x0108	Sn_TTLR (SOCKET n IP Time To Live Register)	R=W	0x80
0x010C	Sn_FRGR0 (SOCKET n Fragment Offset in IP Header Register)	R=W	0x40
0x010D	Sn_FRGR1	R=W	0x00
0x0110	Sn_MSSR0 (SOCKET n Maximum Segment Size Register)	RW	0x00
0x0111	Sn_MSSR1	RW	0x00
0x0114	Sn_PORTR0 (SOCKET n Source Port Register)	R=W	0x00
0x0115	Sn_PORTR1	R=W	0x00
0x0118	Sn_DHAR0 (SOCKET n Destination Hardware Address Register)	RW	0x00
0x0119	Sn_DHAR1	RW	0x00
0x011A	Sn_DHAR2	RW	0x00
0x011B	Sn_DHAR3	RW	0x00
0x011C	Sn_DHAR4	RW	0x00
0x011D	Sn_DHAR5	RW	0x00
0x0120	Sn_DIPR0 (SOCKET n Destination IPv4 Address Register)	RW	0x00
0x0121	Sn_DIPR1	RW	0x00
0x0122	Sn_DIPR2	RW	0x00
0x0123	Sn_DIPR3	RW	0x00
0x0130	Sn_DIP6R0 (SOCKET n Destination IPv6 Address Register)	RW	0x00
0x0131	Sn_DIP6R1	RW	0x00
0x0132	Sn_DIP6R2	RW	0x00
0x0133	Sn_DIP6R3	RW	0x00
0x0134	Sn_DIP6R4	RW	0x00

0x0135	Sn_DIP6R5	RW	0x00
0x0136	Sn_DIP6R6	RW	0x00
0x0137	Sn_DIP6R7	RW	0x00
0x0138	Sn_DIP6R8	RW	0x00
0x0139	Sn_DIP6R9	RW	0x00
0x013A	Sn_DIP6R10	RW	0x00
0x013B	Sn_DIP6R11	RW	0x00
0x013C	Sn_DIP6R12	RW	0x00
0x013D	Sn_DIP6R13	RW	0x00
0x013E	Sn_DIP6R14	RW	0x00
0x013F	Sn_DIP6R15	RW	0x00
0x0140	Sn_DPORTR0 (SOCKET n Destination Port Register)	RW	0x00
0x0141	Sn_DPORTR1	RW	0x00
0x0144	Sn_MR2 (SOCKET n Mode Register 2)	R=W	0x00
0x0180	Sn_RTR0 (SOCKET n Retransmission Time Register)	RW	0x00
0x0181	Sn_RTR1	RW	0x00
0x0184	Sn_RCR (SOCKET n Retransmission Count Register)	RW	0x00
0x0188	Sn_KPALVTR (SOCKET n Keep Alive Time Register)	R=W	0x00
0x0200	Sn_TX_BSR (SOCKET n TX Buffer Size Register)	R=W	0x02
0x0204	Sn_TX_FSR0 (SOCKET n TX Free Size Register)	RO	0x08
0x0205	Sn_TX_FSR1	RO	0x00
0x0208	Sn_TX_RD0 (SOCKET n TX Read Pointer Register)	RO	0x00
0x0209	Sn_TX_RD1	RO	0x00
0x020C	Sn_TX_WR0 (SOCKET n TX Write Pointer Register)	RW	0x00
0x020D	Sn_TX_WR1	RW	0x00
0x0220	Sn_RX_BSR (SOCKET n RX Buffer Size Register)	R=W	0x02
0x0224	Sn_RX_RSR0 (SOCKET n RX Received Size Register)	RO	0x00
0x0225	Sn_RX_RSR1	RO	0x00
0x0228	Sn_RX_RD0 (SOCKET n RX Read Pointer Register)	RW	0x00
0x0229	Sn_RX_RD1	RW	0x00
0x022C	Sn_RX_WR0 (SOCKET n RX Write Pointer Register)	RO	0x00
0x022D	Sn_RX_WR1	RO	0x00

## 4 Register Descriptions

### Register Notation

* Register Symbol (Register full Name)							
- [Register Type, 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]를 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 Register/bit
- [1] : Always read '1'
- [0] : Always read '0'
- [-] : Not available

### [Address Offset]: 레지스터의 Address Offset 을 나타낸다.

### [Reset Value]: Default Value.

Ex1) 4.1.28 NETMR (Network Mode Register)

[R=W][0x4008][0x00]

NETMR 은 각 종 Block Mode 와 WOL 을 설정한다.

7	6	5	4	3	2	1	0
-	-	ANB	M6B		WOL	IP6B	IP4B
		R=W	R=W		R=W	R=W	R=W

Ex2) NETMR[ANB]

NETMR 의 ANB bit

Ex3) NETMR[7:0]

NETMR 의 7<sup>th</sup> bit 부터 0<sup>th</sup> bit

## 4.1 Common Registers

### 4.1.1 CIDR (Chip Identification Register)

[RO][0x0000-0x0001] [0x6100]

CHIP ID 는 0x6100 이다.

CIDR0(0x0000)	CIDR1(0x0001)
0x61	0x00

### 4.1.2 VER (Version Register)

[RO][0x0002-0x0003] [0x4661]

Version 은 0x4661 이다.

VER0(0x0002)	VER1(0x0003)
0x46	0x61

### 4.1.3 SYSR (System Status Register)

[RO][0x2000] [0xEU]

CHIP/NET/PHY Configuration Lock 정보와 HOST Interface Mode 를 보여준다.

7	6	5	4	3	2	1	0
CHPL	NETL	PHYL	-	-	-	IND	SPI
RO	RO	RO				RO	RO

Bit	Symbol	Description															
7	CHPL	<p><b>CHIP Lock Status</b>            CHPLCKR(Chip Configuration Lock Register)에 의해 설정된다.</p> <p>0 : Unlock - SYCR0 &amp; SYCR1 설정 가능            1 : Lock - SYCR0 &amp; SYCR1 (System Configuration Register) 설정 불가</p>															
6	NETL	<p><b>NET Lock Status</b>            NETLCKR(Network Configuration Lock Register)에 의해 설정된다.</p> <p>0 : Unlock - Network Configuration Registers 설정 가능            1 : Lock - Network Configuration Registers 설정 불가</p> <p><b>참조) Network Configuration Registers to be locked by NETL</b></p> <table border="1"> <thead> <tr> <th></th> <th>Symbol</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td rowspan="4">IPv4</td> <td>SHAR</td> <td>Source Hardware Address Register</td> </tr> <tr> <td>GAR</td> <td>Gateway IP Address Register</td> </tr> <tr> <td>SUBR</td> <td>Subnet Mask Register</td> </tr> <tr> <td>SIPR</td> <td>Source IP Address Register</td> </tr> <tr> <td>IPv6</td> <td>LLAR</td> <td>Link Local Address Register</td> </tr> </tbody> </table>		Symbol	Description	IPv4	SHAR	Source Hardware Address Register	GAR	Gateway IP Address Register	SUBR	Subnet Mask Register	SIPR	Source IP Address Register	IPv6	LLAR	Link Local Address Register
	Symbol	Description															
IPv4	SHAR	Source Hardware Address Register															
	GAR	Gateway IP Address Register															
	SUBR	Subnet Mask Register															
	SIPR	Source IP Address Register															
IPv6	LLAR	Link Local Address Register															

		<table border="1"> <tr> <td>GUAR</td> <td>Global Unicast Address Register</td> </tr> <tr> <td>SUB6R</td> <td>IPv6 Subnet Prefix Register</td> </tr> <tr> <td>GA6R</td> <td>IPv6 Gateway IP Address It is excluded from lock mechanism</td> </tr> </table>	GUAR	Global Unicast Address Register	SUB6R	IPv6 Subnet Prefix Register	GA6R	IPv6 Gateway IP Address It is excluded from lock mechanism
GUAR	Global Unicast Address Register							
SUB6R	IPv6 Subnet Prefix Register							
GA6R	IPv6 Gateway IP Address It is excluded from lock mechanism							
		<b>* CAUTION : GA6R 은 NETL 에 무관하게 설정가능하다.</b>						
5	PHYL	<b>PHY Lock Status</b> PHYLOCKR(PHY Configuration Lock Register)에 의해 설정된다.  0 : Unlock - PHYCR0, PHYCR1 설정 가능 1 : Lock - PHY Control Register (PHYCR0, PHYCR1) 설정 불가						
[4:2]	-	Reserved						
1	IND	<b>Parallel Bus Interface Mode</b> 0 : others 1 : PIN MODE[3:0] = "010X"						
0	SPI	<b>SPI Interface Mode</b> 0 : others 1 : PIN MODE[3:0] = "000X"						

#### 4.1.4 SYCR0 (System Config Register 0)

[WO][0x2004] [0x80]

SYCR0 은 Software reset 을 수행한다.

SYCR0 은 SYSR[CHPL] = '0' 일 경우 제어가 가능하다.

	7	6	5	4	3	2	1	0
RST	-	-	-	-	-	-	-	-
WO								

Bit	Symbol	Description
7	RST	<b>Software Reset</b> W6100 S/W Reset. 모든 Register 가 초기화된다.  0 : W6100 Reset 1 : Normal operation
[6:0]	-	Reserved

#### 4.1.5 SYCR1 (System Config Register 1)

[R=W][0x2005] [0x80]

SYCR1 을 통해 Interrupt Enable 과 System Operation Clock(SYS\_CLK)을 설정할 수 있다.

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

Bit	Symbol	Description
7	IEN	<b>Interrupt Enable</b> INTn Enable bit 이다.  0 : Disable - INTn 은 항상 High 이다. 1 : Enable - 해당 Event 발생시 INTn 은 Low 가 된다.
[6:1]	-	Reserved
0	CLKSEL	<b>System Operation Clock Select</b> SYSR[CHPL] = '0' 인 경우, SYS_CLK 를 선택한다.  0 : 100MHz 1 : 25MHz

#### 4.1.6 TCNTR (Tick Counter Register)

**[RO][0x2016-0x2017][0x0000]**

100us TICK 을 Count 한 값이 설정된다.

#### 4.1.7 TCNTRCLR (TCNTR Clear Register)

**[WO][0x2020][0x00]**

TCNTRCLR 에 임의의 값을 Write 할 경우 TCNTR 을 Clear 한다.

#### 4.1.8 IR (Interrupt Register)

**[RO] [0x2100] [0x00]**

IR 은 WOL(Wake On LAN), Destination Unreachable 과 같은 Event 발생했을 경우 해당 bit 가 1 로 설정된다.

7	6	5	4	3	2	1	0
WOL			UNR6		IPCONF	UNR4	PTERM
RO			RO		RO	RO	RO

Bit	Symbol	Description
7	WOL	<b>WOL(Wake On LAN) Magic Packet</b> 0 : Others 1 : WOL MAGIC Packet 수신
[6:5]	-	Reserved
4	UNR6	<b>Destination IPv6 Port Unreachable</b> 0 : Others 1 : ICMPv6 Destination Port Unreachable Packet 수신

		<i>참조</i> ) 수신한 Unreachable Packet 의 Unreachable IPv6 Address 와 Port Number 는 각각 UIP6R (Unreachable IPv6 Address Register) 과 UPORT6R (Unreachable IPv6 Port Register) 에 저장된다.
3	-	Reserved
2	IPCONF	<b>IP Conflict</b> 0 : Others 1 : IPv4 Address Conflict 발생
1	UNR4	<b>Destination IPv6 Port Unreachable</b> 0 : Others 1: ICMPv4 Destination Port Unreachable Packet 수신  <i>참조</i> ) 수신한 Unreachable Packet 의 Unreachable IP Address 와 Port Number 는 각각 UIPR (Unreachable IP Address Register)과 UPORTR (Unreachable Port Register)에 저장된다.
0	PTERM	<b>PPPoE Terminated</b> 0 : Others 1 : PPPoE 에서 PPPT 또는 LCPT Packet 수신하여 PPPoE 연결이 종료된 경우

#### 4.1.9 SIR (SOCKET Interrupt Register)

**[RO] [0x2101] [0x00]**

SOCKET n Event 가 발생할 경우 '1'로 설정된다.

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

Bit	Symbol	Description
[7:0]	Sn_INT-	<b>Socket n Interrupt</b> 0 : Sn_IR 이 '0'인 경우 1 : Sn_IR 이 '0'이 아닌 경우

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

**[RO] [0x2102] [0x00]**

SLCR(SOCKET-less Command Register)의 해당 Command 가 성공적으로 수행되거나, 수행된 Command 에 대한 Timeout 이 발생하거나, IPv6 Gateway(Router)로부터 ICMPv6 RA Packet 을 수신한 경우 해당 bit 가 설정된다.

7	6	5	4	3	2	1	0
TOUT	ARP4	PING4	ARP6	PING6	NS	RS	RA
RO	RO	RO	RO	RO	RO	RO	RO



Bit	Symbol	Description
7	TOUT	<b>Timeout Interrupt</b> 0 : Others 1 : 임의의 SLCR Command 후 TIMEOUT 발생
6	ARP4	<b>ARP Interrupt</b> 0 : Others 1 : SLCR 의 ARP Command 수행 후 ARP Reply 수신
5	PING4	<b>PING Interrupt</b> 0 : Others 1 : SLCR 의 PING Command 수행 후 PING Reply 수신
4	ARP6	<b>IPv6 ARP Interrupt</b> 0 : Others 1 : SLCR 의 ARP6 Command 수행 후 ARP6 Reply 수신
3	PING6	<b>IPv6 PING Interrupt</b> 0 : Others 1 : SLCR 의 PING6 Command 수행 후 PING6 Reply 수신
2	NS	<b>DAD NS Interrupt</b> 0 : Others 1 : SLCR 의 DAD(Duplicated Address Detection) NS Command 를 수행 후 NA 수신  <i>참조) NS bit 는 IPv6 Address Confliction Detection 에 활용된다.</i>
1	RS	<b>Auto configuration RS Interrupt</b> 0 : Others 1 : SLCR 의 RS Command 수행 후 RA 수신
0	RA	<b>RA Receive Interrupt</b> 0 : Others 1 : IPv6 Gateway 로부터 All-node RA 수신

SLIR[RS] = '1' 또는 SLIR[RA] = '1' 인 경우, RA Packet 에 대한 정보는 다음과 같이 Register 에 저장되며, IPv6 Auto-configuration 에 활용될 수 있다.

- PLR (Prefix Length Register)
- PFR (Prefix Flag Register)
- VLTR (RA Valid Life Time Register)
- PLTR (RA Preferred Life Time Register)
- PAR (Prefix Address Register)

**\* CAUTION :** 수신된 RA message 의 첫 번째 Option 이 Source link-layer address(0x01) 두번째 Option 이 Prefix information Option(0x03) 인 경우만 상기 Register 이 제대로 설정된다. 그렇지 않은 경우는 IPRAW6 Mode SOCKET 을 이용하여 RA message 를 수신하여 해당 정보를 처리한다.

#### 4.1.11 IMR (Interrupt Mask Register)

**[R=W] [0x2104] [0x00]**

IMR 은 IR 에 1:1 대응되는 bit 를 Mask 한다.

7	6	5	4	3	2	1	0
WOL			UNR6		IPCONF	UNR4	PTERM
R=W			R=W		R=W	R=W	R=W

Bit	Symbol	Description
7	WOL	<b>WOL(Wake On LAN) Magic Packet Interrupt Mask</b> 0 : Disable WOL Interrupt 1 : Enable WOL Interrupt
[6:5]	-	Reserved
4	UNR6	<b>Destination Port Unreachable IPv6 Interrupt Mask</b> 0 : Disable UNREACH6 Interrupt 1 : Enable UNREACH6 Interrupt
3	-	Reserved
2	IPCONF	<b>IPv4 Conflict Interrupt Mask</b> 1 : Enable CONFLICT Interrupt 0 : Disable CONFLICT Interrupt
1	UNR4	<b>Destination Port Unreachable Interrupt Mask</b> 1 : Enable UNREACH Interrupt 0 : Disable UNREACH Interrupt
0	PTERM	<b>PPPoE Terminated Interrupt Mask</b> 1 : Enable PPPTERM Interrupt 0 : Disable PPPTERM Interrupt

#### 4.1.12 IRCLR (IR Clear Register)

**[W1] [0x2108] [0x00]**

IR의 특정 bit에 1:1 대응되는 IRCLR Bit를 '1'로 Write 할 경우 해당 IR Bit가 Clear 된다.

7	6	5	4	3	2	1	0
WOL			UNR6		IPCONF	UNR4	PTERM
W1			W1		W1	W1	W1

#### 4.1.13 SIMR (SOCKET Interrupt Mask Register)

**[R=W] [0x2114] [0x00]**

SIMR은 SIR에 1:1로 대응되는 bit를 Mask한다.

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

Bit	Symbol	Description
[7:0]	Sn_INT	<b>Socket n Interrupt Mask</b> 1 : Enable Socket n Interrupt 0 : Disable Socket n Interrupt

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

**[R=W] [0x2124] [0x00]**

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

7	6	5	4	3	2	1	0
TOUT	ARP4	PING4	ARP6	PING6	NS	RS	RA
R=W	R=W	R=W	R=W	R=W	R=W	R=W	R=W

Bit	Symbol	Description
7	TOUT	<b>TIMEOUT Interrupt Mask</b> 1 : Enable TIMEOUT Interrupt 0 : Disable TIMEOUT Interrupt
6	ARP4	<b>ARP Interrupt Mask</b> 1 : Enable ARP4 Interrupt 0 : Disable ARP4 Interrupt
5	PING4	<b>PING Interrupt Mask</b> 1 : Enable PING4 Interrupt 0 : Disable PING4 Interrupt
4	ARP6	<b>IPv6 ARP Interrupt Mask</b> 1 : Enable ARPv6 Interrupt 0 : Disable ARPv6 Interrupt
3	PING6	<b>IPv6 PING Interrupt Mask</b> 1 : Enable PINGv6 Interrupt 0 : Disable PINGv6 Interrupt
2	NS	<b>DAD NS Interrupt Mask</b> 1 : Enable DAD NS Interrupt 0 : Disable DAD NS Interrupt
1	RS	<b>Auto configuration RS Interrupt Mask</b> 1 : Enable AUTO RS Interrupt 0 : Disable AUTO RS Interrupt
0	RA	<b>RA Receive Interrupt Mask</b> 1 : Enable RA RECV Interrupt 0 : Disable RA RECV Interrupt

#### 4.1.15 SLIRCLR (SLIR Clear Register)

**[W1] [0x2128] [0x00]**

SLIRCLR 내의 특정 Bit 를 '1'로 Write 할 경우 1:1 로 매칭되는 SLIR bit 를 Clear 한다.

7	6	5	4	3	2	1	0
TOUT	ARP4	PING4	ARP6	PING6	NS	RS	RA
W1	W1	W1	W1	W1	W1	W1	W1

#### 4.1.16 SLPSR (SOCKET-less Prefer Source IPv6 Address Register)

**[R=W] [0x212C] [0x00]**

SLPSR 은 SLCR(SOCKET-less Command Register)을 통해 전송되는 IPv6 Packet 의 Source Address 를 설정한다.

Value	Symbol	Description
0x00	AUTO	Destination IPv6 Address(Sn_DIP6R (SOCKET n Destination IPv6 Address Register))에 따라 Source IPv6 Address(SIP6)를 선택한다.  SLDIP6R 이 LLA 이면 SIP6 은 LLAR 로 설정 SLDIP6R 이 GUA 이면 SIP6 은 GUAR 로 설정
0x02	LLA	SIP6 은 LLAR 로 고정된다.
0x03	GUA	SIP6 은 GUAR 로 고정된다.

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

**[RW, AC] [0x2130] [0x00]**

SLCR 은 SOCKET 없이 특정 Packet 을 전송하는 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
-	ARP4	PING4	ARP6	PING6	NS	RS	UNA
	RW	RW	RW	RW	RW	RW	RW

Bit	Symbol	Description
7	-	Reserved
6	ARP4	<b>ARP Request Transmission Command</b> 1 : ARP Request 를 전송한다. 0 : Ready
5	PING4	<b>IPv4 PING Request Transmission Command</b> 1 : PING Request 를 전송한다. 0 : Ready
4	ARP6	<b>NS ARP Transmission Command</b> 1 : NS ARP 전송한다. 0 : Ready
3	PING6	<b>IPv6 PING Request Transmission Command</b> 1 : IPv6 PING Request 를 전송한다. 0 : Ready
2	NS	<b>NS Transmission Command for DAD</b> 1 : DAD 용 NS packet 을 전송한다. 0 : Ready

1	RS	<b>Auto configuration RS Transmission Command</b> 1 : RS packet 을 전송한다. 0 : Ready
0	UNA	<b>Unsolicited NA Transmission Command</b> 1 : Unsolicited NA packet 을 전송한다. 0 : Ready

#### 4.1.18 PHYSR (PHY Status Register)

**[RO] [0x3000] [0x00]**

PHYSR 은 PHYCR0(PHY Control Register 0)을 통해 설정된 PHY Operation Mode 와 LINK 상태를 확인한다.

7	6	5	4	3	2	1	0
CAB	-	MODE2	MODE1	MODE0	DPX	SPD	LNK
RO		RO	RO	RO	RO	RO	RO

Bit	Symbol	Description																								
7	CAB	<b>Cable OFF bit</b> 1 : Cable Unplugged 0 : Cable Plugged																								
6	-	Reserved																								
[5:3]	MODE [2:0]	<b>PHY OPMODE</b> <table border="1" data-bbox="561 1169 1273 1496"> <thead> <tr> <th>MODE2</th> <th>MODE1</th> <th>MODE0</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>X</td> <td>X</td> <td>Auto Negotiation</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>100BASE-TX FDX</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>100BASE-TX HDX</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>10BASE-T FDX</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>10BASE-T HDX</td> </tr> </tbody> </table>	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-T FDX	1	1	1	10BASE-T HDX
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-T FDX																							
1	1	1	10BASE-T HDX																							
2	DPX	<b>Flag Duplex bit (When Link Up)</b> 1 : Half Duplex 0 : Full Duplex																								
1	SPD	<b>Flag Speed bit (When Link Up)</b> 1 : 10Mbps 0 : 100Mbps																								
0	LNK	<b>Flag Link bit</b> 1 : Link Up 0 : Link Down																								

#### 4.1.19 PHYRAR (PHY Register Address Register)

**[R=W] [0x3008] [0x00]**

PHYRAR 은 내장된 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]	ADDR [4:0]	PHY Register Address PHY Register Address 를 설정

#### 4.1.20 PHYDIR (PHY Data Input Register)

**[R=W] [0x300C-0x300D] [0x0000]**

PHYDIR 은 PHYRAR 로 지정된 PHY Register 값을 설정한다.

Ex) PHYDIR = 0x1234

PHYDIR0(0x300C)	PHYDIR1(0x300D)
0x34	0x12

#### 4.1.21 PHYDOR (PHY Data Output Register)

**[RO] [0x3010-0x3011] [0x0000]**

PHYDOR 은 PHYRAR 로 지정된 PHY Register 값을 Read 한다.

Ex) PHYDOR = 0x1234

PHYDOR0(0x0042)	PHYDOR1(0x0043)
0x34	0x12

#### 4.1.22 PHYACR (PHY Access Control Register)

**[RW, AC] [0x3014] [0x00]**

PHYACR 은 PHYRAR 로 지정된 PHY Register 를 PHYDIR 이나 PHYDOR 를 통해 Access 한다.

Access 완료 후 PHYACR 은 자동으로 Clear 된다.

Access Type	Value	related Register
Write	0x01	PHYDIR
Read	0x02	PHYDOR

#### 4.1.23 PHYDIVR (PHY Division Register)

**[R=W] [0x3018] [0x01]**

내장 Ethernet PHY 제어를 위해 System Operation Clock(SYS\_CLK)에서 분주된 Clock 을 사용하며, 분주 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

#### 4.1.24 PHYCR0 (PHY Control Register 0)

[WO] [0x301C] [0x00]

PHYCR0 는 SYSR[PHYL] = '0'((PHYLOCKR(PHY Lock Register)가 Unlock 상태)인 경우, Ethernet PHY Operation Mode 를 설정한다. PHYCR0 를 통해 설정된 bits 는 PHYSR [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	<table border="1"> <thead> <tr> <th>MODE2</th> <th>MODE1</th> <th>MODE0</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>x</td> <td>x</td> <td>Auto Negotiation</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>100BASE-TX FDX</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>100BASE-TX HDX</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>10BASE-TX FDX</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>10BASE-TX HDX</td> </tr> </tbody> </table>	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
		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																							

#### 4.1.25 PHYCR1 (PHY Control Register 1)

[R=W] [0x301D] [0x40]

PHYCR1 은 SYSR[PHYL] = '0'((PHYLOCKR(PHY Lock Register)이 Unlock 상태)인 경우, PHY Power down Mode, PHY HW Reset 을 설정한다.

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

Bit	Symbol	Description
7	-	Reserved
6	-	Should be always written by '1'
5	PWDN	<b>PHY Power Down</b> 0 : Disable Power Down Mode SYS_CLK 은 SYCR1[CLKSEL] 설정에 따라 변경된다.

		SYCR1[CLKSEL]	SYS_CLK
		0	100 MHz
		1	25 MHz
		1 : Enable Power Down Mode SYS_CLK 은 25MHz 로 자동 변경된다. <b>참조) 8.4.1 Reset Timing</b>	
4	-	Reserved	
3	TE	10BASE-Te MODE PHYSR[MODE2:MODE0] = '000' 일 경우에만 유효하다.  0 : Disable 10BASE-Te MODE 1 : Enable 10BASE-Te MODE	
[2:1]	-	Reserved	
0	RST	<b>PHY Reset</b> PHY HW Reset 시, SYS_CLK 은 25MHz 로 변경된다. Reset 이 완료되었을 경우, 이 bit 는 Auto-clear 되고, SYS_CLK 은 이전 설정 Clock 으로 복구된다. <b>참조) 8.4.1 Reset Timing</b>  0 : Normal Operation 1 : PHY HW Reset	

#### 4.1.26 NET4MR (Network IPv4 Mode Register)

**[R=W] [0x4000] [0x00]**

NET4MR 은 IPv4 와 관련한 Network Mode 를 설정한다.

7	6	5	4	3	2	1	0
-	-	-	-	UNRB	PARP	RSTB	PB
-	-	-	-	R=W	R=W	R=W	R=W

Bit	Symbol	Description
[7:4]	-	Reserved
3	UNRB	<b>UDP4 Port Unreachable Packet Block</b> 설정되지 않은 Port number 의 SOCKET 으로 UDP4 Packet 을 전송할 경우, Destination Port Unreachable Packet 을 전송하는데 이는 UDP Port scan 방식의 공격대상이 된다. 이를 방지하기 위해서 Unreachable Packet 전송을 Block 할 수 있다.  0 : Unblock 1 : Block
2	PARP	<b>ARpv4 for PINGv4 Reply</b>



		<p>PINGv4 Reply 전송 전에 ARPv4 를 수행할 수 있는데, 이 bit 는 해당 기능의 설정/해제를 지정한다.</p> <p>0 : Disable 1 : Enable</p>
1	RSTB	<p><b>TCP4 RST Packet Block</b></p> <p>지정한 Port number 로 Listen 하는 SOCKET 없는 시스템에 SYN Packet 을 전송할 경우, 해당 시스템은 RST Packet 을 전송하는데 이는 TCP Port Scan 방식의 공격 대상이 될 수 있다. 이를 방지하기 위해서 RST Packet 전송을 Block 할 수 있다.</p> <p>0 : Unblock 1 : Block</p>
0	PB	<p><b>PINGv4 Reply Block</b></p> <p>PINGv4 Request 에 대한 Reply 를 전송하지 않도록 지정할 수 있다.</p> <p>0 : Unblock 1 : Block</p>

#### 4.1.27 NET6MR (Network IPv6 Mode Register)

[R=W] [0x4004] [0x00]

NET6MR 은 IPv6 와 관련한 Network Mode 를 설정한다.

7	6	5	4	3	2	1	0
-	-	-	-	UNRB	PARP	RSTB	PB
-	-	-	-	R=W	R=W	R=W	R=W

Bit	Symbol	Description
[7:4]	-	Reserved
3	UNRB	<p><b>UDP6 Port Unreachable Packet Block</b></p> <p>준비되지 않은 Port number 로 UDP6 Packet 을 전송할 경우, Destination Port Unreachable Packet 을 전송하는데 이는 UDP Port scan 방식의 공격대상이 될 수 있다. 이를 방지하기 위해서 Unreachable Packet 전송을 Block 할 수 있다.</p> <p>0 : Unblock 1 : Block</p>
2	PARP	<p><b>ARPv6 for PINGv6 Reply</b></p> <p>PINGv6 Reply 전송 전에 ARPv6(ND -Neighbor Discovery) Process 을 수행하도록 지정한다.</p> <p>0 : Disable 1 : Enable</p>

1	RSTB	<b>TCP6 RST Packet Block</b> 지정한 Port number 로 Listen 하지 않는 시스템에 SYN Packet 을 전송할 경우, 해당 시스템은 RST Packet 을 전송하는데 이는 TCP Port Scan 방식의 공격 대상이 될 수 있다. 이를 방지하기 위해서 RST Packet 전송을 Block 할 수 있다.  0 : Unblock 1 : Block
0	PB	<b>PINGv6 Reply Block</b> PINGv6 Request 에 대한 Reply 를 전송하지 않도록 지정한다.  0 : Unblock 1 : Block

#### 4.1.28 NETMR (Network Mode Register)

**[R=W] [0x4008] [0x00]**

NETMR 은 각 종 Block mode 와 WOL 을 설정한다.

7	6	5	4	3	2	1	0
-	-	ANB	M6B	-	WOL	IP6B	IP4B
-	-	R=W	R=W	-	R=W	R=W	R=W

Bit	Symbol	Description
[7:6]	-	Reserved Should be always '0'.
5	ANB	<b>IPv6 ALLNODE Block</b> All-Node Multicasting address 를 갖는 PING6-Request 를 block 한다.  0 : Disable 1 : Enable
4	M6B	<b>IPv6 Multicast Block5</b> 자신이 속한 Multicasting Group Address 를 갖는 PING6-Request 를 block 한다.  0 : Disable 1 : Enable
3	-	Reserved Should be always '0'
2	WOL	<b>WOL(Wake On LAN)</b> 0 : Disable 1 : Enable

1	IP6B	<b>IPv6 Packet Block</b> 0 : Unblock 1 : Block - ANB & M6B bit 는 무시된다.
0	IP4B	<b>IPv4 Packet Block</b> 0 : Unblock 1 : Block

#### 4.1.29 NETMR2 (Network Mode Register 2)

**[R=W] [0x4009] [0x00]**

NETMR2 는 PPPoE Mode 를 설정한다.

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

Bit	Symbol	Description
7	DHAS	<b>Destination Hardware Address Selection in ARP/ ND process</b> 0 : Select the Ethernet Frame MAC 1 : Select the ARP Target MAC
[6:1]	-	Reserved
0	PPPoE	<b>PPPoE Mode</b> 0 : PPP Mode disable 1 : PPP Mode enable

#### 4.1.30 PTMR (PPP Link Control Protocol Request Timer Register)

**[R=W] [0x4100] [0x28]**

PTMR 은 LCP Echo Request 주기를 설정한다.

단위는 25ms 이다. PTMR 는 PPPoE Mode 인 경우, 유효하다.

Ex) PTMR = 200 (0xC8),  
200 \* 25ms = 5s

#### 4.1.31 PMNR (PPP Link Control Protocol Magic number Register)

**[R=W] [0x4104] [0x00]**

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

PMNR 은 PPPoE Mode 인 경우, 유효하다.

Ex) PMNR = 0x01

PMNR(0x4104)

0x01

LCP Magic number = 0x01010101

#### 4.1.32 PHAR (PPPoE Server Hardware Address Register on PPPoE) [R=W] [0x4108-0x410D] [0x0000]

PPPoE Destination Hardware Address 를 설정한다.

PHAR 은 PPPoE Mode 인 경우, 유효하다.

Ex) PHAR = “11:22:33:AA:BB:CC”

PHAR0(0x4108)	PHAR1(0x4109)	PHAR2(0x410A)
0x11	0x22	0x33
PHAR3(0x410B)	PHAR4(0x410C)	PHAR5(0x410D)
0xAA	0xBB	0xCC

#### 4.1.33 PSIDR (PPPoE Session ID Register on PPPoE) [R=W] [0x4110-0x4111] [0x0000]

PPPoE Session ID 를 설정한다.

PSIDR 은 PPPoE Mode 인 경우, 유효하다.

Ex) PSIDR = 0x1234

PSIDR0(0x4110)	PSIDR1(0x4111)
0x12	0x34

#### 4.1.34 PMRUR (PPPoE Maximum Receive Unit Register) [R=W] [0x4114-0x4115] [0xFFFF]

PMRUR 은 PPPoE Mode 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 Mode 인 경우, 유효하다.

Ex) PMUR = 1000 (0x03E8)

PMUR0(0x4114)	PMUR1(0x4115)
0x03	0xE8

#### 4.1.35 SHAR (Source Hardware Address Register) [R=W] [0x4120-0x4125] [0x00000\_0000\_0000]

SHAR 은 SYSR[NETL] = '0'(NETLCKR(Network Lock Register)이 Unlock)인 경우, Source Hardware Address 를 설정한다.

Ex) SHAR = “11:22:33:AA:BB:CC”

SHAR0(0x4120)	SHAR1(0x4121)	SHAR2(0x4122)
0x11	0x22	0x33

SHAR3(0x4123)	SHAR4(0x4124)	SHAR5(0x4125)
0xAA	0xBB	0xCC

#### 4.1.36 GAR (Gateway IP Address Register)

**[R=W] [0x4130-0x4133] [0x0000\_0000]**

GAR 은 SYSR[NETL] = '0'(NETLCKR(Network Lock Register)이 Unlock)인 경우, Source Gateway Address 를 설정한다.

Ex) GAR = "192.168.0.1"

GAR0(0x4130)	GAR1(0x4131)	GAR2(0x4132)	GAR3(0x4133)
192 (0xC0)	168 (0xA8)	0 (0x00)	1 (0x01)

#### 4.1.37 SUBR (Subnet Mask Register)

**[R=W] [0x4134-0x4137] [0x0000\_0000]**

SUBR 은 SYSR[NETL] = '0'(NETLCKR(Network Lock Register)이 Unlock)인 경우, Subnet Mask 범위를 설정한다.

Ex) SUBR = "255.255.255.255"

SUBR0(0x4134)	SUBR0(0x4135)	SUBR0(0x4136)	SUBR0(0x4137)
255 (0xFF)	255 (0xFF)	255 (0xFF)	255 (0xFF)

#### 4.1.38 SIPR (IPv4 Source Address Register)

**[R=W] [0x4138-0x413B] [0x0000\_0000]**

SIPR 은 SYSR[NETL] = '0'(NETLCKR(Network Lock Register)이 Unlock)인 경우, Source IP Address 를 설정한다.

Ex) SIPR = "192.168.0.100"

SIPR0(x4138)	SIPR1(0x4139)	SIPR2(0x413A)	SIPR3(0x413B)
192 (0xC0)	168 (0xA8)	0 (0x00)	100(0x64)

#### 4.1.39 LLAR (Link Local Address Register)

**[R=W] [0x4140-0x414F] [0x0000\_0000\_0000\_0000\_0000\_0000\_0000\_0000]**

LLAR 은 SYSR[NETL] = '0'(NETLCKR(Network Lock Register)이 Unlock)인 경우, Link Local Address 를 설정한다.

Ex) LLAR = "FE80::AB:CDEF"

LLAR0(0x4140)	LLAR1(0x4141)	LLAR2(0x4142)	LLAR3(0x4143)
0xFE	0x80	0x00	0x00
LLAR4(0x4144)	LLAR5(0x4145)	LLAR6(0x4146)	LLAR7(0x4147)
0x00	0x00	0x00	0x00

LLAR8(0x4148)	LLAR9(0x4149)	LLAR10(0x414A)	LLAR11(0x414B)
0x00	0x00	0x00	0x00
LLAR12(0x414C)	LLAR13(0x414D)	LLAR14(0x414E)	LLAR15(0x414F)
0x00	0xAB	0xCD	0xEF

#### 4.1.40 GUAR (Global Unicast Address Register)

**[R=W] [0x4150-0x415F] [0x0000\_0000\_0000\_0000\_0000\_0000\_0000\_0000]**

GUAR 은 SYSR[NETL] = '0'(NETLCKR(Network Lock Register)이 Unlock)인 경우, Global Unicast Address 를 설정한다.

Ex) GUAR = "2001::AB:CDEF"

GUAR0(0x4150)	GUAR1(0x4151)	GUAR2(0x4152)	GUAR3(0x4153)
0x20	0x01	0x00	0x00
GUAR4(0x4154)	GUAR5(0x4155)	GUAR6(0x4156)	GUAR7(0x4157)
0x00	0x00	0x00	0x00
GUAR8(0x4158)	GUAR9(0x4159)	GUAR10(0x415A)	GUAR11(0x415B)
0x00	0x00	0x00	0x00
GUAR12(0x415C)	GUAR13(0x415D)	GUAR14(0x415E)	GUAR15(0x415F)
0x00	0xAB	0xCD	0xEF

#### 4.1.41 SUB6R (IPv6 Subnet Prefix Register)

**[R=W] [0x4160-0x416F] [0x0000\_0000\_0000\_0000\_0000\_0000\_0000\_0000]**

SUB6R 은 SYSR[NETL] = '0'(NETLCKR(Network Lock Register)이 Unlock)인 경우, Prefix Mask 를 설정한다

Prefix Length 만큼 최상위 비트(MSB)부터 '1'로 설정하고 나머지 비트들은 모두 '0'으로 설정한다.

Ex) SUB6R = "FFFF:FFFF:FFFF:FFFF::"

PRFXR0(0x4160)	PRFXR1(0x4161)	PRFXR2(0x4162)	PRFXR3(0x4163)
0xFF	0xFF	0xFF	0xFF
PRFXR4(0x4164)	PRFXR5(0x4165)	PRFXR6(0x4166)	PRFXR7(0x4167)
0xFF	0xFF	0xFF	0xFF
PRFXR8(0x4168)	PRFXR9(0x4169)	PRFXR10(0x416A)	PRFXR11(0x416B)
0x00	0x00	0x00	0x00
PRFXR12(0x416C)	PRFXR13(0x416D)	PRFXR14(0x416E)	PRFXR15(0x416F)
0x00	0x00	0x00	0x00

#### 4.1.42 GA6R (IPv6 Gateway Address Register)

**[R=W] [0x4170-0x417F] [0x0000\_0000\_0000\_0000\_0000\_0000\_0000\_0000]**

IPv6 Gateway Address 를 설정한다.

Ex) GA6R = “FE80::FE:DCBA”

GA6R0(0x4170)	GA6R1(0x4171)	GA6R2(0x4172)	GA6R3(0x4173)
0xFE	0x80	0x00	0x00
GA6R4(0x4174)	GA6R5(0x4175)	GA6R6(0x4176)	GA6R7(0x4177)
0x00	0x00	0x00	0x00
GA6R8(0x4178)	GA6R9(0x4179)	GA6R10(0x417A)	GA6R11(0x417B)
0x00	0x00	0x00	0x00
GA6R12(0x417C)	GA6R13(0x417D)	GA6R14(0x417E)	GA6R15(0x417F)
0x00	0xFE	0xDC	0xBA

#### 4.1.43 SLDIP6R (SOCKET-less Destination IPv6 Address Register)

**[R=W] [0x4180-0x418F] [0x0000\_0000\_0000\_0000\_0000\_0000\_0000\_0000]**

SLDIP6R 은 SLCR 에 의해 전송될 Packet 의 Destination IPv6 Address 를 설정한다.

Ex) SLDIP6R = “FE80::AB:CDEF”

SLDIP6R0(0x4180)	SLDIP6R1(0x4181)	SLDIP6R2(0x4182)	SLDIP6R3(0x4183)
0xFE	0x80	0x00	0x00
SLDIP6R4(0x4184)	SLDIP6R5(0x4185)	SLDIP6R6(0x4186)	SLDIP6R7(0x4187)
0x00	0x00	0x00	0x00
SLDIP6R8(0x4188)	SLDIP6R9(0x4189)	SLDIP6R10(0x418A)	SLDIP6R11(0x418B)
0x00	0x00	0x00	0x00
SLDIP6R12(0x418C)	SLDIP6R13(0x418D)	SLDIP6R14(0x418E)	SLDIP6R15(0x418F)
0x00	0xAB	0xCD	0xEF

#### 4.1.44 SLDIPR (SOCKET-less Destination IPv4 Address Register)

**[R=W] [0x418C-0x418F] [0x0000\_0000]**

SLDIPR 은 SLCR 에 의해 전송될 Packet 의 Destination IPv4 Address 를 설정하며

Sn\_DIP6R12 부터 Sn\_DIP6R15 까지 같은 Address 를 사용한다.

Ex) SLDIPR = “192.169.0.21”

SLDIPR0 / SLDIP6R12(0x418C)	SLDIPR1 / SLDIP6R13(0x418D)	SLDIPR2 / SLDIP6R14(0x418E)	SLDIPR3 / SLDIP6R15(0x418F)
192(0xC0)	168(0xA8)	0(0x00)	21(0x15)

#### 4.1.45 SLDHAR (SOCKET-less Destination Hardware Address Register)

**[RO] [0x4190-0x4195] [0x0000\_0000\_0000]**

SLDHAR 은 SLCR[ARP4] 혹은 SLCR[ARP6] Command 에 대한 Reply Packet 을 수신한 경우 (SLIR[ARP4] = '1' 또는 SLIR[ARP6] = '1'), Destination Hardware Address 로 설정된다.

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

SLDHAR0(0x4190)	SLDHAR1(0x4191)	SLDHAR2(0x4192)
0x11	0x22	0x33
SLDHAR3(0x4193)	SLDHAR4(0x4194)	SLDHAR5(0x4195)
0xAA	0xBB	0xCC

#### 4.1.46 PINGIDR (PING ID Register)

**[R=W] [0x4198-0x4199] [0x0000]**

PINGIDR 은 SLCR[PING4] 또는 SLCR[PING6]에 의해 전송될 PING Request Packet 의 ID 를 설정한다.

Ex) PINGIDR = 256 (0x0100)

PINGIDR0(0x4198)	PINGIDR1(0x4199)
0x61	0x00

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

**[R=W] [0x419C-0x419D] [0x0000]**

PINGSEQR 은 SLCR[PING4] 또는 SLCR[PING6]에 의해 전송될 PING Request Packet 의 Sequence Number 를 설정하며, 자동 증가하지 않는다.

Ex) PINGSEQR = 1000 (0x03E8)

PINGSEQR0(0x419C)	PINGSEQR1(0x419D)
0x03	0xE8

#### 4.1.48 UIPR (Unreachable IP Address Register)

**[RO] [0x41A0-0x41A3] [0x0000\_0000]**

UIPR 은 ICMPv4 Unreachable Packet 을 수신한 경우(IR[UNR4] = '1'), 수신한 Packet 의 Destination IPv4 Address 로 설정된다.

Ex) Unreachable IP Address = "192.169.10.10"

UIPR0(0x41A0)	UIPR1(0x41A1)	UIPR2(0x41A2)	UIPR3(0x41A3)
192(0xC0)	168(0xA8)	10(0x0A)	10(0x0A)

#### 4.1.49 UPORTR (Unreachable Port Register)

**[RO] [0x41A4-0x41A5] [0x0000]**



UPOTRR 은 ICMPv4 Unreachable Packet 을 수신한 경우(IR[UNR4] = '1'), 수신한 Packet 의 Destination Port 로 설정된다.

Ex) Unreachable PORT = "3000" (0x0BB8)

UPOTRR0(0x41A4)	UPOTRR1(0x41A5)
0x0B	0xB8

#### 4.1.50 UIP6R (Unreachable IPv6 Address Register)

**[RO] [0x41B0-0x41BF] [0x0000\_0000\_0000\_0000\_0000\_0000\_0000\_0000]**

UIP6R 은 ICMPv6 Unreachable Packet 을 수신한 경우(IR[UNR6] = '1'), 수신한 Packet 의 Destination IP Address 로 설정된다.

Ex) Unreachable IP is "FE80::AB:CDEF"

UIP6R0(0x41B0)	UIP6R1(0x41B1)	UIP6R2(0x41B2)	UIP6R3(0x41B3)
0xFE	0x80	0x00	0x00
UIP6R4(0x41B4)	UIP6R5(0x41B5)	UIP6R6(0x41B6)	UIP6R7(0x41B7)
0x00	0x00	0x00	0x00
UIP6R8(0x41B8)	UIP6R9(0x41B9)	UIP6R10(0x41BA)	UIP6R11(0x41BB)
0x00	0x00	0x00	0x00
UIP6R12(0x41BC)	UIP6R13(0x41BD)	UIP6R14(0x41BE)	UIP6R15(0x41BF)
0x00	0xAB	0xCD	0xEF

#### 4.1.51 UPORT6R (Unreachable IPv6 Port Register)

**[RO] [0x41C0-0x41C1] [0x0000]**

UPOINT6R 은 ICMPv6 Unreachable Packet 을 수신한 경우(IR[UNR6] = '1'), 수신한 Packet 의 Destination Port 로 설정된다.

Ex) Unreachable PORT is "3000" (0x0BB8)

UPOINT6R0(0x41C0)	UPOINT6R1(0x41C1)
0x0B	0xB8

#### 4.1.52 INTPTMR (Interrupt Pending Time Register)

**[RW][0x41C5-0x41C6][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(0x41C5)	INTPTMR1(0x41C6)
0x03	0xEB

#### 4.1.53 PLR (Prefix Length Register)

**[RO] [0x41D0] [0x00]**

RA(Router Advertisement) Packet 을 수신한 경우(SLIP[RS] = '1' 혹은 SLIP[RA] = '1'), 수신된 RA 의 Prefix Information Option 내의 Prefix Length Field 로 설정된다.

Ex) RA Prefix Length = 0x10

PLR(0x41D0)
0x10

#### 4.1.54 PFR (Prefix Flag Register)

**[RO] [0x41D4] [0x00]**

RA(Router Advertisement) Packet 을 수신한 경우(SLIP[RS] = '1' 혹은 SLIP[RA] = '1'), 수신된 RA 의 Prefix Information Option 내의 Prefix Flag Field 로 설정된다.

Ex) Flag = 0xC0

PFR(0x41D4)
0xC0

#### 4.1.55 VLTR (Valid Life Time Register)

**[RO] [0x41D8-0x41DB] [0x0000\_0000]**

RA(Router Advertisement) Packet 을 수신한 경우(SLIP[RS] = '1' 혹은 SLIP[RA] = '1'), 수신된 RA 의 Prefix Information Option 내의 Valid Life Time Field 로 설정된다.

Ex) Valid Life Time = 2592000

VLTR0(0x41D8)	VLTR1(0x41D9)	VLTR2(0x41DA)	VLTR3(0x41DB)
0x00	0x27	0x8D	0x00

#### 4.1.56 PLTR (Preferred Life Time Register)

**[RO] [0x41DC-0x41DF] [0x0000\_0000]**

RA(Router Advertisement) Packet 을 수신(SLIP[RS] = '1' 혹은 SLIP[RA] = '1')한 경우, 수신된 RA 의 Prefix Information Option 내의 Preferred Life Time Field 로 설정된다.

Ex) Preferred Life Time = 604800

PLTR0(0x41DC)	PLTR1(0x41DD)	PLTR2(0x41DE)	PLTR3(0x41DF)
0x00	0x09	0x3A	0x80

#### 4.1.57 PAR (Prefix Address Register)

**[RO] [0x41E0-0x41EF] [0x0000\_0000\_0000\_0000\_0000\_0000\_0000]**

RA(Router Advertisement) Packet 을 수신(SLIP[RS] = '1' 혹은 SLIP[RA] = '1')한 경우, 수신된 RA 의 Prefix Information Option 내의 Prefix Address Field 로 설정된다.

Ex) Prefix is "2001:2b8:10:1::"

PAR0(0x41E0)	PAR1(0x41E1)	PAR2(0x41E2)	PAR3(0x41E3)
0x20	0x01	0x02	0xb8
PAR4(0x41E4)	PAR5(0x41E5)	PAR6(0x41E6)	PAR7(0x41E7)
0x00	0x10	0x00	0x01
PAR8(0x41E8)	PAR9(0x41E9)	PAR10(0x41EA)	PAR11(0x41EB)
0x00	0x00	0x00	0x00
PAR12(0x41EC)	PAR13(0x41ED)	PAR14(0x41EE)	PAR15(0x41EF)
0x00	0x00	0x00	0x00

#### 4.1.58 ICMP6BLKR (ICMPv6 Block Register)

**[R=W] [0x41F0] [0x00]**

PING6, Multicast Listener Discovery(MLD) Query, Router Advertisement(RA), Neighbor Advertisement (NA), Neighbor Solicitation(NS)와 같은 ICMPv6 Packet 수신을 선택적으로 Block 한다. Block 한 Packet 들은 IPRAW6 SOCKET 을 통해 수신할 수 있다.

7	6	5	4	3	2	1	0
			PING6	MLD	RA	NA	NS
			R=W	R=W	R=W	R=W	R=W

Bit	Symbol	Description
[7:5]	-	Reserved
4	PING6	<b>ICMPv6 Echo Request Block</b> 1 : Block Echo request Packet 0 : Normal Operation
3	MLD	<b>ICMPv6 Multicast Listener Discovery(MLD) Query Block</b> 1 : Block Multicast Listener Discovery Query Packet 0 : Normal Operation
2	RA	<b>ICMPv6 Router Advertisement Block</b> 1 : Block Router Advertisement Packet 0 : Normal Operation
1	NA	<b>ICMPv6 Neighbor Advertisement Block</b> 1 : Block Neighbor Advertisement Packet 0 : Normal Operation
0	NS	<b>ICMPv6 Neighbor Solicitation Block</b> 1 : Block Neighbor Solicitation Packet 0 : Normal Operation

#### 4.1.59 CHPLCKR (Chip Lock Register)

**[WO] [0x41F4] [0x00]**

SYSR[CHPL]을 설정한다.

SYSR[CHPL]이 'Unlock' 인 경우, SYCR0, SYCR1 을 설정할 수 있다.

Unlock	Lock
0xCE	Others

#### 4.1.60 NETLCKR (Network Lock Register)

**[WO] [0x41F5] [0x00]**

SYSR[NETL]을 설정한다.

SYSR[NETL]이 'Unlock'이면 Network Configuration Registers(SHAR, GAR, SUBR, SIPR, LLAR, GUAR, SUB6R)를 설정할 수 있다.

Unlock	Lock
0x3A	0xC5

#### 4.1.61 PHYLCKR (PHY Lock Register)

**[WO] [0x41F6] [0x00]**

SYSR[PHYL]을 설정한다.

SYSR[PHYL]이 'Unlock'인 경우, PHYCR0, PHYCR1 을 설정할 수 있다.

Unlock	Lock
0x53	Others

#### 4.1.62 RTR (Retransmission Time Register)

**[R=W] [0x4200-0x4201] [0x07D0]**

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

단위는 100us 이다.

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

**참조** [6.7 Retransmission](#)

Ex) RTR = 5000 (0x1388)

5000\*100us = 0.5s

RTR0(0x4200)	RTR1(0x4201)
0x13	0x88

#### 4.1.63 RCR (Retransmission Count Register)

**[R=W] [0x4204] [0x08]**

RCR 은 Sn\_RCR(SOCKET n Retransmission Count Register)의 초기값을 설정한다.

RTR(Retransmission)과 함께 SOCKET 에 의한 Packet (ARP/ND, ICMPv6, TCP) 재전송에 관여한다.

**참조) 6.7 Retransmission**

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

**[R=W] [0x4208-0x4209] [0x07D0]**

SLRTR 은 SLCR 의 Retransmission Time 을 설정한다. 단위는 100us 이다. SLCR 에 의해 전송된 Request Packet 에 대한 응답이 없는 경우 재전송이 발생하며, 재전송 회수가 SLRCR (SOCKET-less Retransmission Count Register)에 지정된 값을 초과하면 Timeout(SLIR [TOUT] = '1')이 발생한다.

**참조) 6.7 Retransmission**

Ex) SLRTR = 5000 (0x1388),  
5000 \* 100us = 0.5s

SLRTR0(0x4208)	SLRTR1(0x4209)
0x013	0x88

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

**[R=W] [0x420C] [0x00]**

SLRCR 은 SLCR 의 Retransmission Counter 를 설정한다.

Retransmission Counter 가 SLRCR 을 초과하면 SLIR[TOUT]는 '1'이 된다.

**참조) 6.7 Retransmission**

#### 4.1.66 SLHOPR (Hop limit Register)

**[RW] [0x420F] [0x80]**

SLCR 에 의해 전송되는 ND Messages(NS, NA)의 HOP 을 설정한다.

Ex) SLHOPR = 128

SLHOPR(0x420F)
0x80 (128)

## 4.2 SOCKET Register

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

[R=W] [0x0000] [0x00]

Sn\_MR 은 SOCKET n 의 Mode 와 Option 을 설정한다.

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

7	6	5	4	3	2	1	0
MULTI/ MF	BRDB/ FPSH	ND/ MC/ SMB/ MMB	UNIB/ MMB6	P3	P2	P1	P0
R=W	R=W	R=W	R=W	R=W	R=W	R=W	R=W

Bit	Symbol	Description
7	MULTI/ MF	<p><b>MULTI : Multicast Mode</b> Sn_MR[3:0]이 UDP4, UDP6, UDPD 인 경우, 유효하다. <i>참조</i> 6.3.3 UDP Multicast 0 : Disable UDP Multicast 1 : Enable UDP Multicast</p> <p>UDP6, UDPD 인 경우 IPv6 Multicasting 만 지원한다.</p> <p><b>MF : MAC Filter Enable</b> Sn_MR[3:0]이 MACRAW Mode 인 경우, 유효하다. 0 : Disable MAC Filter (모든 Packet 수신) 1 : Enable MAC Filter (Multicast, Broadcast 와 패킷의 Destination MAC 이 SHAR 값과 같은 경우 수신)</p>
6	BRDB/ FPSH	<p><b>BRDB : Broadcast Block</b> Sn_MR[3:0]이 UDP4, UDP6, UDPD, MACRAW Mode 인 경우, 유효하다. <i>참조</i> 6.3.2 UDP Broadcast 0 : UnblockUDP Broadcast 1 : Block UDP Broadcast</p> <p><b>FPSH : Force Push flag</b> Sn_MR[3:0]이 TCP4, TCP6, TCPD 인 경우, 전송되는 모든 DATA Packet 의 PSH flag 를 설정한다. 0 : No Force PSH flag (SEND Command 에 의해 전송되는 DATA Packet 중 마지막 DATA Packet 에만 PSH flag 설정) 1 : Force PSH flag</p>
5	ND/ MC/	<p><b>ND : No Delayed ACK</b> Sn_MR[3:0]이 TCP4, TCP6, TCPD 인 경우, 유효하다. 0 : Delayed ACK (Sn_RTR 시간 이 후 ACK Packet 전송)</p>

	SMB/ MMB	<p>1 : No Delayed ACK (DATA Packet 수신 시 ACK Packet 전송)</p> <p><i>참조) ND 설정과 무관하게 Sn_CR[RECV] 수행 이후, SOCKET n Window Size 가 MSS 보다 작은 경우는 ACK Packet 을 곧바로 전송 한다.</i></p> <p><b>MC : Multicast IGMP Version</b> Sn_MR[3:0] = UDP4 이고 Sn_MR[MULTI] = '1' 인 경우, 유효하다. 0 : Using IGMP version 2 1 : Using IGMP version 1</p> <p><b>SMB : UDP6 Solicited Multicast Block</b> Sn_MR[3:0] = UDP6 or UDPD 인 경우, 유효하다. W6100 의 Solicited Multicast Address 로 송신된 Packet 의 수신여부를 결정한다. 0 : Unblock Solicited Multicast 1 : Block Solicited Multicast</p> <p><b>MMB : UDP4 Multicast Block in MACRAW Mode</b> Sn_MR [3:0] = MACRAW 이고 Sn_MR [MF] = '1'인 경우, 유효하다. 0 : Unblock IPv4 Multicast 1 : Block IPv4 Multicast</p>																		
4	UNIB/ MMB6	<p><b>UNIB : Unicast Block</b> Sn_MR[3:0]이 UDP4, UDP6, UDPD 이고 MULTI = 1 인 경우, 유효하다. <i>참조) 6.3.5.2 UDP Block</i> 0 : Unblock UDP Unicast 1 : Block UDP Unicast</p> <p><b>MMB6 : UDP6 Multicast Block in MACRAW Mode</b> Sn_MR [3:0] = MACRAW 이고 Sn_MR [MF] = '1'인 경우, 유효하다. 0 : Unblock IPv6 Multicast 1 : Block IPv6 Multicast</p>																		
[3:0]	P[3:0]	<p><b>P[3:0] : Protocol Mode</b> SOCKET n Protocol Mode 를 설정한다.</p> <table border="1" data-bbox="699 1592 1150 2078"> <thead> <tr> <th>P[3:0]</th> <th>Protocol Mode</th> </tr> </thead> <tbody> <tr> <td>0000</td> <td>Socket Closed</td> </tr> <tr> <td>0001</td> <td>TCP4</td> </tr> <tr> <td>0010</td> <td>UDP4</td> </tr> <tr> <td>0011</td> <td>IPRAW4</td> </tr> <tr> <td>0111</td> <td>MACRAW</td> </tr> <tr> <td>1001</td> <td>TCP6</td> </tr> <tr> <td>1010</td> <td>UDP6</td> </tr> <tr> <td>1011</td> <td>IPRAW6</td> </tr> </tbody> </table>	P[3:0]	Protocol Mode	0000	Socket Closed	0001	TCP4	0010	UDP4	0011	IPRAW4	0111	MACRAW	1001	TCP6	1010	UDP6	1011	IPRAW6
P[3:0]	Protocol Mode																			
0000	Socket Closed																			
0001	TCP4																			
0010	UDP4																			
0011	IPRAW4																			
0111	MACRAW																			
1001	TCP6																			
1010	UDP6																			
1011	IPRAW6																			

		1101	TCP Dual (TCPD)
		1110	UDP Dual (UDPD)
MACRAW Mode 는 SOCKET 0 인 경우, 유효하다.			

#### 4.2.2 Sn\_PSR (SOCKET n Prefer Source IPv6 Address Register)

**[RW] [0x0004] [0x00]**

Sn\_PSR 은 SOCKET n 의 Source IPv6 Address(SIP6)를 결정한다.

Value	Symbol	Description
0x00	AUTO	Destination IPv6 Address(Sn_DIP6R (SOCKET n Destination IPv6 Address Register))에 따라 Source IPv6 Address(SIP6)를 선택한다.  Sn_DIP6R 이 LLA 이면 SIP6 은 LLAR 로 설정 Sn_DIP6R 이 GUA 이면 SIP6 은 GUAR 로 설정
0x02	LLA	SIP6 은 LLAR 로 고정된다.
0x03	GUA	SIP6 은 GUAR 로 고정된다.

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

**[RW,AC] [0x0010] [0x00]**

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

Value	Symbol	Description																				
0x01	OPEN	<p><b>OPEN Command</b> SOCKET n 을 OPEN 한다. Sn_MR[3:0] 설정에 따른 Command 수행 결과는 Sn_SR(SOCKET n Status Register)로 확인한다.</p> <table border="1" data-bbox="590 1512 1289 2049"> <thead> <tr> <th>Sn_MR[3:0]</th> <th>Sn_SR</th> </tr> </thead> <tbody> <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>S0_MR_MACRAW('0111')</td> <td>SOCK_MACRAW(0x42)</td> </tr> <tr> <td>Sn_MR_TCP6('1001')</td> <td>SOCK_INIT(0x13)</td> </tr> <tr> <td>Sn_MR_UDP6('1010')</td> <td>SOCK_UDP(0x22)</td> </tr> <tr> <td>Sn_MR_IPRAW6('1011')</td> <td>SOCK_IPRAW6(0x33)</td> </tr> <tr> <td>Sn_MR_TCPD('1101')</td> <td>SOCK_INIT(0x13)</td> </tr> </tbody> </table>	Sn_MR[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)	S0_MR_MACRAW('0111')	SOCK_MACRAW(0x42)	Sn_MR_TCP6('1001')	SOCK_INIT(0x13)	Sn_MR_UDP6('1010')	SOCK_UDP(0x22)	Sn_MR_IPRAW6('1011')	SOCK_IPRAW6(0x33)	Sn_MR_TCPD('1101')	SOCK_INIT(0x13)
Sn_MR[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)																					
S0_MR_MACRAW('0111')	SOCK_MACRAW(0x42)																					
Sn_MR_TCP6('1001')	SOCK_INIT(0x13)																					
Sn_MR_UDP6('1010')	SOCK_UDP(0x22)																					
Sn_MR_IPRAW6('1011')	SOCK_IPRAW6(0x33)																					
Sn_MR_TCPD('1101')	SOCK_INIT(0x13)																					



		Sn_MR_UDPD('1110')	SOCK_UDP(0x22)
0x02	LISTEN	<b>LISTEN Command</b> Sn_MR[3:0]이 TCP4, TCP6, TCPD 이고 Sn_SR = SOCK_INIT 인 경우, Destination 의 접속을 대기한다. <b>참조) 6.2.1 TCP SERVER : LISTEN</b>	
0x04	CONNECT	<b>TCP CONNECT Command</b> Sn_MR[3:0]이 TCP4, TCPD 이고 Sn_SR = SOCK_INIT 인 경우, Destination 에게 접속을 요청한다. <b>참조) 0 TCP CLIENT: CONNECT</b>	
0x84	CONNECT6	<b>TCP6 CONNECT Command</b> Sn_MR[3:0]이 TCP6, TCPD Mode 이고 Sn_SR = SOCK_INIT 인 경우, Destination 에게 접속을 요청한다.	
0x08	DISCON	<b>TCP DISCON Command</b> Sn_MR[3:0]이 TCP4, TCP6, TCPD 이고, Sn_SR = SOCK_ESTABLISHED 또는 Sn_SR = SOCK_CLOSE_WAIT 인 경우, Destination 에게 접속 해제를 요청한다. 즉, 접속중인 Destination 에게 Disconnect-Request (FIN Packet)를 전송한다. <b>참조) 6.2.1 TCP SERVER: Disconnected (Active Close)</b>	
0x10	CLOSE	<b>CLOSE Command</b> SOCKET n 을 CLOSE 한다. Sn_MR[3:0] 설정과 상관 없이 Sn_SR 은 SOCK_CLOSED 로 변경된다. <b>* CAUTION : TCP 인 경우, FIN Packet 전송 없이 강제 CLOSE 된다.</b>	
0x20	SEND *	<b>SEND Command</b> Sn_MR[3:0]이 TCP4, TCP6, TCPD, UDP4, UDPD, IPRAW4, MACRAW 인 경우, DATA 를 전송한다.	
0xA0	SEND6 *	<b>IPv6 SOCKET SEND Command</b> Sn_MR[3:0]이 UDP6, UDPD Mode, IPRAW6 Mode 인 경우, DATA 를 전송한다.	
0x22	SEND_KEEP	<b>TCP SEND_KEEP Command</b> Sn_MR[3:0]이 TCP4, TCP6, TCPD Mode 이고 최소 1 Byte 이상의 DATA 를 전송한 후 수행 가능하다. SEND_KEEP Command 는 Destination 에게 Keep alive(KA) Packet 을 송신하여 Connection 이 유효한지 확인한다. 만약 Destination 이 응답이 없는 경우 설정된 Retransmission Time 이후 Sn_IR [TIMEOUT] = '1' 이고 되고, Sn_SR = SOCK_CLOSED 이 된다. <b>참조) 6.2.4.2 Keep Alive</b>	
0x40	RECV	<b>SOCKET RECV Command</b>	

		<p>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><b>참조</b> 4.2.28 Sn_RX_RSR (SOCKET n RX Received Size Register), 0 Sn_RX_WR (SOCKET n RX Write Pointer Register), 4.2.29 Sn_RX_RD (SOCKET n RX Read Pointer Register)</p>
--	--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

- \* Sn\_TX\_WR (SOCKET n TX Write Point Register)과 Sn\_TX\_RD (SOCKETn Rx Read Pointer Register)로 계산된 크기의 DATA 를 전송하며, 전송 DATA 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.
- \* TCP4, TCP6, TCPD, UDP4, UDP6, UDPD Mode 인 경우, 전송하려는 DATA 가 MSS(Maximum Segment Size)를 초과할 경우 자동으로 MSS 단위로 나누어 전송한다.
- \* IPRAW4, IPRAW6, MACRAW Mode 인 경우, HOST 는 전송하려는 DATA 를 MSS 단위로 직접 나누어 전송해야 한다. TCP4, TCP6, TCPD Mode 인 경우, DATA 를 Destination 에게 성공적으로 전송하지 못한 경우 (Destination 으로부터 ACK 를 수신하지 못한 경우), Sn\_IR[TIMEOUT] = '1' 이 되고 Sn\_SR = SOCK\_CLOSED 이 된다. TCP4, TCP6, TCPD, UDP4, UDP6, UDPD, IPRAW4, IPRAW6, MACRAW Mode 인 경우, Sn\_IR [SENDOK] = '1' 이후 Sn\_TX\_FSR 은 전송한 DATA Size 만큼 증가한다.

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

[RO] [0x0020] [0x00]

Sn\_IR 은 SOCKET n 의 상태변화나 Sn\_CR 수행 결과를 확인한다.

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

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

Bit	Symbol	Description
[7:5]	-	Reserved
4	SENDOK	<b>SEND OK Interrupt</b> Sn_CR[SEND]이 완료된 경우, '1'로 설정된다.
3	TIMEOUT	<b>TIMEOUT Interrupt</b> ARP/ND 또는 TCP 통신과정에서 재전송 횟수가 Sn_RCR(SOCKET n Retrans mission Count Register)을 초과한 경우, '1'로 설정된다.
2	RECV	<b>RECEIVED Interrupt</b> SOCKET n RX Buffer 에 DATA 가 수신되거나, Sn_CR [RECV] 수행 후 DATA 가 남아있는 경우, '1'로 설정된다.
1	DISCON	<b>DISCONNECTED Interrupt</b>

		Sn_CR[DISCON]을 통해 FIN Packet 을 전송 후 Destination 으로부터 ACK 응답 받은 경우나, 혹은 Destination 으로부터 FIN Packet 을 수신하거나 RST Packet 을 수신한 경우, '1'로 설정된다.
0	CON	<b>CONNECTED Interrupt</b> Sn_CR[CONNET] 또는 Sn_CR[CONNECT6]을 통해 TCP 접속이 완료된 경우, 혹은 Destination 으로부터의 SYN Packet 수신하고 접속이 완료된 경우에 '1'로 설정된다.

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

**[R=W] [0x0024] [0xFF]**

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

7	6	5	4	3	2	1	0
-	-	-	SENDOK	TIMEOUT	RECV	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	RECV	Sn_IR[RECV] Interrupt Mask
1	DISCON	Sn_IR[DISCON] Interrupt Mask
0	CON	Sn_IR[CON] Interrupt Mask

#### 4.2.6 Sn\_IRCLR (Sn\_IR Clear Register)

**[WO] [0x0028] [0x00]**

Sn\_IMR 은 Sn\_IR 의 1:1 대응되는 bit 를 Clear 한다.

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

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

[RO] [0x0030] [0x00]

Sn\_SR 은 Sn\_CR 수행 또는 DATA 송수신에 의해 변경되는 SOCKET n Status 를 확인한다.

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 Destination 의 접속을 기다리는 상태
0x17	SOCK_ESTABLISHED	SOCKET 이 TCP Mode 이고 Destination 과 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.
0x33	SOCK_IPRAW6	SOCKET 이 IPRAW6 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_SYNRECV	Connect-Request 를 수신한 상태.
0x18	SOCK_FIN_WAIT	SOCKET 이 Closing 되는 상태.
0x1B	SOCK_TIME_WAIT	
0x1D	SOCK_LAST_ACK	

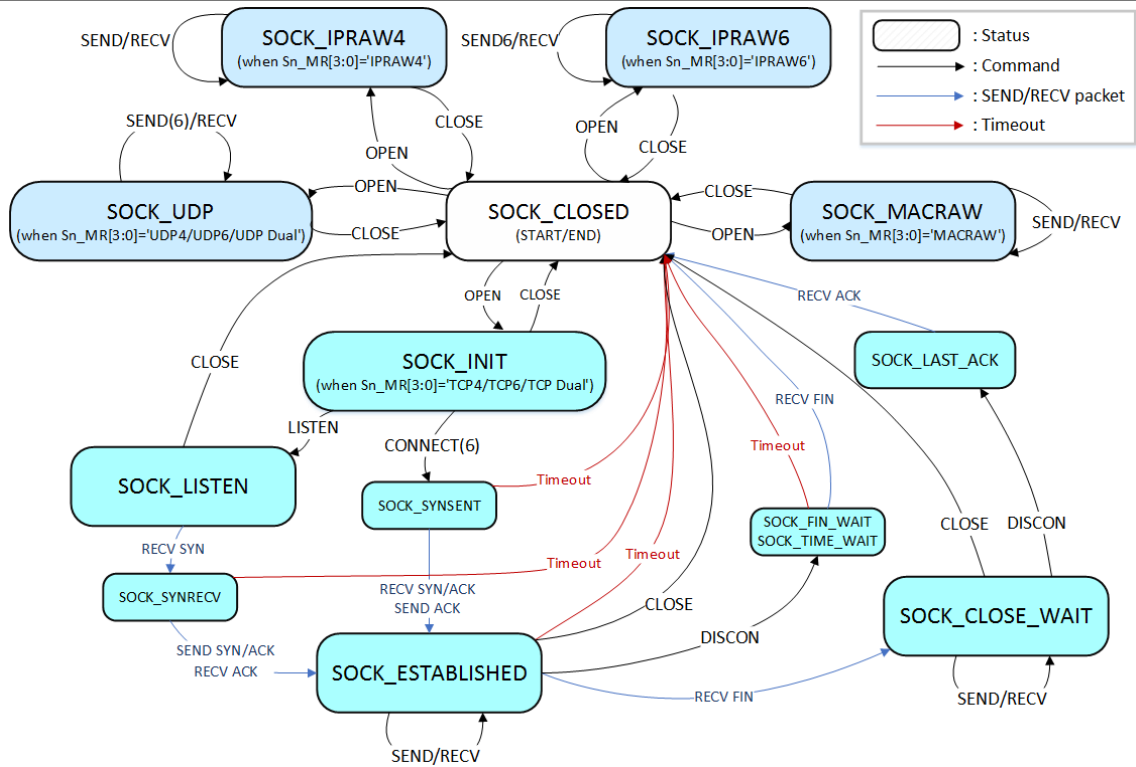


Figure 4 State Diagram

#### 4.2.8 Sn\_ESR (SOCKET n Extension Status Register)

**[RO] [0x0031] [0x00]**

Sn\_ESR 은 Sn\_MR[3:0]이 TCP4, TCP6, TCPD 인 경우, SOCKET n Extension Status 를 확인 한다.

7	6	5	4	3	2	1	0
-	-	-	-	-	TCPM	TCPOP	IP6T
-	-	-	-	-	RO	RO	RO

Bit	Symbol	Description
[7:3]	-	Reserved
2	TCPM	<b>TCP Mode</b> Sn_MR [3:0]이 TCPD 이고 Destination 과 접속한 경우, 접속된 실제 TCP Mode 를 확인한다.  0 : TCP4 1 : TCP6
1	TCPOP	<b>TCP Operation Mode</b> 0 : TCP Client 1 : TCP Server
0	IP6T	<b>IPv6 address type</b> Sn_MR[3:0]이 TCP6, TCPD 이고, Destination 에게 전송한 Packet 의 Source IPv6 Address 를 확인한다.  0 : LLA 1 : GUA

#### 4.2.9 Sn\_PNR (SOCKET n IP Protocol Number Register)

[R=W] [0x0100] [0x0000]

SOCKET 이 IPRAW4 또는 IPRAW6 Mode 인 경우, Sn\_PNR 은 각각의 Mode 에 따라 SOCKET 이 송수신하는 IPv4 의 상위 Protocol Number 또는 IPv6 의 Next Header 를 설정한다.

Sn\_PNR 설정 값은 [Table 7](#) 및 [IANA\\_Protocol\\_Numbers](#) 를 참고하라. 단, Sn\_PNR 은 TCP(0x06)나 UDP(0x11)는 설정될 수 없다.

#### 4.2.10 Sn\_TOSR (SOCKET n IP Type of Service Register)

[R=W] [0x0104] [0x00]

Sn\_TOSR 은 SOCKET n 의 IPv4 Header TOS field 를 설정한다.

참조) [IANA\\_IP\\_Parameters](#)

**\* CAUTION** IPv6 Header 의 Traffic Class 와 Flow Label Field 설정은 지원하지 않으며, 0 값으로 고정된다.

#### 4.2.11 Sn\_TTLR (SOCKET n IP Time To Live Register)

[R=W] [0x0108] [0x80]

Sn\_TTLR 은 SOCKET n 의 IPv4 header TTL field 나 IPv6 의 HOP Limit field 를 설정한다.

참조) [IANA\\_IP\\_Parameters](#)

#### 4.2.12 Sn\_FRGR (SOCKET n Fragment Offset in IP Header Register)

[R=W] [0x010C-0x010D] [0x4000]

Sn\_FRGR 은 SOCKET n 의 IPv4 Header Fragment field 를 설정한다.

**\* CAUTION** Fragment field 는 설정이 가능하나, IP Fragment 기능을 수행하지는 않으며, 수신된 Fragment Packet 역시 처리되지 않는다.

Ex) S0\_FRGR0 = 0x0000 (Don't Fragment)

SO_FRGR0(0x010C)	SO_FRGR1(0x010D)
0x00	0x00

#### 4.2.13 Sn\_MSSR (SOCKET n Maximum Segment Size Register)

[RW] [0x0110-0x0111] [0x0000]

Sn\_MSSR 은 Sn\_CR[OPEN] 이전에 SOCKET n MSS 를 설정한다. 최대 MSS 를 초과할 수 없으며 초과할 경우, 자동으로 최대 MSS 로 설정된다. 아래는 Sn\_MR[3:0]과 NETMR2[PPPoE] 설정에 따른 최대 MSS 를 나타낸 표이다.

Sn_MR[3:0]	Normal Range (NETMR2[PPPoE]='0')	PPPoE Range (NETMR2[PPPoE]='1')
TCP	1-1460	1-1452
TCP6	1-1440	1-1432

UDP	1-1472	1-1464
UDP6	1-1452	1-1444
IPRAW	1-1480	1-1472
IPRAW6	1-1460	1-1452
MACRAW	1-1514	

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

SO_MSSR0(0x0110)	SO_MSSR1(0x0111)
0x05	0xB4

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

**[R=W] [0x0114-0x0115] [0x0000]**

SOCKET n Source Port Number 를 설정한다.

Ex) SO\_PORTR = 5000 (0x1388)

SO_PORTR0(0x0114)	SO_PORTR1(0x0115)
0x013	0x88

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

**[RW] [0x0118-0x011D] [0x0000\_0000\_0000]**

Sn\_DHAR 은 Sn\_MR[3:0]이 TCP4, TCP6, TCPD Mode 이고 Destination 과 접속한 경우 (Sn\_SR = SOCK\_ESTABLISHED), Destination Hardware Address 가 기록된다.

Sn\_DHAR 은 Sn\_MR[3:0]이 UDP4, UDP6 이고 Sn\_MR[MULTI] = '1' 인 경우, Multicast Group 의 Hardware Address 가 설정된다. [참조\) 6.3.3 UDP Multicast](#)

Ex) SO\_DHAR = "11:22:33:AA:BB:CC"

SO_DHAR0(0x0118)	SO_DHAR1(0x0119)	SO_DHAR2(0x011A)
0x11	0x22	0x33
SO_DHAR3(0x011B)	SO_DHAR4(0x011C)	SO_DHAR5(0x011D)
0xAA	0xBB	0xCC

#### 4.2.16 Sn\_DIPR (SOCKET n Destination IPv4 Address Register)

**[RW] [0x0120-0x0123] [0x0000\_0000]**

Sn\_DIPR 은 Sn\_MR[3:0]에 설정된 Protocol type 에 따라 아래와 같은 기능을 수행한다.

Sn_MR[3:0]	Sn_MR[MULTI]	Sn_DIPR
TCP4	Don't care	Set or Get Destination IPv4 Address

TCPD	Don't care	Set Destination IPv4 Address
UDP4	0	
UPPD	Don't care	
IPRAW4	Don't care	
UDP4	1	Set Multicast Group IPv4 Address

**참조) 6.3.3 UDP Multicast**

Ex) SO\_DIPR = "192.168.0.11"

SO_DIPR0(0x0120)	SO_DIPR1(0x0121)	SO_DIPR2(0x0122)	SO_DIPR3(0x0123)
192 (0xC0)	168 (0xA8)	0 (0x00)	11 (0x0B)

#### 4.2.17 Sn\_DIP6R (SOCKET n Destination IPv6 Address Register)

**[RW] [0x0130-0x013F] [0x0000\_0000\_0000\_0000\_0000\_0000\_0000\_0000]**

Sn\_DIP6R 은 Destination IPv6 Address 지정하는데, Protocol Type 에 따라서 그 의미가 달라진다.

Sn_MR[3:0]	Sn_MR[MULTI]	Sn_DIP6R
TCP6	Don't care	Set or Get Destination IPv6 Address
TCPD	Don't care	
UDP6	0	Set Destination IPv6 Address
UPPD	Don't care	
IPRAW6	Don't care	
UDP6	1	Set Multicast Group IPv6 Address

**참조) 6.3.3 UDP Multicast**

Ex) Destination IP is "FE80::AB:CDEF"

SO_DIP6R0(0x0130)	SO_DIP6R1(0x0131)	SO_DIP6R2(0x0132)	SO_DIP6R3(0x0133)
0xFE	0x80	0x00	0x00
SO_DIP6R4(0x0134)	SO_DIP6R5(0x0135)	SO_DIP6R6(0x0136)	SO_DIP6R7(0x0137)
0x00	0x00	0x00	0x00
SO_DIP6R8(0x0138)	SO_DIP6R9(0x0139)	SO_DIP6R10(0x013A)	SO_DIP6R11(0x013B)
0x00	0x00	0x00	0x00
SO_DIP6R12(0x013C)	SO_DIP6R13(0x013D)	SO_DIP6R14(0x013E)	SO_DIP6R15(0x013F)
0x00	0xAB	0xCD	0xEF

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

**[R=W] [0x0140-0x0141] [0x0000]**



Sn\_DPORTR 은 Destination Port 를 지정하는데, Protocol Type 에 따라서 그 의미가 달라진다.

Sn_MR[3:0]	Sn_MR[MULTI]	Sn_DPORTR
TCP4	Don't care	Set or Get Destination Port
TCP6	Don't care	
TCPD	Don't care	
UDP4	0	Set Destination Port
UDP6	0	
UPPD	Don't care	
IPRAW4	Don't care	
IPRAW6	Don't care	
UDP4	1	Set Multicast Group Port
UDP6	1	

TCP4, TCP6, TCPD Mode 인 경우, 접속할 Destination Port 를 설정하거나, 접속된 Destination Port 를 확인한다.

Sn\_DPORTR 은 UDP4, UDP6, UPPD, IPRAW6 Mode 인 경우, 전송할 Destination Port 를 설정한다.

Sn\_DPORTR 은 UDP4, UDP6 Multicast 인 경우, Multicast Group Port 를 설정한다.

**참조) 6.3.3 UDP Multicast**

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

S0_DPORTR0(0x0140)	S0_DPORTR1(0x0141)
0x13	0x88

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

**[R=W] [0x0144] [0x00]**

Sn\_MR2 는 Sn\_MR 과 함께 SOCKET n 의 Option 을 설정한다.

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

Bit	Symbol	Description
[7:2]	-	Reserved
1	DHAM	<b>Destination Hardware address Mode</b> Sn_MR 이 MACRAW 가 아닌 경우 Ethernet Frame 의 Destination Hardware Address 를 설정한다.

		<p>0 : 자동으로 수행되는 ARP/ND-process 에서 획득한 Address 을 사용</p> <p>1 : HOST 가 설정한 Sn_DHAR 를 사용</p>
0	FARP	<p><b>Force ARP</b></p> <p>Sn_MR 이 UDP4, UDP6, UDPD, IPRAW4, IPRAW6 이고, Sn_CR [SEND]나 Sn_CR[SEND6]을 수행 시 마다 ARP/ND-process 를 수행하여 획득한 Destination Hardware Address 를 사용한다.</p> <p>Sn_MR 이 TCP4, TCP6, TCPD 이고, “TCP SERVER”인 경우, “TCP CLIENT”에게 SYN/ACK packet 을 전송하기 이전에 ARP/ND-process 를 수행하여 획득한 Destination Hardware Address 를 사용한다.</p> <p>0 : Disable 1 : Enable</p> <p><b>* CAUTION</b> DHAM = ‘1’ 인 경우, ARP/ND-process 를 수행하지만 Sn_DHAR 을 Destination Hardware Address 로 사용한다.</p>

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

**[R=W] [0x0180-0x0181] [0x0000]**

Sn\_RTR 은 100us 단위의 SOCKET n Retransmission Time 을 설정한다.

Sn\_RTR = ‘0’인 경우, Sn\_CR[OPEN] = ‘1’에 의해 RTR 값으로 초기화된다.

**참조) 6.7 Retransmission**

Ex) S0\_RTR = 5000 (0x1388),  
5000 \* 100us = 0.5s

S0_RTR0(x0180)	S0_RTR1(0x0181)
0x013	0x88

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

**[R=W] [0x0184] [0x00]**

Sn\_RCR 은 SOCKET n Retransmission Counter 를 설정한다. Sn\_RCR = ‘0’인 경우, Sn\_CR[OPEN] = ‘1’에 의해 RCR 값으로 초기화된다. **참조) 6.7 Retransmission**

#### 4.2.22 Sn\_KPALVTR (SOCKET n Keep Alive Time Register)

**[R=W] [0x0188] [0x00]**

Sn\_KPALVTR 는 5s 단위로 SOCKET n TCP Keep Alive(KA) Time 을 설정한다. KA Packet 은 Sn\_SR 이 SOCK\_ESTABLISHED 이고 1 Byte 이상의 DATA 를 전송한 이후부터 전송 가능하며, Sn\_KPALVTR 시간 이후 전송된다.

단, Sn\_KPALVRT = 0x00 인 경우, Sn\_CR [SENDKEEP]에 의해 수동으로 전송된다.

Ex) SO\_KPALVTR = 10 (0x0A),  
10 \* 5s = 50s

SO_KPALVTR(0x0188)
0x0A

#### 4.2.23 Sn\_TX\_BSR (SOCKET n TX Buffer Size Register)

[R=W] [0x0200] [0x02]

Sn\_TX\_BSR 는 SOCKET n TX Buffer Size 를 0, 1, 2, 4, 8, 16 Kbytes 단위로 설정한다.

TX Memory 는 SOCKET 0 부터 SOCKET 7 까지 Sn\_TX\_BSR 값으로 순차적으로 할당된다.

만약 그 외의 값으로 설정되거나 Sn\_TX\_BSR 의 총합이 16 KB 를 초과하는 경우, SOCKET TX Buffer 의 Read/Write Access 과정에서 오작동을 초래할 수 있다.

Value (Dec)	0	1	2	4	8	16
Buffer size	0KB	1KB	2KB	4KB	8KB	16KB

Ex) SO\_TX\_BSR= 4 Kbytes

SO_TX_BSR(0x0200)
0x04

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

[RO] [0x0204-0x205] [0x0800]

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

**In UDP, IPRAW and MACRAW mode**

$Sn\_TX\_FSR = Sn\_TX\_BSR - | Sn\_TX\_WR^{(1)} - Sn\_TX\_RD^{(2)} |$

**In TCP mode,**

$Sn\_TX\_FSR = Sn\_TX\_BSR - | Sn\_TX\_WR - Internal\ Pointer^{(3)} |$

(1) SOCKET n TX Write Pointer Register

(2) SOCKET n TX Read Pointer Register

(3) TCP ACK Pointer managed by W6100

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

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

SO_TX_FSR0(0x0204)	SO_TX_FSR1(0x0205)
0x04	0x00

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

[RO] [0x0208-0x0209] [0x0000]

Sn\_TX\_RD 는 Sn\_CR[OPEN]에 의해 초기화된다. 단, Sn\_MR[3:0]이 TCP4, TCP6, TCP Dual 인 경우, TCP 연결 단계에서 초기값이 재설정된다.

Sn\_CR[SEND]는 SOCKET n TX Buffer 에서 Sn\_TX\_RD 부터 Sn\_TX\_WR 까지 저장된 DATA 를 전송하고 Sn\_TX\_RD 를 Sn\_TX\_WR 과 같은 값으로 자동으로 증가시켜 준다. 만약, 자동 증가한 값이 16bit Offset Address 의 최대 값 0xFFFF 를 초과하여 Carry Bit(17<sup>th</sup> bit)가 발생한 경우, 그 Carry bit 는 무시되고, 하위 16bits 값으로 자동 설정된다.

Ex) S0\_TX\_RD = 0xd4b3

S0_TX_RD0(0x0208)	S0_TX_RD1(0x0209)
0xd4	0xb3

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

**[RW] [0x020C-0x20D] [0x0000]**

Sn\_TX\_WR은 Sn\_CR[OPEN]에 의해 초기화된다. 단, Sn\_MR[3:0]이 TCP4, TCP6, TCPD 인 경우, TCP 연결 단계에서 초기값이 재설정된다.

Sn\_TX\_WR 은 또한 DATA 를 전송하기 위해 다음 절차에 따라 읽거나 갱신한다.

1. HOST 는 전송할 DATA 를 저장할 시작 Address 인 Sn\_TX\_WR 을 읽는다.
2. HOST 는 읽은 Sn\_TX\_WR 을 시작 Address 로 하여 전송할 DATA 를 SOCKET n TX Buffer 에 저장한다.
3. HOST 는 저장한 DATA Size 만큼 Sn\_TX\_WR 를 증가시킨다. 만약, Sn\_TX\_WR 값이 16bits Offset Address 의 최대 값 0xFFFF 를 초과할 경우, 그 Carry bit 는 무시하고 하위 16bits 의 값으로 갱신해야 한다.
4. HOST 는 Sn\_CR[SEND]를 수행하여 SOCKET n TX Buffer 에 저장된 DATA 를 전송한다.

Ex) S0\_TX\_WR = 0x0800

S0_TX_WR0(0x020C)	S0_TX_WR1(0x020D)
0x08	0x00

#### 4.2.27 Sn\_RX\_BSR (SOCKET n RX Buffer Size Register)

**[R=W] [0x0220] [0x02]**

Sn\_RX\_BSR 은 SOCKET n RX Buffer Size 를 0, 1, 2, 4, 8, 16 Kbytes 단위로 설정한다.

SOCKET n RX Buffer 는 RX Memory 에 S0\_RX\_BSR 에서 S7\_RX\_BSR 까지 순차적으로 할당된다. 만약 그 외의 값으로 설정되거나 Sn\_RX\_BSR 의 총합이 16 Kbytes 를 초과하는 경우, SOCKET n RX Buffer 의 Read/Write Access 과정에서 오작동을 초래할 수 있다.

Value (Dec)	0	1	2	4	8	16
Buffer size	0KB	1KB	2KB	4KB	8KB	16KB

Ex) S0\_RX\_BSR = 8 Kbytes

SO\_RX\_BSR(0x0220)

0x08

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

[RO] [0x0224-0x0225] [0x0000]

Sn\_RX\_RSR 은 SOCKET n RX Buffer 에 수신된 DATA Size 를 확인한다.

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) SO\_RX\_RSR = 2048 (0x0800)

SO_RX_RSR0(0x0224)	SO_RX_RSR1(0x0225)
0x08	0x00

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

[RW] [0x0228-0x0229] [0x0000]

Sn\_RX\_RD 는 Sn\_CR[OPEN]에 의해서 초기화된다. SOCKET n Buffer 에 수신된 DATA 는 다음 절차에 따라 읽거나 갱신한다.

1. HOST 는 SOCKET n Buffer 에 수신된 DATA 의 시작 Address 인 Sn\_RX\_RD 을 읽는다.
2. HOST 는 시작 Address 부터 수신된 DATA 를 읽는다.
3. HOST 는 Sn\_RX\_RD 를 읽은 DATA Size 만큼 증가시킨다. 증가된 Sn\_RX\_RD 값이 16bits Offset Address 의 최대 값 0xFFFF 를 초과하여 Carry bit(17<sup>th</sup> bit)가 발생한 경우, 그 Carry bit 는 무시되고 하위 16bits 값으로 갱신한다.
4. HOST 는 Sn\_CR[RECV]를 수행하여 SOCKET n Buffer 를 비운다.

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

SO_RX_RD0(0x0228)	SO_RX_RD1(0x0229)
0x06	0x00

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

[RO] [0x022C-0x022D] [0x0000]

Sn\_RX\_WR 은 SOCKET n RX Buffer Block 에 마지막으로 수신된 DATA 의 Address 이다.

Sn\_RX\_WR 은 Sn\_CR[OPEN]에 의해 초기화 되며 DATA 수신에 의해 자동 증가된다. 만약, 증가된 값이 16bit Offset Address 의 최대 값인 0xFFFF 를 초과하여 Carry bit(17<sup>th</sup> bit)가 발생한 경우, 그 Carry bit 는 무시되고 하위 16bits 값으로 자동 설정된다.

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

SO_RX_WR0(0x022C)	SO_RX_WR1(0x022D)
0x06	0x00

## 5 HOST Interface Mode

W6100 은 HOST 와의 통신을 위한 interface Mode 로써 SPI(Serial Peripheral Interface)와 Parallel BUS 를 제공하며 이는 PIN MOD[3:0]를 통해 설정할 수 있다.

SPI 는 CSn, SCLK, MOSI, MISO 로 구성되어 있고 Parallel BUS 의 경우 4 개의 Control Signal(CSn, WRn, RDn, INTn)과 Address Bus(2bits), Data Bus(8bits) 로 구성되어 있다.

### 5.1 SPI Mode

W6100 은 MOD[3:0] = '000X' 일 경우, SPI Mode 로 동작하며, SPI Slave 만으로 사용 가능하다. HOST 와 W6100 은 '[5.1.2 Variable Length Data Mode \(VDM\)](#)' & '[5.1.3 Fixed Length Data Mode \(FDM\)](#)'에서 설명될 SPI Operation Mode 에 따라서 아래의 [Figure 5 오류! 참조 원본을 찾을 수 없습니다.](#) 와 [Figure 6](#) 의 두 가지 방식으로 연결될 수 있다. [Figure 5 오류! 참조 원본을 찾을 수 없습니다.](#) 방식은 SPI Master 의 선택에 따라 다른 SPI Slave 들과 공유할 수 있는 반면, [Figure 6](#) 은 W6100 전용 SPI 로 사용되어 다른 SPI Slave 들과 공유할 수 없다.

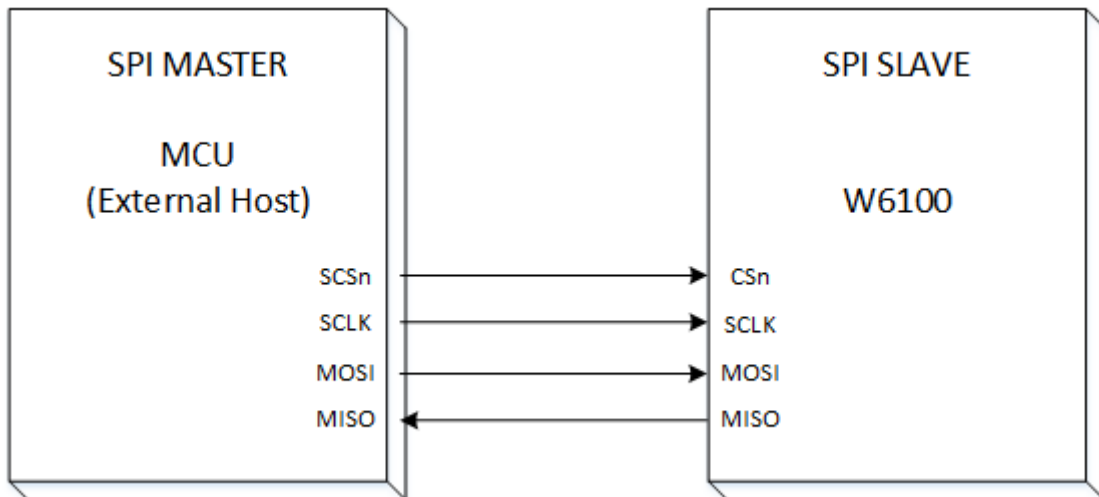


Figure 5 Variable Length Data Mode (CSn controlled by Host)

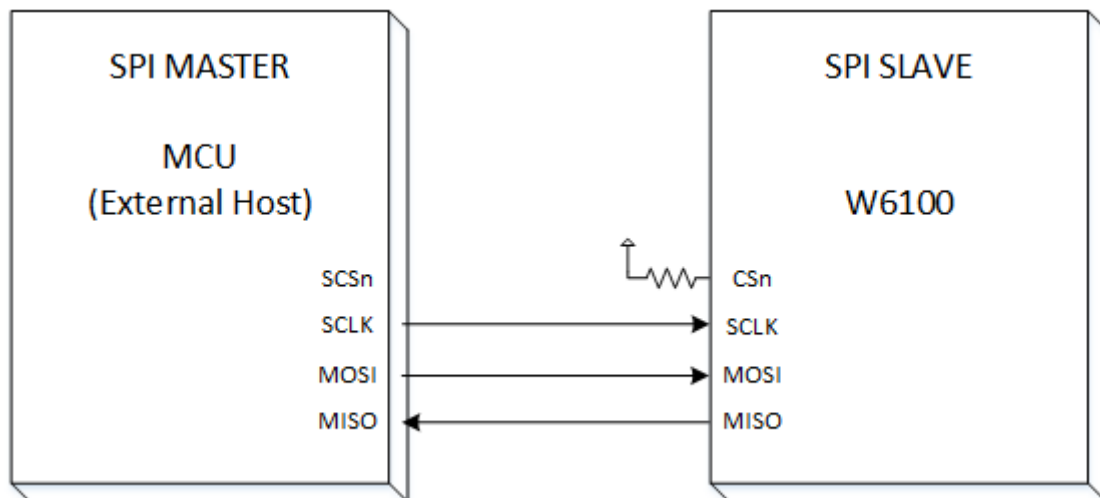


Figure 6 Fixed Length Data Mode (CSn is always connected by Ground)

또한 W6100 은 Figure 7 과 같이 SPI Mode 0 과 Mode 3 만을 지원하며, MOSI 는 SCLK 의 Rising edge 인 경우, 수신(Sampling)되고, MISO 는 Falling edge 인 경우, 송신(Toggling)된다. 그리고 MOSI 와 MISO 는 SCLK 마다 MSB 에서 LSB 까지 순차적으로 송수신된다.

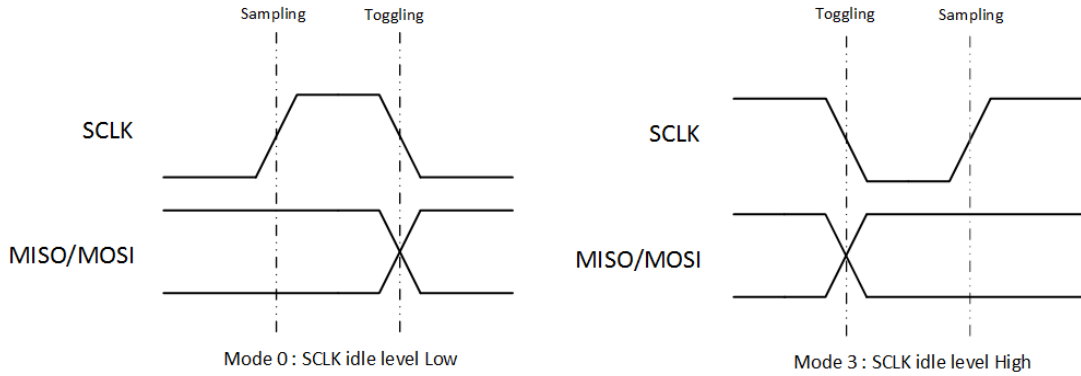


Figure 7 SPI Mode 0 & Mode 3

### 5.1.1 SPI Frame

W6100 은 HOST 로부터 송수신되는 SPI Frame 에 의해 제어되며 해당 SPI Frame 은 Figure 8 과 같이 Address Phase, Control Phase, Data Phase 로 구성되어 있다.

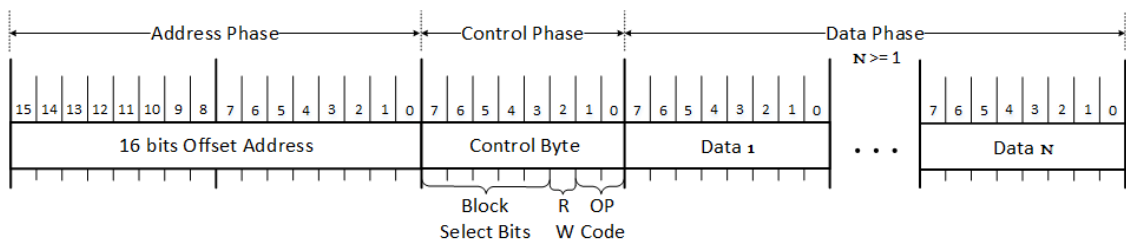


Figure 8 SPI Frame Format

Address Phase 는 W6100 의 Register 나 TX/RX Memory 에 대한 16bits Offset Address 를 지정한다. Control Phase 는 Address Phase 에서 지정된 Offset Address 가 속한 Block 을 선택하고, Read/Write Access Mode 및 SPI Operation Mode(Variable Length Data / Fixed Length Data Mode)를 지정한다. 마지막으로 Data Phase 에서는 SPI Operation Mode 에 따른 임의 길이(N byte) Data 또는 1,2,4 byte Data 를 지정한다.

SPI Operation Mode 가 Variable Length Data Mode(VDM)일 경우 CSn 은 반드시 HOST 에 의해 SPI Frame 단위로 제어 되어야 한다. VDM 인 경우, HOST 는 W6100 에게 CSn 를 Assert(High-to-Low)함으로써 SPI Frame 의 시작을 알리고, De-assert(Low-to-High)를 통해 종료(N byte Data Phase 전송종료)를 알린다.

VDM 과 FDM 을 간단히 설명하자면, VDM 은 SCS 의 컨트롤로 Data 를 전송하기 때문에 length 제한이 없고 FDM 은 CSn 이 low 로 고정 되어있고, SPI Operation Mode 에 따라 1,2,4 bytes 를 전송할 수 있다.

### 5.1.1.1 Address Phase

Address Phase 는 W6100 의 Common 및 Socket Register, Socket n TX/RX Buffer Block 에 대한 16bits Offset Address 를 지정한다. 이때 16bits Offset Address 값은 MSB 부터 LSB 순으로 순차적으로 송신된다.

W6100 SPI Interface 는 SPI Frame 의 Offset Address 를 시작(Base)으로 1 byte Data 전송 마다 자동으로 Offset Address 가 1 씩 증가하는 Sequential Data Read/Write 를 지원한다.

### 5.1.1.2 Control Phase

Control Phase 는 Block Select Bits(BSB[4:0])를 통해 Address Phase 에서 지정된 Offset Address 가 속한 Block 을 지정하고, Read/Write Access Mode, SPI Operation Mode 를 각각 Read Write Access Mode Bit(RWB)와 SPI Operation Mode Bits(OM[1:0])를 통해 지정한다.

7	6	5	4	3	2	1	0
BSB4	BSB3	BSB2	BSB1	BSB0	RWB	OM1	OM0

Bit	Symbol	Description																														
7-3	BSB	<b>Block Selection Bits</b> W6100 에는 1 개의 Common, 8 개의 SOCKET n Register(n=0-7) 그리고 각각의 SOCKET n 에 할당되는 TX/RX Buffer 와 같은 Block 들이 존재한다. 아래 표는 BSB[4:0]를 통해 선택되는 Block 을 도식화한다.																														
		<table border="1"> <thead> <tr> <th>BSB[4:2]</th> <th>BSB[1:0]</th> <th>Block</th> </tr> </thead> <tbody> <tr> <td rowspan="4">000</td> <td>00</td> <td>Common Register</td> </tr> <tr> <td>01</td> <td>Socket 0 Register</td> </tr> <tr> <td>10</td> <td>Socket 0 TX Buffer</td> </tr> <tr> <td>11</td> <td>Socket 0 RX Buffer</td> </tr> <tr> <td rowspan="4">001</td> <td>00</td> <td>Reserved</td> </tr> <tr> <td>01</td> <td>Socket 1 Register</td> </tr> <tr> <td>10</td> <td>Socket 1 TX Buffer</td> </tr> <tr> <td>11</td> <td>Socket 1 RX Buffer</td> </tr> <tr> <td rowspan="4">010</td> <td>00</td> <td>Reserved</td> </tr> <tr> <td>01</td> <td>Socket 2 Register</td> </tr> <tr> <td>10</td> <td>Socket 2 TX Buffer</td> </tr> <tr> <td>11</td> <td>Socket 2 RX Buffer</td> </tr> </tbody> </table>	BSB[4:2]	BSB[1:0]	Block	000	00	Common Register	01	Socket 0 Register	10	Socket 0 TX Buffer	11	Socket 0 RX Buffer	001	00	Reserved	01	Socket 1 Register	10	Socket 1 TX Buffer	11	Socket 1 RX Buffer	010	00	Reserved	01	Socket 2 Register	10	Socket 2 TX Buffer	11	Socket 2 RX Buffer
		BSB[4:2]	BSB[1:0]	Block																												
		000	00	Common Register																												
			01	Socket 0 Register																												
			10	Socket 0 TX Buffer																												
			11	Socket 0 RX Buffer																												
		001	00	Reserved																												
			01	Socket 1 Register																												
			10	Socket 1 TX Buffer																												
			11	Socket 1 RX Buffer																												
		010	00	Reserved																												
01	Socket 2 Register																															
10	Socket 2 TX Buffer																															
11	Socket 2 RX Buffer																															



		<p style="text-align: center;">.</p> <p style="text-align: center;">.</p> <p style="text-align: center;">.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 50px;"></td> <td style="width: 50px; text-align: center;">00</td> <td style="background-color: #cccccc;">Reserved</td> </tr> <tr> <td style="text-align: center;">111</td> <td style="text-align: center;">01</td> <td>Socket 7 Register</td> </tr> <tr> <td></td> <td style="text-align: center;">10</td> <td>Socket 7 TX Buffer</td> </tr> <tr> <td></td> <td style="text-align: center;">11</td> <td>Socket 7 RX Buffer</td> </tr> </table> <p>Ex) SOCKET 2 Register 인 경우, BSB[4:2] = '010', BSB[1:0] = '01'.</p>		00	Reserved	111	01	Socket 7 Register		10	Socket 7 TX Buffer		11	Socket 7 RX Buffer
	00	Reserved												
111	01	Socket 7 Register												
	10	Socket 7 TX Buffer												
	11	Socket 7 RX Buffer												
2	RWB	<p><b>Read/Write Access Mode Bit</b> SPI Read/Write Access Mode 를 설정한다.</p> <p>0 : Read 1 : Write</p>												
1~0	OM[1:0]	<p><b>SPI Operation Mode Bits</b> SPI Operation Mode 를 설정하며 Operation Mode 는 Variable Length Data Mode(VDM)와 Fixed Length Data Mode(FDM)가 있다.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th style="background-color: #add8e6;">OM[1:0]</th> <th style="background-color: #add8e6;">Mode</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">00</td> <td>VDM, N bytes Data Phase (1 ≤ N)</td> </tr> <tr> <td style="text-align: center;">01</td> <td>FDM, 1 byte Data Phase</td> </tr> <tr> <td style="text-align: center;">10</td> <td>FDM, 2 bytes Data Phase</td> </tr> <tr> <td style="text-align: center;">11</td> <td>FDM, 4 bytes Data Phase</td> </tr> </tbody> </table> <p><b>Variable Length Data Mode (VDM)</b> SPI Frame 의 Data Phase 에서 N-bytes Data 를 송수신하는 Mode 로, CSn 제어를 통해 Data Length 가 결정된다. HOST 는 CSn 을 Assert(High-to-Low)함으로써 W6100 에게 SPI Frame 의 Address Phase 송신 시작을 알리고, OM[1:0] = '00'을 포함하는 Control Phase 를 송신하고, N-bytes Data Phase 송수신을 완료 후, CSn 을 De-assert(Low-to-High)시켜 W6100 에 SPI Frame 송수신 이 완료 되었음을 알린다. VDM 에서 CSn 은 반드시 HOST 에 의해 SPI Frame 단위로 제어 되어야 한다.</p> <p><b>Fixed Length Data Mode (FDM)</b> VDM 에서는 HOST 의 CSn Control 에 의해 Data Length 가 결정되는 반면, Fixed Length Data Mode 에서는 OM[1:0]값에 의해 Data Length 가 결정된다. 따라서, CSn 은 항상 Low 상태를 유지해야 하며, Data</p>	OM[1:0]	Mode	00	VDM, N bytes Data Phase (1 ≤ N)	01	FDM, 1 byte Data Phase	10	FDM, 2 bytes Data Phase	11	FDM, 4 bytes Data Phase		
OM[1:0]	Mode													
00	VDM, N bytes Data Phase (1 ≤ N)													
01	FDM, 1 byte Data Phase													
10	FDM, 2 bytes Data Phase													
11	FDM, 4 bytes Data Phase													

Length 는 OM[1:0] 값 설정에 따라 반드시 1, 2, 4 bytes 중 하나의 값을 갖는다.

### 5.1.1.3 Data Phase

Data Phase 의 크기는 Control Phase 의 SPI Operation Mode Bits(OM[1:0])의 설정에 따라 N bytes(VDM) 또는 1, 2, 4 bytes(FDM)로 결정된다. Data 는 byte 단위로 송수신되며 반드시 MSB 부터 LSB 순으로 MOSI 또는 MISO 를 통해 1 bits 씩 순차적으로 송수신된다.

## 5.1.2 Variable Length Data Mode (VDM)

VDM 은 HOST 의 CSn Control 에 의해 SPI Frame 의 Data Phase Length 가 결정된다. 즉, Data Phase 의 Length 는 CSn Control 에 따라 1 byte 부터 N byte 까지 임의의 길이를 가질 수 있다. 또한 VDM 에서 Control Phase 의 OM[1:0]는 반드시 '00'값으로 설정되어야 한다.

### 5.1.2.1 Write Access in VDM

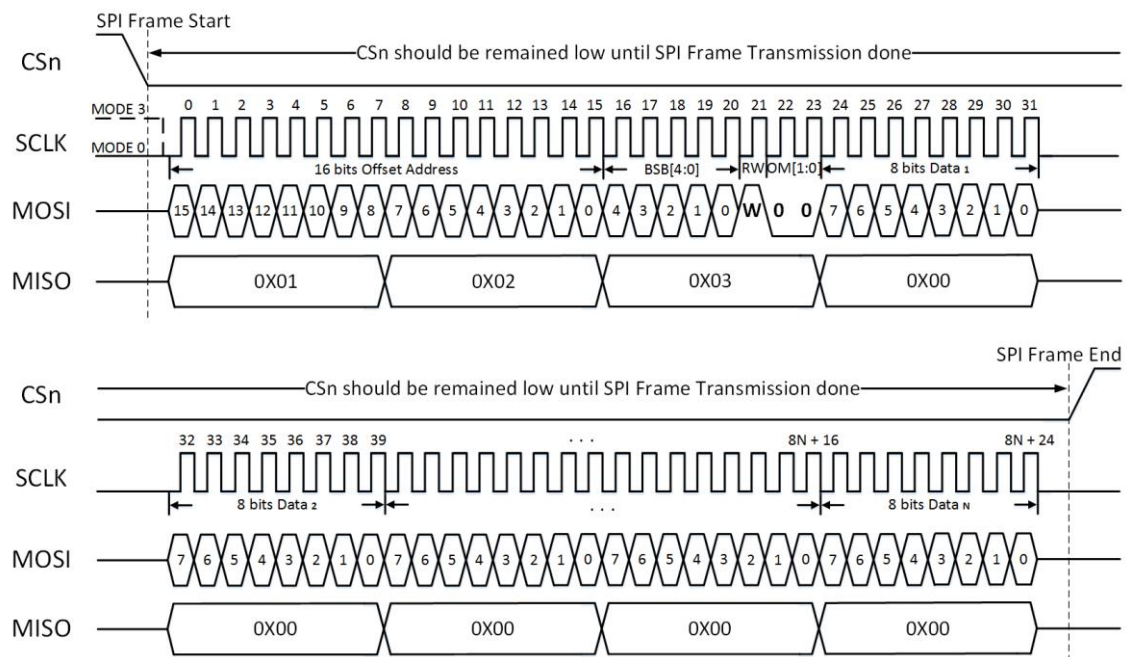


Figure 9 Write SPI Frame in VDM

Figure 9 는 VDM 으로 Write Access 하는 SPI Frame 을 보여준다. HOST 는 CSn 을 Assert(High-to-Low)함으로써 W6100 에게 Address Phase 전송을 알린다. Control Phase 전송시 RWB 를 '1', OM[1:0]을 '00'으로 설정해 Write Access 및 VDM 임을 알린다. MOSI 를 통해 전송되는 Data Bits 는 Toggling SCLK(Falling-Edge)에 동기화되어 W6100 으로 전송된다. VDM 은 CSn 이 Low 이고 Data Phase 가 계속 전송될 경우 Sequential Data Write 를 지원한다. 모든 Data Phase 의 전송이 완료되면 CSn 을 De-assert(Low-to-High)한다.

### 5.1.2.2 Read Access in VDM

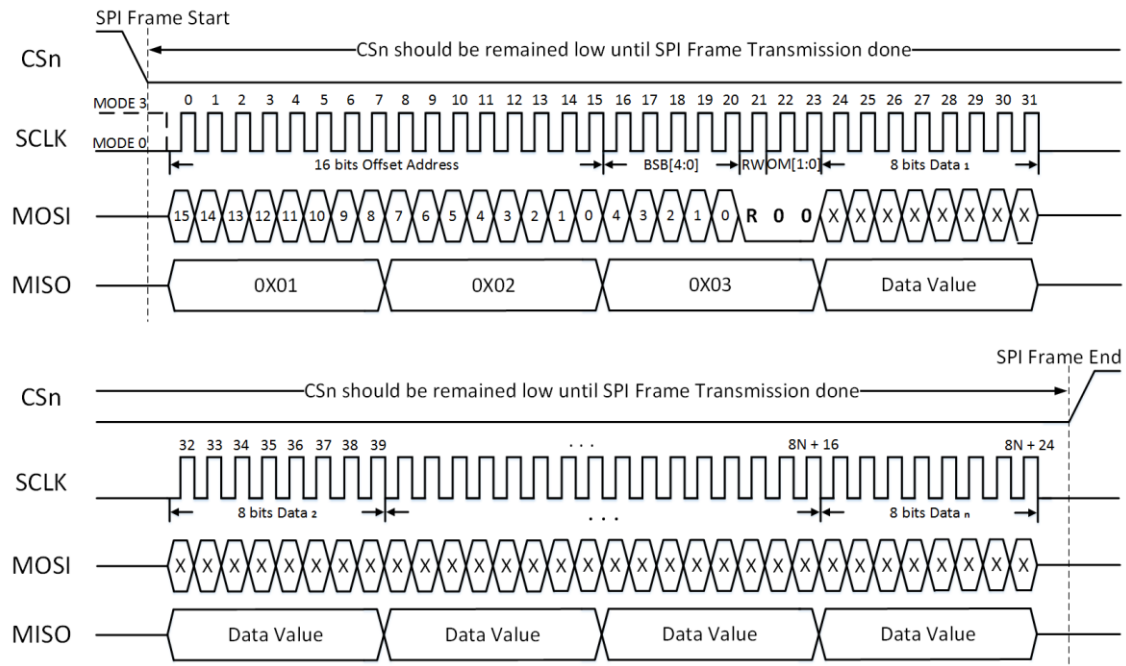


Figure 10 Read SPI Frame in VDM

Figure 10 은 VDM 으로 Read Access 하는 SPI Frame 을 보여준다. HOST 는 CSn 을 Assert(High-to-Low)함으로써 W6100 에게 Address Phase 전송을 알린다. Control Phase 전송시 RWB 를 '0'으로, OM[1:0]을 '00'으로 설정해 Read Access 및 VDM 임을 알린다. Address & Control Phase 를 Toggling SCLK(Falling-Edge)에 동기화하여 MOSI 을 통해 1 Bit 씩 전송하고 Sampling SCLK(Rising-Edge)에 동기화하여 MISO 를 통해 들어오는 Data 를 수신한다. CSn 이 Low 이고 Data Phase 를 계속 수신할 경우 Sequential Data Read 를 지원한다. Data Phase 의 수신이 완료되면 CSn 을 De-assert(Low-to-High)한다.

### 5.1.3 Fixed Length Data Mode (FDM)

FDM 은 HOST 에서 CSn 을 Control 할 수 없고 W6100 의 CSn 이 Always Low (Connected GND)인 경우에 사용된다.

VDM 에서 CSn 을 Assert 및 De-assert 함으로써 Data Phase 의 Length 를 결정하는 것에 반해 FDM 에서는 Control Phase 의 SPI Operation Mode Bits(OM[1:0])를 통해 Data Phase Length 를 결정한다. OM[1:0]의 값이 '01' / '10' / '11'에 따라서 Data Phase Length 는 각기 1 / 2 / 4 bytes 로 결정되며 OM[1:0]의 설정과 다른 Data Phase Length 를 갖는 SPI Frame 의 전송은 W6100 의 비정상 동작을 야기한다.

SPI Frame 은 VDM 과 동일하므로 생략한다. 참조) [5.1.2 Variable Length Data Mode \(VDM\)](#)

### 5.1.3.1 Write Access in FDM

#### 1 byte Write Access

Address Phase														Control Phase			Data Phase														
														BSB			RWB	OM	1 <sup>st</sup> Data												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	4	3	2	1	0	R/W	1	0	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	1	0	1	x	x	x	x	x	x	x	x

Figure 11 1 byte Data Write Access SPI Frame in FDM

#### 2 bytes Write Access

Address Phase														Control Phase			Data Phase														
														BSB			RWB	OM	1 <sup>st</sup> Data												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	4	3	2	1	0	R/W	1	0	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	1	1	0	x	x	x	x	x	x	x	x

Data Phase							
2 <sup>nd</sup> Data							
7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x

Figure 12 2 bytes Data Write Access SPI Frame in FDM

#### 4 bytes Write Access

Address Phase														Control Phase			Data Phase														
														BSB			RWB	OM	1 <sup>st</sup> Data												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	4	3	2	1	0	R/W	1	0	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	1	1	1	x	x	x	x	x	x	x	x

Data Phase								Data Phase								Data Phase							
2 <sup>nd</sup> Data								3 <sup>rd</sup> Data								4 <sup>th</sup> Data							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Figure 13 4 bytes Data Write Access SPI Frame in FDM

### 5.1.3.2 Read Access in FDM

#### 1 byte Read Access

Address Phase														Control Phase			Data Phase														
														BSB			RWB	OM	1 <sup>st</sup> Data												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	4	3	2	1	0	R/W	1	0	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0	1	x	x	x	x	x	x	x	x

Figure 14 1 byte Data Read Access SPI Frame in FDM

#### 2 bytes Read Access

Address Phase														Control Phase			Data Phase														
														BSB			RWB	OM	1 <sup>st</sup> Data												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	4	3	2	1	0	R/W	1	0	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	1	0	x	x	x	x	x	x	x	x

Data Phase							
2 <sup>nd</sup> Data							
7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x

Figure 15 2 bytes Data Read Access SPI Frame in FDM

#### 4 bytes Read Access

Address Phase														Control Phase			Data Phase														
														BSB			RWB	OM	1 <sup>st</sup> Data												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	4	3	2	1	0	R/W	1	0	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	1	1	x	x	x	x	x	x	x	x

Data Phase								Data Phase								Data Phase							
2 <sup>nd</sup> Data								3 <sup>rd</sup> Data								4 <sup>th</sup> Data							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Figure 16 4 bytes Data Read Access SPI Frame in FDM

## 5.2 Parallel BUS Mode

PIN MOD[3:0] = '010X' 일 경우 Parallel BUS Mode 를 지원하며, HOST 와 W6100 은 Figure 17 과 같이 연결된다.

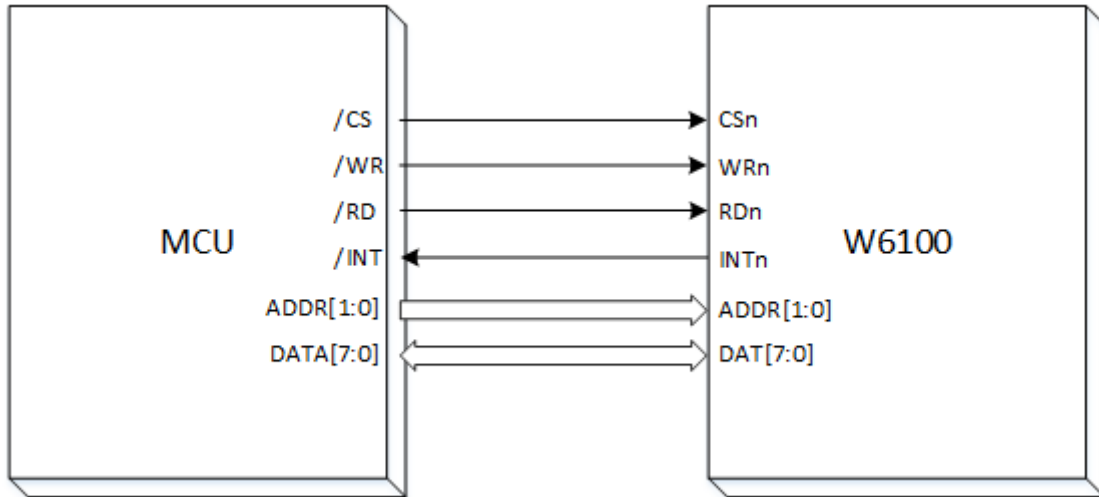


Figure 17 Host Interface in Parallel BUS Mode

Parallel BUS Mode 에서 W6100 은 HOST 에서 전송되는 ADDR[1:0]을 통해 DAT [7:0]는 Read/Write Access 할 16 bits Offset Address Register 인지, 지정된 Offset Address 가 속한 Block 을 선택하는 Block Select Bits 인지, 또는 지정된 Offset Address 에 저장되는 Data 인지를 구분된다.

ADDR[1:0]은 Table 3 에서 보여주는 것처럼 값에 따라서 16 bits Offset Address Register 의 상위 8 bits 를 지정하는 IDM\_ARH, 하위 8 bits 를 지정하는 IDM\_ARL, 지정된 Offset Address 가 속한 Block 을 선택하는 IDM\_BSB 그리고 Offset Address 에 저장되는 Data 를 나타내는 IDM\_DR 로 구분된다.

Table 3 Registers in Parallel BUS Mode

ADDR[1:0]	Symbol	Description
00	IDM_ARH	<b>Indirect Mode High Address Register</b> 상위 8 bits Offset Address Register
01	IDM_ARL	<b>Indirect Mode Low Address Register</b> 하위 8 bits Offset Address Register

10	IDM_BSR	<b>Block Select Register</b>					
		<p>W6100 에는 Common, 8 개의 Socket n Register(n=0-7) 그리고 각각의 Socket n 에 할당되는 TX/RX Buffer 와 같은 Block 들이 존재한다.</p> <p>아래 표는 IDM_BSR[7:3]을 통해 선택되는 Block 을 도식화한다.</p>					
				[7:5]	[4:3]	[2:0]	Description
			000		00	Reserved	Common Register
				01	Socket 0 Register		
				10	Socket 0 TX Buffer		
				11	Socket 0 RX Buffer		
			001		00		Reserved
				01	Socket 1 Register		
				10	Socket 1 TX Buffer		
				11	Socket 1 RX Buffer		
			010		00		Reserved
				01	Socket 2 Register		
				10	Socket 2 TX Buffer		
				11	Socket 2 RX Buffer		
						⋮	
						⋮	
						⋮	
			111		00	Reserved	
				01	Socket 7 Register		
	10	Socket 7 TX Buffer					
	11	Socket 7 RX Buffer					
11	IDM_DR	8 Bits Data Register					

### 5.2.1 Parallel BUS Mode Data Write

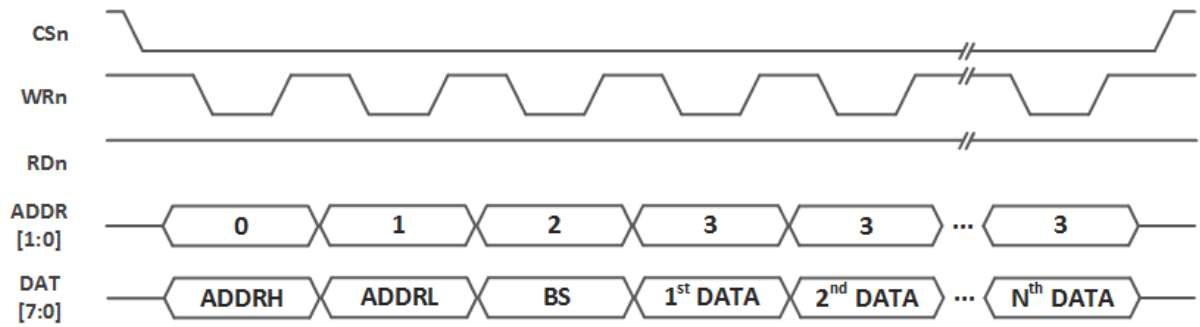


Figure 18 Parallel Bus N-Bytes Data Write Access

Figure 18 은 N-Bytes Data Write 를 보여준다. HOST 는 CSn 을 Assert(High-to-Low) 함으로써 W6100 에게 N-Bytes Data 전송을 알린다. Write Access 시 ADDR[1:0]의 값을 순차적으로 '00' / '01' / '10' / '11'으로 설정함으로써 16 bits Offset Address Register 의 상위 8bits, 하위 8bits, Block Select Bits 를 설정하고 Data 를 전송한다.

Parallel Bus Mode 는 N-Bytes Sequential Data Write 를 지원한다.

### 5.2.2 Parallel Bus Mode Data Read

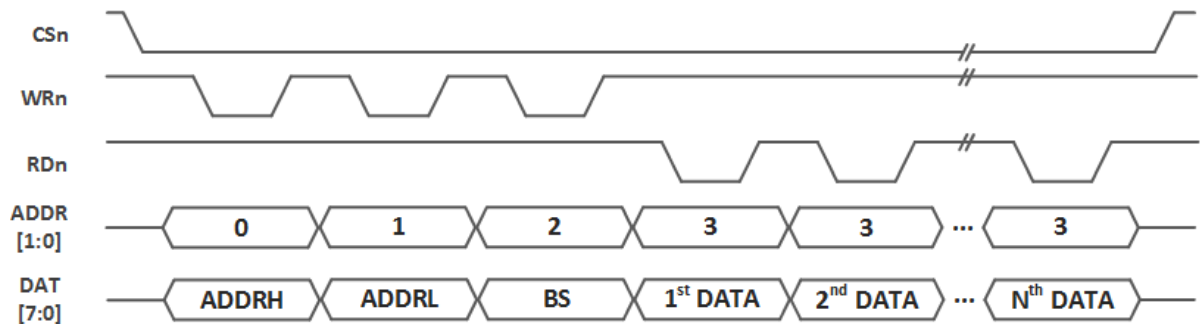


Figure 19 Parallel Bus Mode Continuous Read Access

Figure 19 는 N-Bytes Data Read 를 보여준다. HOST 는 CSn 을 Assert(High-to-Low) 함으로써 W6100 에게 N-Bytes Data 전송을 알린다. Write Access 시 ADDR[1:0]의 값을 순차적으로 '00' / '01' / '10' / '11'으로 설정함으로써 16 bits Offset Address Register 의 상위 8bits, 하위 8bits, Block Select Bits 를 설정하고 W6100 으로부터 Data 를 수신한다.

Parallel Bus Mode 는 N-Bytes Sequential Data Read 를 지원한다.

## 6 Functional Description

W6100 은 간단한 Register 조작만으로 Internet Connectivity 를 제공한다. Functional Description 은 W6100 의 Initialization 및 각 Protocol(TCP, UDP, IPRAW, MACRAW) 과 추가 기능에서 다른 DATA 통신방법에 대하여 단계별 Pseudo Code 를 설명한다.

### 6.1 Initialization

Network 정보와 SOCKET n TX/RX Buffer Block 등을 초기화한다.

#### 6.1.1 Network Information Setting

IPv4 및 IPv6 통신을 위한 기본 Network Information 을 설정한다.

<pre> Network Configuration Unlock: {     /* Network Unlock before set Network Information */     NETLCKR = 0x3A; } </pre>
<pre> Source Hardware Address: {     /* Source Hardware Address, 11:22:33:AA:BB:CC */     SHAR[0:5] = { 0x11, 0x22, 0x33, 0xAA, 0xBB, 0xCC }; } </pre>
<pre> IPv4 Network Information: {     /* Gateway IP Address, 192.168.0.1 */     GAR[0:3] = { 0xC0, 0xA8, 0x00, 0x01 };      /* Subnet MASK Address, 255.255.255.0 */     SUBR[0:3] = { 0xFF, 0xFF, 0xFF, 0x00 };      /* W6100 IP Address, 192.168.0.100 */     SIPR[0:3] = { 0xC0, 0xA8, 0x00, 0x64 }; } </pre>
<pre> IPv6 Network Information: {     /* Link Local Address, FE80::1322:33FF:FEAA:BBCC */     LLAR[0:15] = { 0xFE, 0x80, 0x00, 0x01, 0x00, 0x00, 0x00, 0x00,                   0x13, 0x22, 0x33, 0xFF, 0xFE, 0xAA, 0xBB, 0xCC };      /* Global Unicast Address, 2001:0DB8:E001::1222:33FF:FEAA:BBCC */     GUAR[0:15] = { 0x20, 0x01, 0x0D, 0xB8, 0xE0, 0x01, 0x00, 0x00,                   0x13, 0x22, 0x33, 0xFF, 0xFE, 0xAA, 0xBB, 0xCC };      /* IPv6 Subnet Mask Address, FFFF:FFFF:: */     SUB6R[0:15] = { 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,                   0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };      /* IPv6 Gateway Address, FE80::1322:33FF:FE44:5566 */     GA6R[0:15] = { 0xFE, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,                   0x13, 0x22, 0x33, 0xFF, 0xFE, 0x44, 0x55, 0x66 }; } </pre>



```

0x13, 0x22, 0x33, 0xFF, 0xFE, 0x44, 0x55, 0x66 };
}
Network Configuration Lock:
{
    /* Network Lock before set Network Information */
    NETLCKR = Any value except 0x3A;
}

```

### 6.1.2 SOCKET TX/RX Buffer Size Setting

SOCKET 통신 이전에 사용자는 Sn\_TX\_BSR/Sn\_RX\_BSR 설정을 통해 SOCKET n TX/RX Buffer Size 를 결정한다.

SOCKET n TX/RX Buffer Size 는 0,1,2,4,8,16KB 로 설정 가능하나 각 총합이 TX/RX Memory 의 크기 16KB 를 초과할 수 없다.

```

In case of, assign 2Kbytes RX/TX Memory per SOCKET
{
    // set Base Address of TX/RX Memory for SOCKET n
    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<7; n++) {
        Sn_TX_BSR = 2; // assign 2 Kbytes TX Memory per SOCKET
        Sn_RX_BSR = 2; // assign 2 Kbytes RX Memory per SOCKET
        TxTotalSize = TxTotalSize + Sn_TX_BSR;
        RxTotalSize = RxTotalSize + Sn_RX_BSR;
        If( TxTotalSize > 16 or RxTotalSize > 16 ) goto ERROR; // invalid Total Size
    } // end for
}

```

## 6.2 TCP

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

TCP 는 1:1 연결 기반이므로 Destination 과의 DATA 송수신을 위해서 Destination 에게 연결요청을 하거나 Destination 으로부터 연결요청을 받는다. 이 과정에서 먼저 연결요청을 하는 쪽을 ‘TCP CLIENT’, 연결요청을 받는 쪽을 ‘TCP SERVER’로 구분한다. TCP 는 신뢰성 있는 전송 프로토콜로써 연결이 완료된 이후의 DATA 송수신 과정에서 확인 응답 메커니즘을 통해 전송 DATA 의 수신여부를 확인하고 손실되거나 훼손된 DATA 는 재전송한다. ‘TCP SERVER’와 ‘TCP CLIENT’는 TCP 연결종료가 이루어지기 전까지 연결을 유지하며 DATA 를 송수신한다.

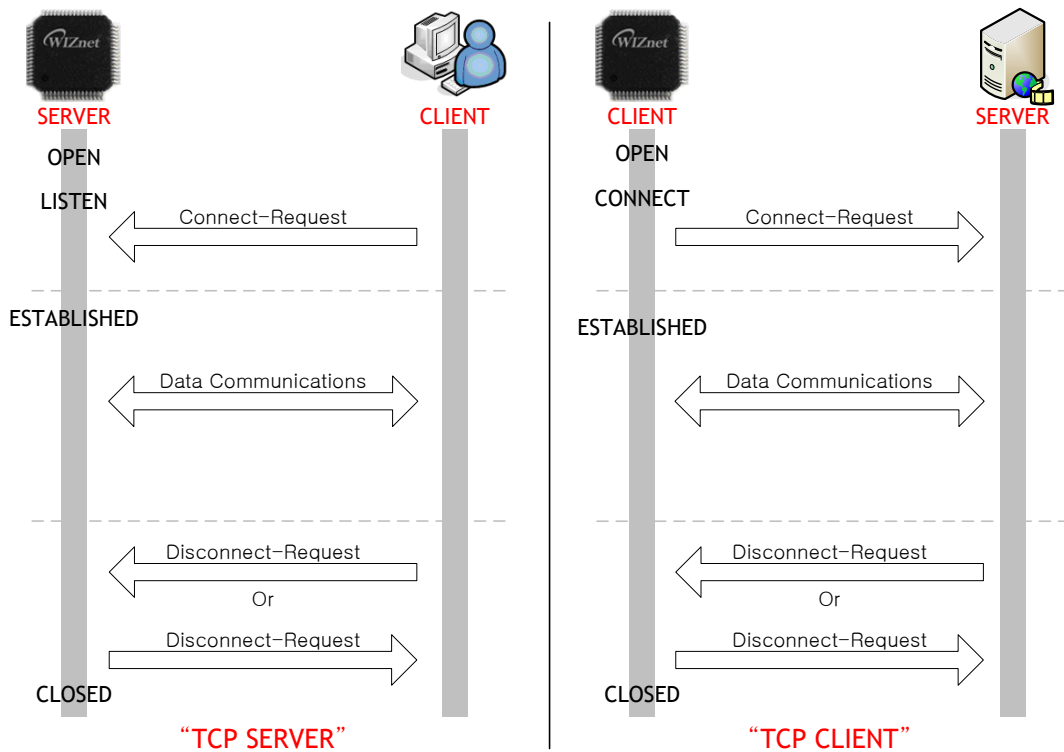


Figure 20 TCP SERVER and TCP CLIENT

6.2.1 TCP SERVER

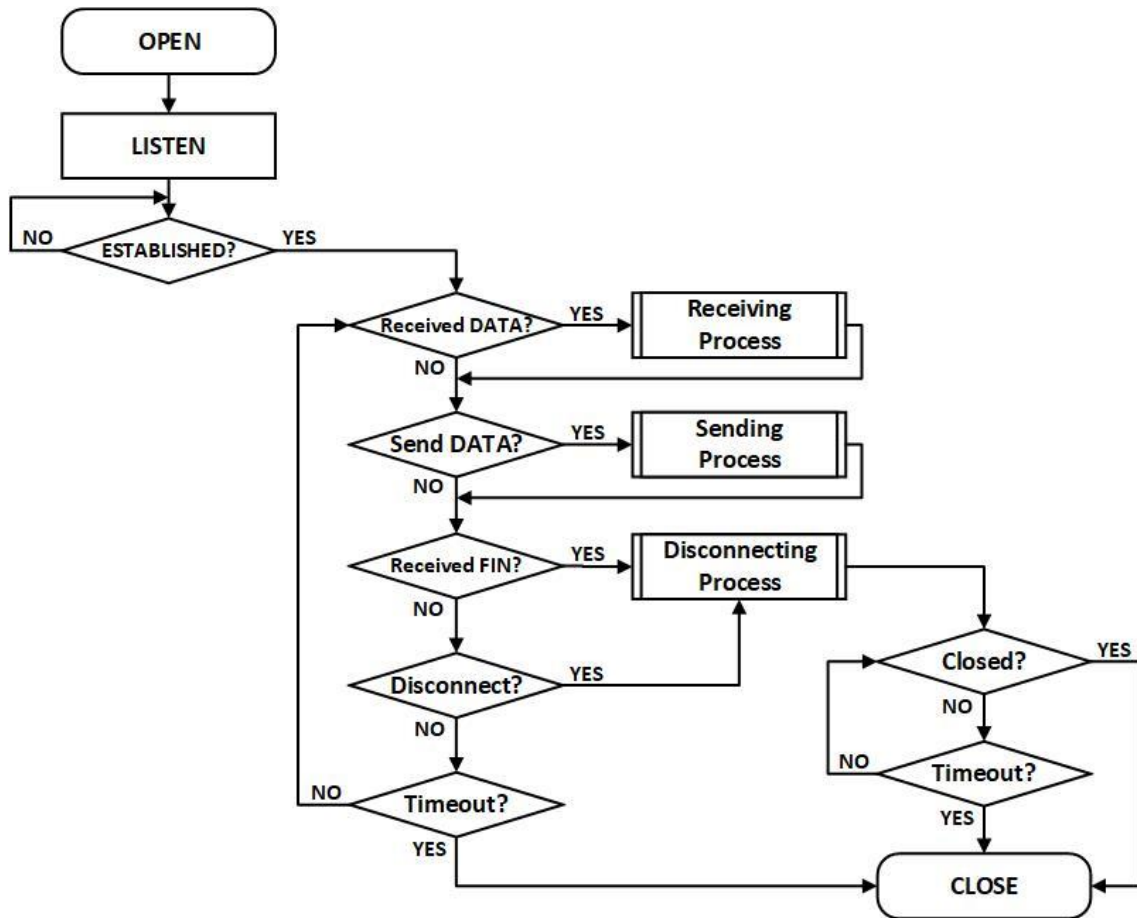


Figure 21 TCP SERVER Operation Flow

• OPEN

SOCKET n 을 TCP4/TCP6 Mode 로 OPEN 한다.

```

TCP Mode : TCP4, TCP6
{
START :
Sn_MR[3:0] = '0001'; /* set TCP4 Mode */
// Sn_MR[3:0] = '1001'; /* set TCP6 Mode */

Sn_PORTR[0:1] = {0x13,0x88}; /* set PORT Number, 5000(0x1388) */
Sn_CR[OPEN] = '1'; /* set OPEN Command */
while(Sn_CR != 0x00); /* wait until OPEN Command is cleared*/

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

• LISTEN

SOCKET n 을 'TCP SERVER'로 동작시킨다. Destination 의 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] 또는 Sn\_SR[SOCK\_ESTABLISHED]로 확인할 수 있다. 접속이 완료되면 TCP4/TCP6 Mode 에 따라 Destination Address 는 Sn\_DIPR 또는 Sn\_DIP6R 을 확인할 수 있다.

```

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

  /* check destination address */
  if(Sn_MR[3:0] == TCP6 Mode)
    destination_addr[0:15] = Sn_DIP6R;
  else if(Sn_MR[3:0] == TCP4 Mode)
    destination_addr[0:3] = Sn_DIPR;
}

```

```

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

  /* check destination address */
  if(Sn_MR[3:0] == TCP6 Mode)
    destination_addr[0:15] = Sn_DIP6R;
  else if(Sn_MR[3:0] == TCP4 Mode)
    destination_addr[0:3] = Sn_DIPR;
}

```

• Receive DATA?

SOCKET n 의 DATA 수신여부는 Sn\_RX\_RSR 또는 Sn\_IR[RCV]을 통해 확인한다.

```

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

```

```

Second method :
{
    /* check SOCKET RECV Interrupt bit */
    if (Sn_IR[RECV] == '1')
    {
        Sn_IRCLR[RECV] = '1'; /* clear SOCKET Interrupt */
        goto Receiving Process;
    } // end if
}

```

#### • Receiving Process

SOCKET RX Buffer 로부터 수신된 DATA 를 Read 한다.

수신 DATA 를 Read 후, 그 Size 만큼 Sn\_RX\_RD 를 증가시키고, Sn\_CR[RECV] 를 수행한다. 만약, Sn\_CR[RECV] 수행 이후 SOCKET n RX Buffer 에 DATA 가 남아있는 경우 Sn\_IR[RECV] 이 다시 발생한다.

```

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

    /* calculate SOCKET n RX Buffer Size */
    gSn_RX_MAX = Sn_RX_BSR * 1024;

    /* calculate Read Offset Address */
    get_start_address = Sn_RX_RD;

    /* 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

SOCKET n TX Buffer 에 Write 된 DATA 를 전송한다.

DATA Write 후, 그 Size 만큼 Sn\_TX\_WD 를 증가시키고, Sn\_CR[SEND] 를 수행한다. 다음 DATA 를 전송할 준비가 완료(Sn\_IR[SENDOK] = '1') 될 때까지 다음 “Sending Process” 를 수행할 수 없으며, 전송 TIMEOUT 이 발생할 수 있다. [참조\) 6.7 Retransmission](#)

Sn\_IR[SENDOK]가 발생할 때까지의 시간은 사용된 SOCKET Count, DATA Size 와 Network Traffic 등에 의존적이다.

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 */
  gSn_TX_MAX = Sn_TX_BSR * 1024;

  /* 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_size > 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 = Sn_TX_WR;

  /* copy get_size of get_start_address to destination_address */
  memcpy(get_start_address, destination_address, send_size);

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

  /* set SEND and SEND6 Command in each TCP and TCP6 Mode */
  Sn_CR = SEND; /* set SEND command in TCP Mode */
  while(Sn_CR != 0x00); /* wait until SEND or SEND6 Command is cleared */

  /* wait until SEND or SEND6 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_IRCLR[SENDOK] = '1';
  else
    goto Timeout?;
}

```

• Received FIN (Passive Close)

Destination 으로부터 연결종료 요청(FIN Packet)을 수신한 경우이며 FIN Packet 의 수신여부는 Sn\_SR 또는 Sn\_IR[DISCON]을 통해 확인한다.

First Method: <pre> {   If(Sn_SR == SOCK_CLOSE_WAIT) goto Disconnecting Process; } </pre>
Second Method: <pre> {   If(Sn_IR[DISCON] == '1') goto Disconnecting Process; } </pre>

- **Disconnected (Active Close)**

Destination 에게 연결종료 요청(FIN Packet)을 전송한다.

```

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

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

```

- **Disconnecting Process**

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

Destination 에게 FIN Packet 을 전송한 경우(Active Close), Destination 의 FIN Packet 수신을 대기하고 수신 후 SOCKET 이 Close 된다. 이 Process 에서 Sn\_IR[TIMEOUT]이 발생할 수 있다.

```

Passive Close: /* received FIN Packet from Destination */
{
    Sn_CR = DISCON;    /* send FIN Packet */
    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')
    {
        Sn_IRCLR[DISCON] = '1';    /* clear Interrupt */
        goto CLOSED;
    }
    else goto Timeout?;
}

```

```

Active Close : /* sent FIN Packet to Destination */
{
    /* wait until FIN Packet is received */
    while(Sn_IR[DISCON] == '0' and Sn_IR[TIMEOUT] == '0');
    if (Sn_IR[DISCON] == '1')
    {
        Sn_IRCLR[DISCON] = '1';    /* clear Interrupt */
        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]이 발생한다.

**참조) 6.7 Retransmission**

```

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

```

• CLOSE

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

```

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

```

6.2.2 TCP CLIENT

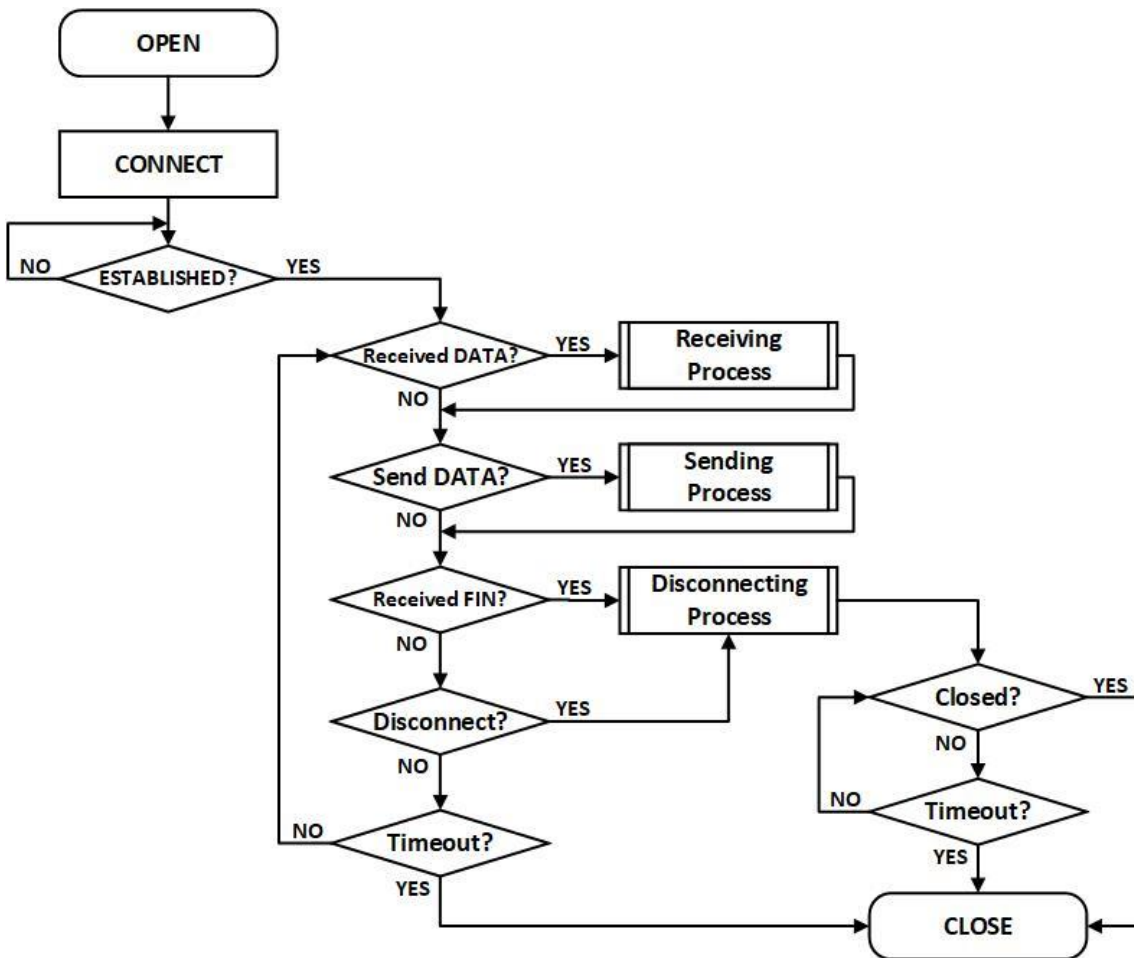


Figure 22 TCP CLIENT Operation Flow

• OPEN

참조) 6.2.1 TCP SERVER



#### • CONNECT

SOCKET n 을 ‘TCP CLIENT’로 동작시킨다.

Sn\_CR[CONNECT] 또는 Sn\_CR[CONNECT6]을 통해 ‘TCP SERVER’로 SYN Packet 을 전송한다.

```

Sn_MR[3:0] = TCP4:
{
    /* 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};

    Sn_CR = CONNECT; /* set CONNECT command in TCP Mode */
    while(Sn_CR != 0x00); /* wait until CONNECT or CONNECT6 command is cleared */
    goto ESTABLISHED?;
}

Sn_MR[3:0] = TCP6:
{
    /* set destination IP address, FE80::10D:FC:34A:EF90 */
    Sn_DIP6R[0:15] = {0xFE, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
                    0x01, 0x0D, 0x00, 0xFC, 0x03, 0x4A, 0xEF, 0x90};
    /* set destination PORT number, 5000(0x1388) */
    Sn_PORTR[0:1] = {0x13, 0x88};

    Sn_CR = CONNECT6; /* set CONNECT6 command in TCP6 Mode */
    while(Sn_CR != 0x00); /* wait until CONNECT or CONNECT6 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 연결이 완료되며, Sn\_IR[CON] 또는 Sn\_SR[SOCK\_ESTABLISHED]로 확인할 수 있다. 접속이 완료되면 TCP4/TCP6 Mode 에 따라 Destination Address 는 Sn\_DIPR 또는 Sn\_DIP6R 을 통해 알 수 있다.

**참조** [6.2.1 TCP SERVER: ESTABLISHED?](#)

#### • Others flow

**참조** [6.2.1 TCP SERVER](#)

### 6.2.3 TCP DUAL

SOCKET 은 W6100 의 Dual Stack(IPv4/IPv6)을 기반으로 TCP Dual(TCPD) Mode 를 제공한다. TCPD Mode 로 OPEN 된 SOCKET 이 Sn\_CR[LISTEN]에 의해 ‘TCP DUAL SERVER’로 동작하는 경우, 접속하는 Destination 의 IP 버전에 따라 TCP4 또는 TCP6 로의 동작이 결정된다. ‘TCP DUAL CLIENT’로 동작하는 경우, Sn\_CR[CONNECT] 또는 Sn\_CR[CONNECT6]을 통해 TCP4 또는 TCP6 로 동작한다.

Destination 과의 접속이 완료되면 HOST 는 Sn\_ESR[TCPM]을 통해 SOCKET 이 TCP4 또는 TCP6 로 동작함을 확인할 수 있다.

### 6.2.3.1 TCP DUAL SERVER

‘TCP DUAL SERVER’의 Operation Flow 는 Figure 21 과 동일하다.

- OPEN

SOCKET n 을 TCPD Mode 로 OPEN 한다.

```
TCP Mode : TCP4, TCP6, TCPD
{
START :
    Sn_MR[3:0] = '1101'; /* set TCPD Mode */

    Sn_PORTR[0:1] = {0x13,0x88}; /* set PORT Number, 5000(0x1388) */
    Sn_CR[OPEN] = '1'; /* set OPEN Command */
    while(Sn_CR != 0x00); /* wait until OPEN Command is cleared */

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

```

- Others flow

[참조\) 6.2.1 TCP SERVER](#)

### 6.2.3.2 TCP DUAL CLIENT

‘TCP DUAL CLIENT’의 Operation Flow 는 Figure 22 와 동일하다.

- OPEN

[참조\) 6.2.3.1 TCP DUAL SERVER: OPEN](#)

- CONNECT

Sn\_CR[CONNECT] 또는 Sn\_CR[CONNECT6]을 통해 ‘TCP SERVER’로 SYN Packet 을 전송하며 TCP4/TCP6 로의 동작이 결정된다.

```
TCP4 :
{
    /* 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};

    Sn_CR = CONNECT; /* set CONNECT command */

    while(Sn_CR != 0x00); /* wait until CONNECT or CONNECT6 command is cleared */
    goto ESTABLISHED?;
}

```

```

TCP6 :
{
  /* set destination IP address, FE80::10D:FC:34A:EF90 */
  Sn_DIP6R[0:15] = {0xFE, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
                    0x01, 0x0D, 0x00, 0xFC, 0x03, 0x4A, 0xEF, 0x90};
  /* set destination PORT number, 5000(0x1388) */
  Sn_PORTR[0:1] = {0x13, 0x88};

  Sn_CR = CONNECT6; /* set CONNECT6 command */

  while(Sn_CR != 0x00); /* wait until CONNECT or CONNECT6 command is cleared */
  goto ESTABLISHED?;
}

```

- Others flow

참조) 6.2.1 TCP SERVER

## 6.2.4 Other Functions

### 6.2.4.1 TCP SOCKET Options

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

- No Delayed ACK : Sn\_MR[ND] = '1'

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

- Delayed ACK : Sn\_MR[ND] = '0'

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

- Force PSH Flag : Sn\_MR [FPSH] = '1'

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

- Auto PSH Flag : Sn\_MR[FPSH] = '0'

Destination 의 Window Size 가 0 이거나, Sn\_CR[SEND]에 의해서 MSS 단위로 나뉘어 전송된 마지막 DATA packet 의 PSH flag 를 설정한다.

- Destination Hardware Address by Sn\_DHAR : Sn\_MR2[DHAM] = '1'

HOST 가 설정한 Sn\_DHAR 을 Destination Hardware Address 로 사용한다.

- Destination Hardware Address by ARP : Sn\_MR2[DHAM] = '0'

'TCP SERVER'인 경우, Destination 으로부터 수신한 SYN Packet 에서  
'TCP CLIENT'인 경우, ARP/ND-process 에서 획득한 Destination Hardware Address 를 사용한다.

- Destination Hardware Address by Sn\_DHAR : Sn\_MR2[FARP] = '1'

---

'TCP SERVER'인 경우, "TCP CLIENT"로부터 수신한 SYN Packet 에 대한 SYN/ACK Packet 을 응답하기 이전에 ARP/ND-process 를 수행하여 획득한 Destination Hardware Address 를 사용한다.

Sn\_MR2[DHAM] = '1' 인 경우 ARP/ND-process 를 수행하지만, Sn\_DHAR 을 Destination Hardware Address 로 사용한다.

- **Destination Hardware Address by Sn\_DHAR : Sn\_MR2[FARP] = '0'**

'TCP SERVER'인 경우, Destination 으로부터 수신한 SYN Packet 에서

'TCP CLIENT'인 경우, ARP/ND-process 에서 획득한 Destination Hardware Address 를 사용한다.

#### 6.2.4.2 Keep Alive

Keep Alive (KA)는 Destination 과의 연결이 유효한지 검사하기 위해 이미 전송한 DATA Packet 의 마지막 1 Byte 를 재전송한다. 따라서 이 기능은 1 Byte 이상의 DATA 를 전송한 경우 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.

## 6.3 UDP

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

아래의 Figure 23 은 UDP Mode SOCKET 의 동작흐름을 나타낸다.

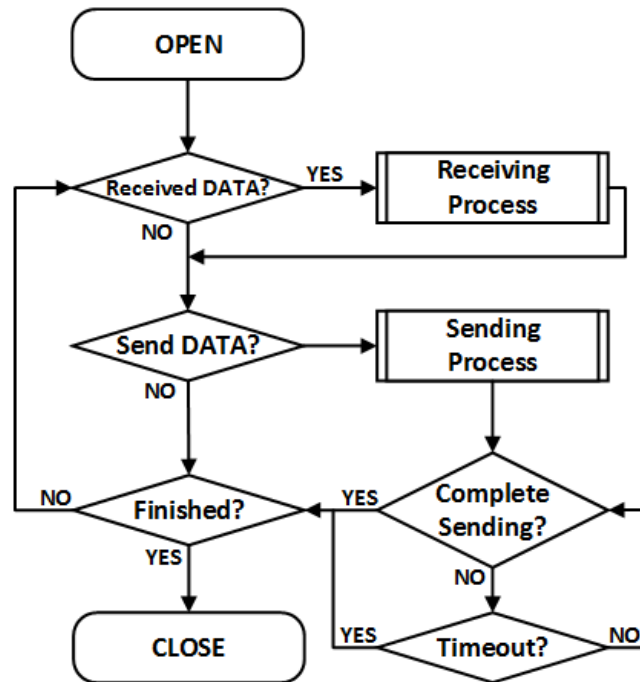


Figure 23 UDP Operation Flow

### 6.3.1 UDP Unicast

UDP Unicast 는 하나의 송신자가 하나의 Destination 에 DATA 를 전송하는 통신방식이다. DATA 전송 전에 UDP SOCKET 은 최초의 Destination 또는 새로운 Destination 에 대해서만 ARP/ND-process 을 수행한다. 이 process 에서 Sn\_IR[TIMEOUT] = '1'이 발생할 수 있다. (참조 6.7 Retransmission)

또한, Sn\_MR2[DHAM] = '1'인 경우, ARP/ND-process 는 생략되고, HOST 가 설정한 Sn\_DHAR 을 Destination Hardware Address 로 사용한다.

UDP Unicast 의 동작흐름은 Figure 23 과 동일하다.

#### • OPEN

SOCKET n 을 UDP4 / UDP6 Mode 로 Open 한다.

UDP4, UDP6 Mode :

```

{
START :
Sn_MR[3:0] = '0010'; /* set UDP4 Mode */
// Sn_MR[3:0] = '1010; /* set UDP6 Mode */

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

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

/* check SOCKET for UDP6 Mode */
if(Sn_SR != SOCK_UDP) goto START;
}

```

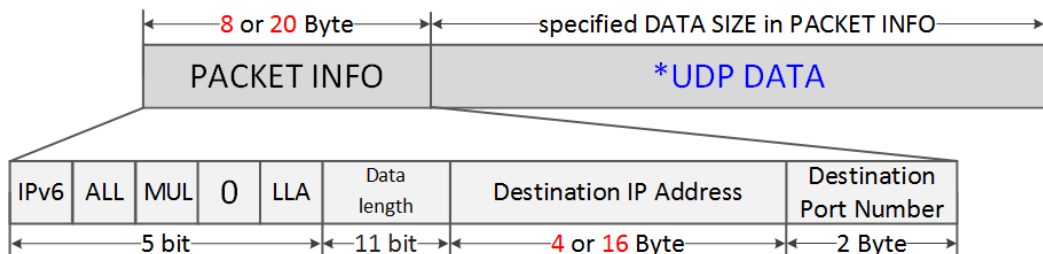
• Received DATA?

참조) 6.2.1 TCP SERVER: Received DATA?

• Receiving Process

UDP 는 하나 이상의 Destination 로부터 DATA Packet 을 수신할 수 있으며, 각 Destination 의 구분을 위해 DATA Packet 은 Figure 24 와 같이 “PACKET INFO”와 함께 SOCKET n RX Buffer 에 저장된다.

HOST 는 SOCKET n RX Buffer 에서 반드시 Figure 24 단위로 Read 해야한다. 수신할 UDP Packet 이 SOCKET n RX Buffer 의 Free Size 보다 크거나 Fragmentation 된 경우 Discard 된다.



\* DATA SIZE is only the size of UDP DATA

Figure 24 Received DATA in UDP Mode SOCKET RX Buffer Block

Table 4 Parameter Description in PACKET INFO

PACKET INFO	Description
IPv6	0 : UDP/IPv4 Packet 수신 1 : UDP/IPv6 Packet 수신
BRD/ALL	0 : others 1 : Broadcast(Allnode) Packet 수신
MUL	0 : others 1 : Multicast Packet 수신
0	always '0'
LLA	0 : GUA

	1 : LLA
DATA Length	*UDP DATA Length 저장
Destination IP Address	UDP4 수신시, Destination IPv4 Address 저장 (4 Byte) UDP6 수신시, Destination IPv6 Address 저장 (16 Byte)
Destination Port Number	Destination Port Number

```

UDP4 Mode :
{
    /* receive PACKINFO */
    goto 6.2.1 TCP SERVER: Receiving Process with get_size = 8 bytes;

    /* extract Destination IP, Port, Size in PACKET INFO */
    data_info = destination_address[0] & "11111000";
    data_size = (destination_address[0] & "00000111" << 8) + destination_address[1];
    if( data_info & '10000000' == 0 ) /* Is Destination IPv4 Address? */
    {
        dest_ip[0:3] = destination_address[2:5];
        dest_port = (destination_address[6] << 8) + destination_address[7];
    }
    /* read UDP DATA */
    goto 6.2.1 TCP SERVER: Receiving Process with get_size = data_size;
}

UDP6 Mode :
{
    /* receive PACKINFO */
    goto 6.2.1 TCP SERVER: Receiving Process with get_size = 20 bytes;

    /* extract Destination IP, Port, Size in PACKET INFO */
    data_info = destination_address[0] & "11111000";
    data_size = (destination_address[0] & "00000111" << 8) + (destination_address[1];
    if( data_info & '10000000' != 0 ) /* Is Destination IPv6 Address? */
    {
        dest_ip[0:15] = destination_address[2:17];
        dest_port = (destination_address[18] << 8) + destination_address[19];
    }
    /* read UDP DATA */
    goto 6.2.1 TCP SERVER: Receiving Process with get_size = data_size;
}

```

• Send DATA? / Sending Process

참조) 6.2.1 TCP SERVER: Send DATA? / Sending Process

```

UDP4 Mode
{
    /* 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};
}

```

```

goto 6.2.1 TCP SERVER: Send DATA? / Sending Process with Sn_CR[SEND];
}
UDP6 Mode
{
/* set destination IP address, FE80::10D:FC:34A:EF90 */
Sn_DIP6R[0:15] = {0xFE, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
                  0x01, 0x0D, 0x00, 0xFC, 0x03, 0x4A, 0xEF, 0x90};
/* set destination PORT number, 5000(0x1388) */
Sn_DPORTR[0:1] = {0x13, 0x88};

goto 6.2.1 TCP SERVER: Send DATA? / Sending Process with Sn_CR[SEND6];
}

```

• Complete Sending? / Timeout?

최초의 Destination 또는 새로운 Destination 으로 DATA Packet 를 전송할 경우, DATA Packet 전송 이전에 ARP/ND-process 을 수행하며, 이 Process 에서 Sn\_IR[TIMEOUT]이 발생할 수 있으며, 해당 DATA packet 은 Discard 된다.

UDP 는 TCP 와 달리 1:N 통신을 지원하므로, Sn\_IR[TIMEOUT] = '1'((참조) 6.7 Retransmission)이 발생하더라도 SOCKET 은 CLOSE 되지 않는다.

```

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

```

• Finished? / CLOSE

더 이상 보낼 DATA 가 없는 경우 Sn\_CR[CLOSE]를 통해 SOCKET 을 CLOSE 한다.

```

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

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

```

### 6.3.2 UDP Broadcast

Broadcast 는 하나의 송신자가 동일 Network 내 다수 Node 들에게 DATA 를 전송하는 통신방식이다.

UDP4 의 경우, 동일 Network 내의 모든 Node 들에게 DATA 를 전송하는 방식(All Node Broadcasting)과, 동일 Network 내 동일한 Subnet 에 속한 Node 들에게만 DATA 를 전송하는 방식(Subnet Broadcasting) 이 있다.



UDP6 의 경우, All-Node Multicast Address 인 FF02::01 를 활용하여 UDP4 의 All-Node Broadcasting 과 동일한 전송 방식을 지원한다.

- OPEN

참조) [6.3.1 UDP Unicast: OPEN](#)

- Received DATA?

참조) [6.2.1 TCP SERVER: Received DATA?](#)

- Receiving Process

참조) [6.3.1 UDP Unicast: Receiving Process](#)

- Send DATA? / Sending Process

UDP4 Broadcasting 과 UDP6 All-Node Multicasting 에 따른 Destination IP Address 를 설정한다.

```

UDP4 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_DPORTR[0:1] = {0x13,0x88};

    goto 6.2.1 TCP SERVER: Sending Process with Sn_CR[SEND];
}

UDP4 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_DPORTR[0:1] = {0x13,0x88};

    goto 6.2.1 TCP SERVER: Sending Process with Sn_CR[SEND];
}

UDP6 All-Node Multicasting :
{
    /* set destination IP address, FF02::01 */
    Sn_DIP6R[0:15] = {0xFF, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
                    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01};
    /* set Destination PORT Number, 5000(0x1388) */
    Sn_DPORTR[0:1] = {0x13,0x88};

    goto 6.2.1 TCP SERVER: Sending Process with Sn_CR[SEND];
}

```

- Complete sending? / Timeout?

참조) [6.3.1 UDP Unicast: Complete Sending? / Timeout?](#)

- Finished? / CLOSE

참조) [6.3.1 UDP Unicast: Finished? / CLOSE](#)

### 6.3.3 UDP Multicast

UDP Multicast는 하나의 송신자가 하나의 Multicast Group에 DATA를 전송하는 통신방식이다. IPv4의 경우, Multicast-Group Address 범위는 224.0.0.0 ~ 239.255.255.255이며 (참조 : [IANA Multicast Address](#)) 또한 이에 대응되는 Hardware Address는 01:00:5E:00:00:00 ~ 01:00:5E:7F:FF:FF이다. Multicast Hardware Address 설정 시 하위 23 Bits는 Multicast-Group Address와 동일해야 한다. (참조, [rfc1112](#))

IPv6의 경우, 아래 Figure 25과 같이 Multicast-Group Address를 지정한다.

UDP Multicast의 동작흐름은 Figure 23과 동일하다.

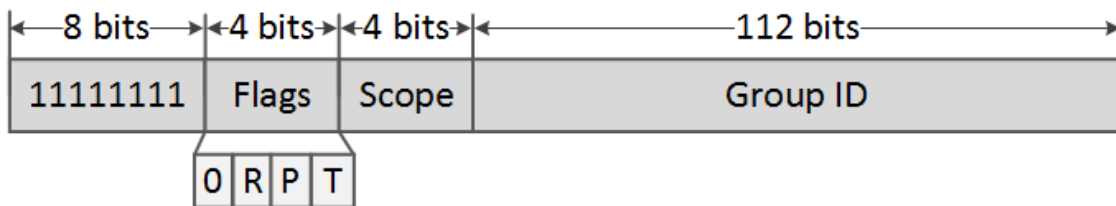


Figure 25 IPv6 Multicast-Group Address Format

Table 5 Parameters of Flags in IPv6 Multicast Address

Flags	Description
0	always '0'
R	1 : Embedded RP 0 : No embedded RP * If R = '1', P must be '1'
P	1 : Based on Unicast Network Prefix 0 : Not based on Unicast Network Prefix * If P = '1', T must be '1'
T	1 : Temporary address(Local assigned) 0 : Permanent address(IANA assigned)

Table 6 Definition of Scope in IPv6 Multicast Address

Scope	Description
1	Node
2	Link
3	Subnet
4	Admin
5	Site
8	Organization
E	Global

• OPEN

Multicast-Group 정보와 Sn\_MR[MULTI]를 설정한 후, Sn\_CR[OPEN]을 수행한다.

UDP4 Multicast Mode 의 경우 IGMPv1 또는 IGMPv2 를 통해, UDP6 Multicast Mode 의 경우 MLDv1 을 통해 Multicast-Group 에 Join 한다.

```

UDP4 Multicast Mode :
{
START :
    /* set Multicast-Group Hardware 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};

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

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

    goto 6.3.1 UDP Unicast : OPEN(UDP Mode)
}

UDP6 Multicast Mode :
{
START :
    /* set Multicast-Group Hardware Address, 33:33:00:AB:34:56 */
    Sn_DHAR[0:5] = {0x33, 0x33, 0x00, 0xAB, 0xCD, 0xEF};
    /* set Multicast-Group IP Address, FF02::100:00AB:CDEF */
    Sn_DIP6R[0:15] = {0xFF, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
                    0x00, 0x00, 0x00, 0x01, 0x00, 0xAB, 0xCD, 0xEF};
    /* set Multicast-Group PORT Number, 3000(0x0BB8) */
    Sn_DPORTR[0:1] = {0x0B, 0xB8};

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

    goto 6.3.1 UDP Unicast : OPEN(UDP6 Mode)
}

```

- Received DATA?

*참조) 6.2.1 TCP SERVER: Received DATA?*

- Receiving Process

*참조) 6.3.1 UDP Unicast: Receiving Process*

- Send DATA? / Sending Process

*참조) 6.3.1 UDP Unicast: Sending Process*

- Complete sending? / Timeout?

*참조) 6.3.1 UDP Unicast: Complete Sending? / Timeout?*

- Finished? / CLOSE

*참조) 6.3.1 UDP Unicast: Finished? / CLOSE*

### 6.3.4 UDP DUAL

UDPD Mode 로 OPEN 된 SOCKET 은 UDP4/UDP6 Packet 을 모두 송수신할 수 있다.

UDP4 DATA 와 UDP6 DATA 송신은 각각 Sn\_CR[SEND]와 Sn\_CR[SEND6]으로 송신가능하며, 또한 UDP4 DATA 와 UDP6 DATA 수신된 Packet 의 PACKET INFO 를 통해 구분할 수 있다.

UDPD 의 동작흐름은 Figure 23 과 동일하다.

- OPEN

HOST 는 SOCKET n 을 UDPD Mode 로 설정한다.

```

UDP6 Mode
{
START :
    Sn_MR[3:0] = '1110'; /* set UDPD Mode */

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

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

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

- Received DATA?

*참조) 6.2.1 TCP SERVER: Received DATA?*

- Receiving Process

UDPD Mode SOCKET 은 하나 이상의 Destination 으로부터 UDP4/UDP6 Packet 을 수신할 수 있으며, 수신된 DATA Packet 은 'PACKET INFO'와 함께 SOCKET n RX Buffer 에 저장된다. HOST 는 'PACKET INFO'를 통해 수신된 DATA Packet 의 IP Version, 전송방식 및 Destination 정보를 확인한다.

HOST 는 SOCKET n RX Buffer 에서 반드시 Figure 24 에 표시된 것과 같은 단위로 Read 해야한다.

수신할 UDP Packet 이 SOCKET n RX Buffer 의 Free Size 보다 크거나 Fragmentation 된 경우, 해당 Packet 은 Discard 된다.

**\*'PACKET INFO' 에 대한 자세한 설명은 Figure 24 와 Table 4 를 참조**

```

{
    
```

```

/* extract upper 2 bytes in PACKINFO */
goto 6.2.1 TCP SERVER: Receiving Process with get_size = 2 bytes;

data_size = (destination_address[0] & "00000111" << 8) + destination_address[1];

/*
check UDP4 or UDP6 DATA Packet, extract Destination IP, Port, Size in PACKE INFO
*/
if(destination_address[0] & "10000000" == 0) /* UDP4 DATA Packet */
{
goto 6.2.1 TCP SERVER: Receiving Process with get_size = 6 bytes;
dest_ip[0:3] = destination_address[0:3];
dest_port = (destination_address[4] << 8) + destination_address[5];
}
else /* UDP6 DATA Packet */
{
goto 6.2.1 TCP SERVER: Receiving Process with get_size = 18 bytes;
dest_ip[0:15] = destination_address[0:15];
dest_port = (destination_address[16] << 8) + destination_address[17];
}
/* read UDP DATA */
goto 6.2.1 TCP SERVER: Receiving Process with get_size = data_size;
}

```

- Send DATA? / Sending Process  
참조) 6.3.1 UDP Unicast : Send Data? / Sending Process
- Complete sending? / Timeout?  
참조) 6.3.1 UDP Unicast: Complete Sending? / Timeout?
- Finished? / CLOSE  
참조) 6.3.1 UDP Unicast: Finished? / CLOSE

### 6.3.5 Other Functions

#### 6.3.5.1 UDP Mode SOCKET Options

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

- Destination Hardware Address by Sn\_DHAR : Sn\_MR2[DHAM]= '1'  
ARP/ND-process 을 생략하고 Sn\_DHAR 에 저장된 Destination Hardware Address 를 사용해 UDP Packet 을 전송한다.
- Destination Hardware Address by ARP : Sn\_MR2[DHAM]= '0'  
ARP/ND-process 을 통해 획득한 Destination Hardware Address 를 사용하여 UDP packet 을 전송한다.
- Force ARP : Sn\_MR2[FARP]= '1'

Sn\_CR[SEND] 혹은 Sn\_CR[SEND6]에 의한 UDP Packet 전송 시 마다 ARP/ND-process 를 수행한다.

Sn\_MR2[DHAM] = '1' 인 경우 ARP/ND-process 를 수행하지만, Sn\_DHAR 을 Destination Hardware Address 로 사용한다.

• **Auto ARP : Sn\_MR2[FARP]= '0'**

ARP/ND-process 를 첫 번째 UDP DATA Packet 을 전송하거나 Destination 이 변경될 경우에만 수행한다.

### 6.3.5.2 UDP Block

UDP SOCKET n (Sn\_MR[MULTI] = '0')은 기본적으로 Unicasting, Broadcasting Packet 수신을 지원하지만, Sn\_MR[BRDB] = '1'인 경우, Broadcasting Packet 을 Block 한다.

UDP SOCKET n 이 Multicast 통신방식을 사용하는 경우(Sn\_MR[MULTI] = '1'), 기본적으로 Unicasting, Broadcasting, Multicasting Packet 수신을 지원하지만, Sn\_MR[UNIB] = '1' 인 경우, Unicasting Packet 을 Block 하며, Sn\_MR[BRDB] = '1'인 경우, Broadcasting Packet 을 Block 한다. UDP Block Options 은 Sn\_CR[OPEN] = '1' 이전에 설정한다.

Sn_MR[MULTI]	Sn_MR[BRDB]	Sn_MR[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

UDP6Mode 이고 Sn\_MR[SMB] = '1'인 경우, 자신이 속한 Solicited Multicast Packet 을 Block 한다.

### 6.3.5.3 Port Unreachable Block

Destination 이 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.

UDP4 인 경우는 NET4MR [UNRB] = '1', UDP6 인 경우는 NET6MR[UNRB] = '1'을 통해 Port Unreachable Packet 전송을 각각 Block 한다.

## 6.4 IPRAW

IPRAW 는 Internet Protocol Layer 에 정의된 다양한 상위 Protocol(참조 [IANA Protocol Numbers](#))중 Table 7 에서 보여주는 Protocol 통신을 지원한다.

SOCKET n 은 IPRAW4 또는 IPRAW6 Mode 로 설정될 수 있으며 Sn\_PNR 를 통해 각각 IPv4 Protocol field 와 IPv6 Extension Header 의 값을 설정할 수 있다. SOCKET n 은 Sn\_PNR 에 설정된 Protocol 이외의 다른 Protocol 을 사용하여 통신할 수 없다.

Table 7 Internet Protocol Supported In IPRAW Mode

Protocol	Number	Semantic	Support
HOPOPT	0	IPv6 Hop-by-Hop Option	O
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	41	IPv6 encapsulation	O
ICMP6	58	ICMP for IPv6	O
others	-	Other Protocols	O

IPRAW4 Mode 에서 Sn\_PNR = ICMP 인 경우, Destination 의 PING-Request 에 대한 Auto PING Reply 를 더 이상 지원하지 않으며, PING-Request Packet 은 IPRAW SOCKET n RX Buffer Block 으로 저장된다.

IPRAW6 Mode 에서 Sn\_PNR = ICMP6 인 경우, Echo Request, NA(Neighbor Advertisement), NS(Neighbor Solicitation), RA(Router Advertisement)에 대한 W6100 의 Auto Reply Packet 전송은 ICMP6BLKR 설정을 통해 각각의 Packet 을 Block 할 수 있으며, Block 된 Packet 의 경우 SOCKET n RX Buffer 에 저장되지 않는다.

Figure 26 은 IPRAW4/IPRAW6 Mode 일때, Socket n 의 동작흐름이다.

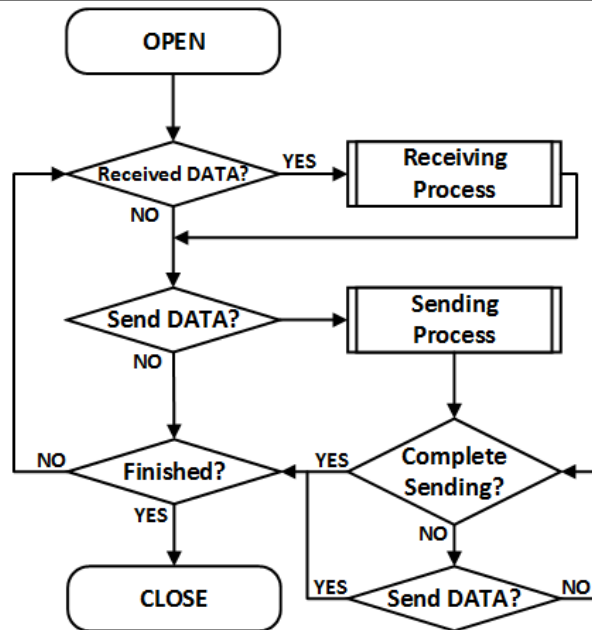


Figure 26 IPRAW Operation Flow

• OPEN

SOCKET n 을 IPRAW4/IPRAW6 Mode 로 설정한다.

IPRAW4 Mode :

```

{
START :
  Sn_PNR = protocol_num; /* set Protocol Number */

  Sn_MR[3:0] = '0011'; /* set IPRAW4 Mode */

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

  /* check SOCKET for IPRAW6 Mode */
  if(Sn_SR != SOCK_IPRAW6) goto START;
}
  
```

IPRAW6 Mode :

```

{
START :
  Sn_PNR = protocol_num; /* set Protocol Number(Next Header) */

  Sn_MR[3:0] = '1001'; /* set IPRAW6 Mode */

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

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

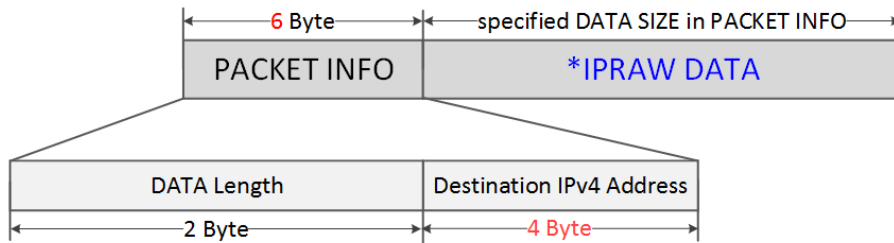
• Received DATA?

[참조\) 6.2.1 TCP SERVER: Received DATA?](#)



• Receiving Process

IPRAW4/IPRAW6 Mode 는 하나 이상의 Destination 으로부터 IP DATA Packet 을 수신할 수 있으며, 각 Destination 의 구분을 위해 DATA Packet 은 Figure 27 또는 Figure 28 과 같이 “PACKET INFO”와 함께 SOCKET n RX Buffer Block 에 저장된다. “PACKET INFO”는 IPRAW4/IPRAW6 Mode 에 따라서 Size 및 Parameters 가 다르며 해당 내용은 Table 8 과 Table 9 에 정의되어 있다. SOCKET n RX Buffer 의 Free Size 보다 큰 Packet 은 Discard 되며 HOST 는 반드시 Figure 27 또는 Figure 28 단위로 Read 해야 한다.

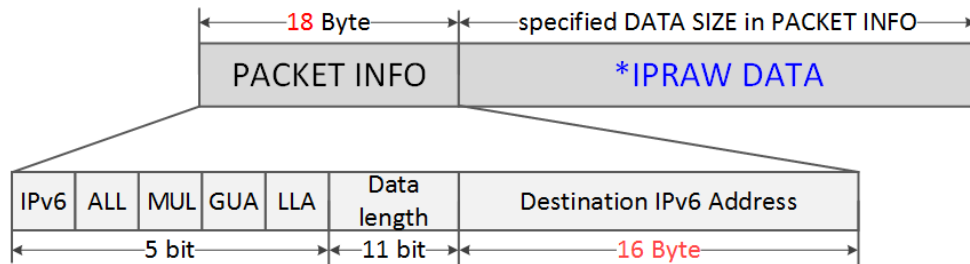


\* IPRAW DATA is only the size of DATA in Received Packet

Figure 27 Received DATA in IPRAW4 Mode SOCKET RX Buffer Block

Table 8 parameters of ‘PACKET INFO’ in IPRAW4 Mode

PACKET INFO	Description
DATA Length	*IPRAW DATA Length 저장
Destination IPv4 Address	Destination IPv4 Address (4 Byte)



\* IPRAW DATA is only the size of DATA in Received Packet

Figure 28 Received DATA in IPRAW6 Mode SOCKET RX Buffer Block

Table 9 parameters of ‘PACKET INFO’ in IPRAW6 Mode

PACKET INFO	Description
IPv6	IPv6 수신한 경우, ‘1’로 설정
ALL	All Node Packet 수신한 경우, ‘1’로 설정
MUL	Multicast Packet 수신한 경우, ‘1’로 설정
GUA	Destination Address 가 GUA 인 경우, ‘1’로 설정
LLA	Destination Address 가 LLA 인 경우, ‘1’로 설정
DATA Length	*IPRAW DATA Length 저장

```

IPRAW4 Mode :
{
  /* receive PACKINFO */
  goto 6.2.1 TCP SERVER: Receiving Process with get_size = 6;

  /* extract Destination DATA Size, IP Address in PACKET INFO*/
  data_size = (destination_address[0] << 8) + destination_address[1];
  dest_ip[0:3] = destination_address[2:5];

  /* read UDP DATA */
  goto 6.2.1 TCP SERVER: Receiving Process with get_size = data_size;
}

IPRAW6 Mode :
{
  /* receive PACKINFO */
  goto 6.2.1 TCP SERVER: Receiving Process with get_size = 18;

  /* extract Destination Information, DATA Size, IP Address in PACKET INFO */
  data_Info = destination_address[0] & "11111000";
  data_size = (destination_address[0] & "00000111" << 8) + (destination_address[1];
  dest_ip[0:15] = destination_address[2:17];

  /* read UDP DATA */
  goto 6.2.1 TCP 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 단위로 직접 나누어 전송한다. IPRAW6 Mode 의 MSS 는 1460 보다 크게 설정될 수 없고, IPRAW Mode 의 MSS 는 1480 보다 크게 설정될 수 없다.

```

IPRAW4 Mode :
{
  /* set Destination IP Address, 192.168.0.11 */
  Sn_DIPR[0:3] = {0xC0, 0xA8, 0x00, 0x0B};

  goto 6.2.1 TCP SERVER: Sending Process;
}

IPRAW6 Mode :
{
  /* set destination IP address, FE80::10D:FC:34A:EF90 */
  Sn_DIP6R[0:15] = {0xFE, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
                  0x01, 0x0D, 0x00, 0xFC, 0x03, 0x4A, 0xEF, 0x90};

  goto 6.2.1 TCP SERVER: Sending Process;
}

```

• **Complete sending? / Timeout?**

최초 Destination 으로의 전송이나, 이전 Destination 과 다른 Destination 으로 DATA Packet 를 전송할 경우, DATA Packet 전송 이전에 ARP/ND-process 을 수행하며, 이 과정에서 Sn\_IR[TIMEOUT] 이 발생할 수 있으며 해당 DATA packet 은 Discard 된다. IPRAW4/IPRAW6 는 UDP 처럼 1:N 통신을 지원하므로, Sn\_IR[TIMEOUT](참조 6.7 Retransmission)이 발생하더라도 SOCKET 은 CLOSE 되지 않는다.

```

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

```

• **Finished? / CLOSE**

더 이상 보낼 DATA 가 없는 경우 Sn\_CR[CLOSE]로 SOCKET 을 CLOSE 할 수 있다.

```

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

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

```

## 6.4.1 Other Functions

### 6.4.1.1 IPRAW Mode SOCKET Options

IPRAW4/6 Mode SOCKET 을 OPEN 하는 과정에서 Sn\_MR 과 Sn\_MR2 를 통해서 SOCKET Option 을 설정한다.

- **Destination Hardware Address by Sn\_DHAR : Sn\_MR2[DHAM]= '1'**  
ARP/ND-process 을 생략하고 Sn\_DHAR 에 저장된 Destination Hardware Address 를 사용해 IPRAW4 또는 IPRAW6 Packet 을 전송한다.
- **Destination Hardware Address by ARP : Sn\_MR2[DHAM]= '0'**  
ARP/ND-process 을 통해 획득한 Destination Hardware Address 를 사용하여 IPRAW4 또는 IPRAW6 packet 를 전송한다.
- **Force ARP : Sn\_MR2[FARP]= '1'**  
Sn\_CR[SEND] 혹은 Sn\_CR[SEND6]에 의한 IPRAW4 또는 IPRAW6 DATA 전송 시 마다 ARP/ND-process 을 수행한다.

Sn\_MR2[DHAM] = '1' 인 경우 ARP/ND-process 를 수행하지만, Sn\_DHAR 을 Destination Hardware Address 로 사용한다.

• Auto ARP : Sn\_MR2[FARP]= '0'

ARP/ND-process 를 첫 번째 IPRAW DATA Packet 을 전송하거나 Destination 이 변경될 경우에만 수행한다.

### 6.5 MACRAW

MACRAW Mode 는 Ethernet MAC 을 이용한 DATA 통신을 제공하며 오직 SOCKET 0 인 경우, 사용 가능하다.

Sn\_MR[MF] = '0' 인 경우, Ethernet PHY 로 수신된 모든 Ethernet Packet 을 수신한다.

Sn\_MR[MF] = '1'인 경우, Destination Hardware Address 가 Broadcast, Multicast, Source Hardware Address(SHAR)인 Packet 만 수신한다.

Figure 29 는 MACRAW SOCKET 0 의 동작흐름을 도식화한다.

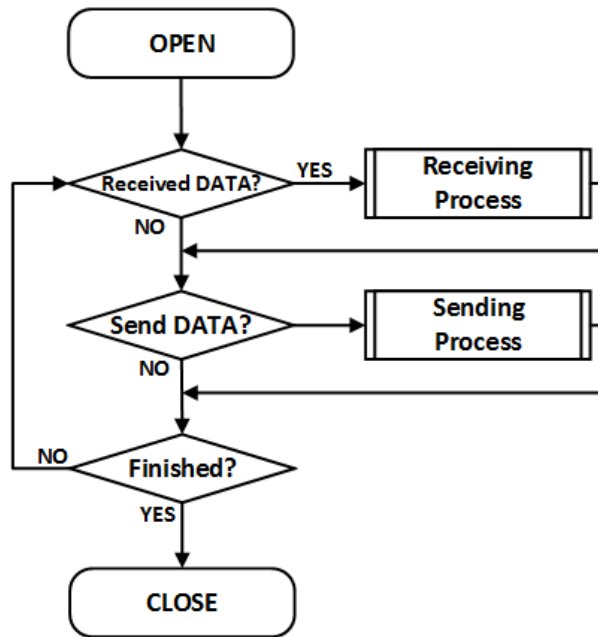


Figure 29 MACRAW Operation Flow

• OPEN

SOCKET 0 을 MACRAW Mode 로 설정한다.

```

{
START :
  SO_MR[3:0] = '0111'; /* set MACRAW Mode */
}
  
```

```

/* MACRAW SOCKET Options */
/* SO_MR[MF] = '1';          // enable MAC Filter
   SO_MR[UNIB] = '1';       // Broadcast Packet Block
   SO_MR[MMB] = '1';       // Multicast Packet Block
   SO_MR[MMB6] = '1';      // IPv6 Packet Block */

SO_CR = OPEN; /* set OPEN Command */
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) 6.2.1 TCP SERVER: Received DATA?

• Receiving Process

MACRAW 는 하나 이상의 Destination 으로부터 DATA Packet 을 수신할 수 있으며, 각 Destination 의 구분을 위해 DATA Packet 은 Figure 30 과 같이 “PACKET INFO”와 함께 SOCKET 0 RX Buffer Block 에 저장된다. HOST 는 반드시 Figure 30 단위로 Read 한다

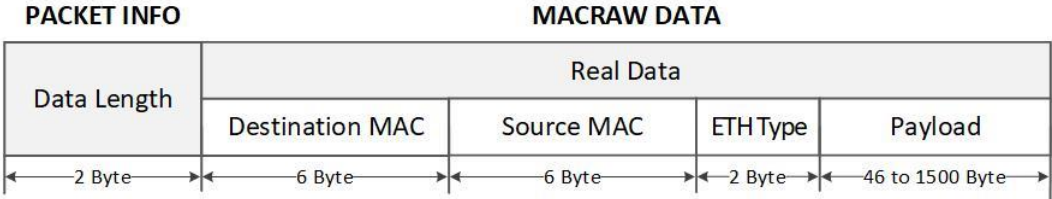


Figure 30 Received DATA Format in MACRAW

```

{
/* receive PACKINFO */
goto 6.2.1 TCP SERVER: Receiving Process with get_size = 2;

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

/* read MACRAW DATA */
goto 6.2.1 TCP 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 60Bytes 보다 작은 경우 60Bytes 로 Zero Padding 되어 전송된다.

참조) 6.2.1 TCP SERVER: Send DATA? / Sending Process

• Finished? / CLOSE

참조) 6.3.1 UDP Unicast: Finished? / CLOSE

## 6.6 SOCKET-less Command (SLCR)

SLCR Command 는 SOCKET 사용없이 ARP Request, PING Request, NS, RS 등과 같은 Request Packet 을 전송한다. 전송된 Packet 에 대한 응답은 SLIR 을 통해 확인할 수 있으며 설정한 재전송 시간 동안 응답이 없는 경우 SLIR[TOUT]이 발생한다. [참조\) 6.7 Retransmission](#)

SLCR 의 Command 는 동시에 수행될 수 없으며, 반드시 해당 Command 에 대한 SLIR 이 발생 하거나 SLIR[TOUT]이 발생한 뒤 다음 Command 를 수행한다.

Figure 31 은 SOCKET-less Command 의 동작흐름을 나타낸다.

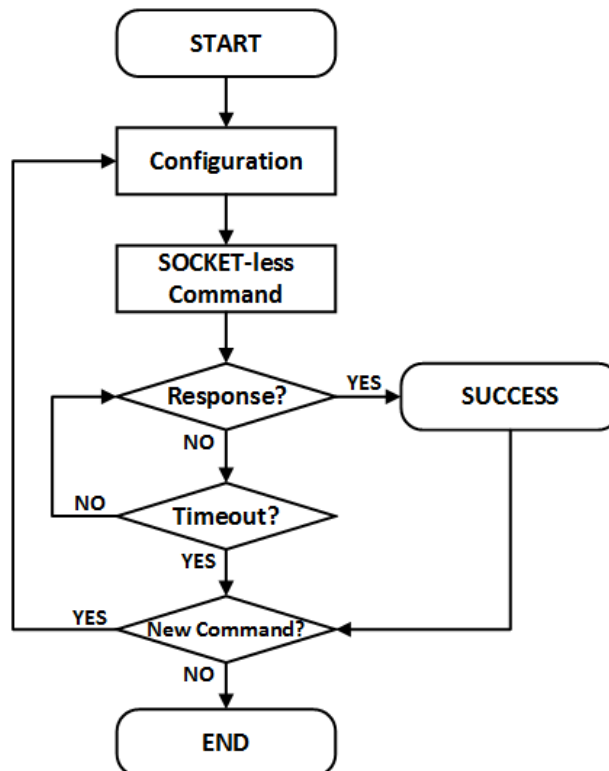


Figure 31 SOCKET-less Command Operation Flow

### 6.6.1 ARP

SLCR[ARP]는 SLDIPR 에 설정된 Destination IP Address 로 ARP Request Packet 을 전송한다. Destination 로부터 ARP Reply 가 수신된 경우, SLIR[ARP]가 설정되고 해당 Destination Hardware Address 는 SLDHAR 에 저장된다. 만약 설정한 재전송 시간 동안 응답을 수신하지 못한 경우 SLIR[TOUT]이 발생한다. [참조\) 6.7 Retransmission](#)

#### • Configuration

SLRTR 과 SLRCR 을 통해 재전송 시간을 설정하고, SLIMR 의 ARP 및 TOUT Interrupt Mask 를 설정하고, SLDIPR 에 Destination IP Address 를 설정한다.

```

{
START :
    /* set SOCKET-less Retransmission Time, 100ms(0x03E8) (Unit 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[TOUT] = '1'; /* TIMEOUT Interrupt Mask Bit */

/* set Destination IP Address, 192.168.0.100 */
SLDIP6R[12:15] = {0xC0, 0xA8, 0x00, 0x64};
}

```

- **SOCKET-less Command**

ARP Command 를 통해 ARP Request Packet 을 전송한다.

```

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

```

- **Response?**

Destination 으로부터 ARP Reply Packet 을 수신한 경우 SLIR[ARP]가 설정된다.

```

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

```

- **Timeout?**

Destination 으로부터 설정된 재전송 시간 동안 ARP Reply Packet 을 수신하지 못한 경우 SLIR[TOUT]이 발생한다.

```

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

```

- **SUCCESS**

Destination Hardware Address 는 SLDHAR 에 저장된다.

```

{

```

```
dst_haddr[0:5] = SLDHAR[0:5]; /* get Destination Hardware Address */
goto END;
}
```

### 6.6.2 PING

SLCR[PING]은 SLDIPR 에 설정된 Destination IP Address 로 ARP 및 PING Request Packet 을 전송한다. ARP Reply Packet 와 PING Reply Packet 을 수신한 경우 SLIR[PING]가 설정되고 해당 Destination Hardware Address 는 SLDHAR 에 저장된다.

만약 설정된 재전송 시간 동안 ARP 와 PING Request Packet 을 수신하지 못한 경우 SLIR[TOUT]이 발생한다. [참조\) 6.7 Retransmission](#)

#### • Configuration

SLRTR 과 SLRCR 을 통해 재전송 시간을 설정하고, SLIMR 의 PING 및 TOUT Interrupt Mask 를 설정하고, SLDIPR 에 Destination IP Address 를 설정한다.

또한 PINGSEQR 과 PINGIDR 을 통해 PING Request Packet 의 Sequence Number 와 ID 를 설정한다.

```
{
START :
    /* set SOCKET-less Retransmission Time, 100ms(0x03E8)(Unit 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[TOUT] = '1'; /* TIMEOUT Interrupt Mask Bit */

    /* set Destination IP Address, 192.168.0.100 */
    SLDIP6R[12:15] = {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.

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

#### • Response?



Destination 으로부터 PING Reply Packet 을 수신한 경우 SLIR[PING]이 설정된다.

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

- Timeout? / SUCCESS

[참조\) 6.6.1 ARP Timeout? / SUCCESS](#)

### 6.6.3 ARP6 (ND, Neighbor Discovery)

SLCR[ARP6]는 SLDIP6R 에 설정된 Destination IP Address 로 ICMPv6 NS(Neighbor Solicitation) Packet 을 전송하며, 이는 ARP-process 기능과 동일하다. Destination 로부터 NA(Neighbor Advertisement) Packet 을 수신한 경우, SLIR[ARP6]가 설정되고 해당 Destination Hardware Address 는 SLDHAR 에 저장된다. 만약 설정된 재전송 시간 동안 응답을 수신하지 못한 경우 SLIR[TOUT]이 발생한다. [참조\) 6.7 Retransmission](#)

- Configuration

SLRTR 과 SLRCR 을 통해 재전송 시간을 설정하고, SLIMR 의 ARP6 및 TOUT Interrupt Mask 를 설정하고, SLDIP6R 에 Destination IP Address 를 설정한다.

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

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

  /* set Interrupt Mask Bit */
  //SLIMR[ARP6] = '1'; /* ARP6 Interrupt Mask Bit */
  //SLIMR[TOUT] = '1'; /* TIMEOUT Interrupt Mask Bit */

  /* set Target IP Address, FE80::1D0:AABB:CCDD */
  SLDIP6R[0:15] = { 0xFE, 0x80, 0x00, 0x01, 0x00, 0x00, 0x00, 0x00,
                  0x13, 0x22, 0x33, 0xFF, 0xFE, 0xAA, 0xBB, 0xCC };
}
```

- SOCKET-less Command

ARP6 Command 를 통해 NS(Neighbor Solicitation) Packet 을 전송한다.

```
{
  SLCR[ARP6] = '1'; /* set ARP6 Command */
}
```

```
while(SLCR != 0x00) ; /* Wait until ARP6 Command is completed*/
}
```

- **Response?**

Destination 으로부터 NA(Neighbor Advertisement) Packet 을 수신한 경우 SLIR[ARP6]가 설정된다.

```
{
    /* check ND Interrupt */
    if(SLIR[ARP6] == '1') /* received NA Packet */
    {
        SLIRCLR[ARP6] = '1'; /* clear Interrupt */
        goto SUCCESS;
    }
    else goto Timeout;
}
```

- **Timeout?**

Destination 으로부터 설정된 재전송 시간 동안 NA Packet 을 수신하지 못한 경우 SLIR[TOUT]이 발생한다.

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

- **SUCCESS**

Destination Hardware Address 는 SLDHAR 에 저장된다.

```
{
    dst_haddr[0:5] = SLDHAR[0:5]; /* get Destination Hardware Address */
    goto END;
}
```

#### 6.6.4 PING6 (ICMPv6 Echo)

SLCR[PING6]은 SLDIP6R 에 설정된 Destination IP Address 로 ICMPv6 NS(Neighbor Solicitation) 및 ICMPv6 Echo Request Packet 을 전송한다. ICMPv6 NA(Neighbor Advertisement) 와 ICMPv6 Echo Reply Packet 을 수신한 경우 SLIR[PING6]가 설정되고, 해당 Destination Hardware Address 는 SLDHAR 에 저장된다.

만약 설정된 재전송 시간 동안 ICMPv6 NA 와 ICMPv6 Echo Reply Packet 을 수신하지 못한 경우 SLIR[TOUT]이 발생한다. [참조 6.7 Retransmission](#)

- **Configuration**

SLRTR 과 SLRCR 을 통해 재전송 시간을 설정하고, SLIMR 의 PING6 및 TIMEOUT Interrupt Mask 를 설정하고, SLDIP6R 에 Destination IP Address 를 설정한다.

```

{
START :
    /* set SOCKET-less Retransmission Timer, 100ms(0x03E8) (Unit 100us) */
    //SLRTR[0:1] = {0x03, 0xE8};
    /* set SOCKET-less Retransmission Counter, 5 */
    //SLRCR = 0x05;

    /* set Interrupt Mask Bit */
    //SLIMR[PING6] = '1'; // PING6 Interrupt Mask Bit
    //SLIMR[TOUT] = '1'; // TIMEOUT Interrupt Mask Bit
}

/* set Destination IP Address, FE80::1D0:AABB:CCDD */
SLDIP6R[0:15] = { 0xFE, 0x80, 0x00, 0x01, 0x00, 0x00, 0x00, 0x00,
                 0x13, 0x22, 0x33, 0xFF, 0xFE, 0xAA, 0xBB, 0xCC };

```

- **SOCKET-less Command**

PING6 Command 를 통해 NS Packet 및 Echo Request Packet 을 전송한다.

```

{
    SLCR[PING6] = '1'; /* set PING6 Command */
    while(SLCR != 0x00) ; /* Wait until PING6 Command is completed*/
}

```

- **Response?**

Destination 으로부터 Echo Reply Packet 을 수신한 경우 SLIR[PING6]가 설정된다.

```

{
    /* check PING6 Interrupt */
    if(SLIR[PING6] == '1') /* received PING6 Packet */
    {
        SLIRCLR[PING6] = '1'; /* clear Interrupt */
        goto SUCCESS;
    }
    else goto Timeout?;
}

```

- **Timeout? / SUCCESS**

[참조\) 6.6.3 ARP6](#) (ND, Neighbor Discovery) [Timeout? / SUCCESS](#)

### 6.6.5 DAD (Duplicate Address Detection)

SLCR[NS]는 SLDIP6R 에 설정된 Destination IP Address 로 DAD(Duplicate Address Detection) 메커니즘을 수행한다.

NS Command 를 통해 DAD NS Packet 을 전송하고 임의의 Destination 으로부터 DAD NA Packet 을 수신하여 SLIR[NS]가 설정된 경우, Destination IP Address 는 이미 사용되는 Address 이므로 Source IPv6 Address 로 사용 할 수 없다.

만약 설정된 재전송 시간 동안 응답을 받지 못하고 SLIR[TOUT]이 발생한 경우, Destination IP Address 를 Source IPv6 Address 로 사용할 수 있다. [참조\) 6.7 Retransmission](#)

아래의 Figure 32 는 DAD Operation Flow 를 보여준다.

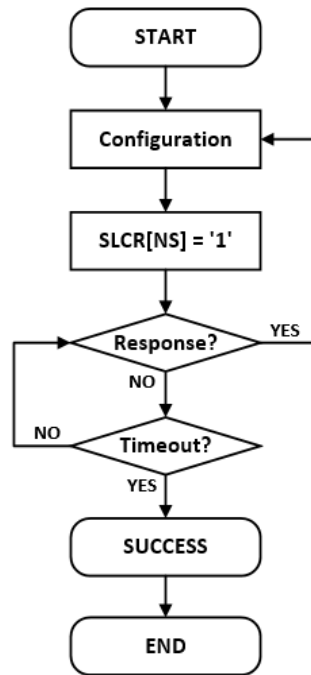


Figure 32 DAD Operation Flow

• **Configuration**

SLRTR 과 SLRCR 을 통해 재전송 시간을 설정하고, SLIMR 의 NS 및 TOUT Interrupt Mask 를 설정하고,

SLDIP6R 에 Destination IP Address 를 설정한다.

```

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

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

  /* set Interrupt Mask Bit */
  //SLIMR[NS] = '1'; /* NS Interrupt Mask Bit */
  //SLIMR[TOUT] = '1'; /* TIMEOUT Interrupt Mask Bit */
}
  
```

```
/* set Target IP Address, FE80::1D0:AABB:CCDD */
SLDIP6R[0:15] = { 0xFE, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
                 0x00, 0x00, 0x01, 0xD0, 0xAA, 0xBB, 0xCC, 0xDD };
}
```

- **SOCKET-less Command**

NS Command 를 통해 DAD NS Packet 을 전송한다.

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

- **Response?**

Destination 으로부터 DAD NA Packet 을 수신한 경우 SLIR[NS]가 설정되며, SLDIP6R 에 설정한 IP Address 를 Source IPv6 Address 로 사용할 수 없다.

```
{
  /* check NS Interrupt */
  if(SLIR[NS] == '1') /* received DAD NA Packet */
  {
    SLIRCLR[NS] = '1'; /* clear Interrupt */
    goto Configuration;
  }
  else goto Timeout?;
}
```

- **Timeout?**

Destination 으로부터 설정된 재전송 시간 동안 DAD NA Packet 을 수신하지 못한 경우 SLIR [TOUT]이 발생한다.

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

- **SUCCESS**

SLDIP6R 에 설정한 Destination IP Address 를 Source Link Local Address 로 사용할 수 있다.

```
{
  LLAR[0:15] = SLDIP6R[0:15]; /* get Source Link-Local Address */
  goto END;
}
```

### 6.6.6 RS (Router Solicitation)

SLCR[RS]는 Link Local All-Router Multicast Address(FF02::2)로 RS(Router Solicitation) Packet 을 전송한다. Router 로부터 RA(Router Advertisement) Packet 을 수신한 경우 SLIR[RS]가 설정되고 RA Packet 의 Prefix Length, Flags, Valid Lifetime, Prefix Lifetime, Prefix Address 는 각각 PLR, PFR, VLTR, PLTR, PAR 에 저장된다. 만약 설정된 재전송 시간 동안 응답을 수신하지 못한 경우 SLIR[TOUT]이 발생한다. [참조\) 6.7 Retransmission](#)

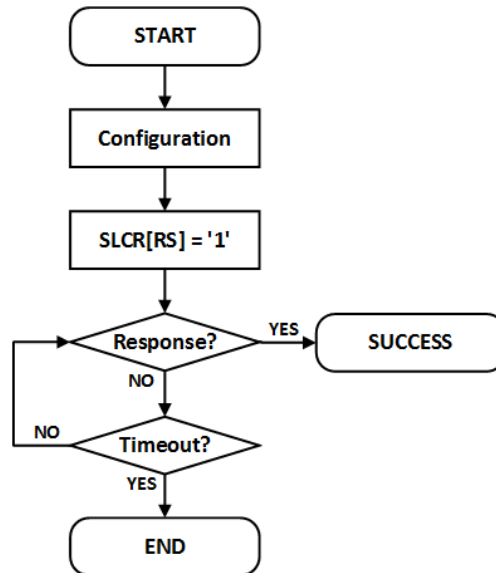


Figure 33 RS Operation Flow

**\*CAUTION:** RA 수신 시 Option Field 의 첫번째 Type 이 Source Link-layer Address, 두번째 Type 이 Prefix Information(0x03) 이 아닐 경우, PLR, PFR, VLTR, PLTR, PAR 값은 보장되지 않는다. 이 경우 IPRAW6 SOCKET 을 이용하여 해당 값들을 수신한다.

#### • Configuration

SLRTR, SLRCR 을 통해 SOCKET-less 의 재전송을 설정하고, SLIMR 의 RS 및 TOUT Interrupt Mask 를 설정한다.

```

{
START :
    /* set SOCKET-less Retransmission Timer, 100ms(0x03E8) (단위, 100us) */
    //SLRTR[0:1] = {0x03, 0xE8};
    /* set SOCKET-less Retransmission Counter, 5 */
    //SLRCR = 0x05;

    /* set Interrupt Mask Bit */
    //SLIMR[RS] = '1'; // RS Interrupt Mask Bit
    //SLIMR[TOUT] = '1'; // TIMEOUT Interrupt Mask Bit
}
  
```

#### • SOCKET-less Command

RS Command 를 통해 RS Packet 을 전송한다.

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

- **Response?**

Router 로부터 RA Packet 을 수신한 경우 SLIR[RS]가 설정된다.

```
{
    /* check RS Interrupt */
    if(SLIR[RS] == '1') /* received RA Packet */
    {
        SLIRCLR[RS] = '1'; /* clear Interrupt */
        goto Configuration;
    }
    else goto Timeout?;
}
```

- **Timeout?**

설정된 재전송 시간 동안 Router 로부터 RA Packet 을 수신하지 못한 경우 SLIR[TOUT]이 발생한다.

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

- **SUCCESS**

RA Packet 의 Prefix Length, Flags, Valid Lifetime, Prefix Lifetime, Prefix Address 는 각각 PLR, PFR, VLTR, PLTR, PAR 에 저장된다.

```
{
    Prefix_length = PLR; /* RA Prefix Length */
    Flags = PFR; /* RA Flags */
    Valid_Lifetime = VLTR; /* RA Valid Life Time */
    Prefix_Lifetime = PLTR; /* RA Prefix Life Time */
    Prefix_address[0:15] = PAR[0:15]; /* RA Prefix Address */
}
```

### 6.6.7 Unsolicited NA(Neighbor Advertisement)

SLCR[NA]는 자신의 새로운 Network 정보를 알리기 위해 Unsolicited NA(Neighbor Advertisement) Packet 을 전송한다. Unsolicited NA 의 Destination 는 All-Node Multicast

Address 인 FF02::1 로 자동 설정되며 Target Address 는 SLPSR 에 따라 LLAR 또는 GUAR 이 선택된다.

Unsolicited NA 는 응답이 없는 형태의 메시지이며 전송을 완료한 경우 SLIR[TOUT]이 발생한다.

아래의 Figure 34 는 Unsolicited NA Operation Flow 를 보여준다.

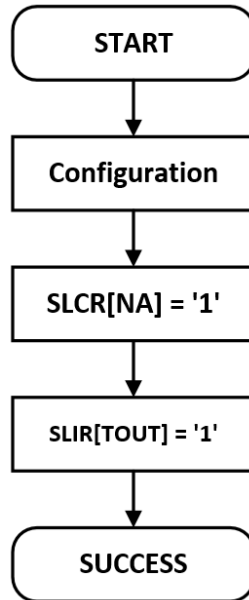


Figure 34 Unsolicited NA Operation Flow

• **Configuration**

LLAR 또는 GUAR 에 바뀐 Address 를 저장하고 SLPSR 을 통해 Target Address 에 들어갈 Address 를 선택한다.

마지막으로 SLIMR 의 TOUT Interrupt Mask 를 설정한다.

```

{
START :
  if (Target Address is Link Local Address)
  {
    LLAR[0:15] = { 0xFE, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
                  0x00, 0x00, 0x00, 0x00, 0x00, 0x12, 0x34, 0x56 };
    SLPSR = 0x10;
  }
  else /* Target Address is Global Unicast Address */
  {
    GUAR[0:15] = { 0x20, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
                  0x00, 0x00, 0x00, 0x00, 0x00, 0x12, 0x34, 0x56 };
    SLPSR = 0x11;
  }

  /* set SOCKET-less TIMEOUT Interrupt Masking bit */
  SLIMR[TOUT] = '1';
}
  
```



• **SOCKET-less Command**

NA Command 를 통해 Unsolicited NA Packet 을 전송한다.

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

• **Timeout**

Unsolicited NA 전송이 완료되면 SLIR[TIMEOUT]이 발생한다.

```
{
  /* check TIMEOUT Interrupt */
  if(SLIR[TOUT] == 1)
  {
    SLIRCLR[TOUT] = '1'; /* clear Interrupt */
    goto SUCCESS;
  }
}
```

## 6.7 Retransmission

### 6.7.1 ARP & PING & ND Retransmission

ARP & PING & ND(Neighbor Discovery) Retransmission 은 Request Packet 전송에 대한 Destination 의 응답이 없을 경우 해당 Packet 에 대한 Retransmission 이 발생한다. 이때 Retransmission 은 Destination 의 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( $ARP_{TO}$ ,  $PING_{TO}$ ,  $ND_{TO}$ )은 다음과 같다.

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

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

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

Ex)  $TIMEOUT_{VAL} = 2000(0x07D0)$ ,  $TIMEOUT_{CNT} = 8(0x0008)$   
 $ARP_{TO} = 2000 \times 0.1ms \times 9 = 1.8s$

$ARP_{TO}$ 은 TCP4 Mode 에서  $Sn\_CR[CONNECT]$ , UDP4 와 IPRAW4 Mode 에서  $Sn\_CR[SEND]$  그리고 SOCKET-less Command 의 SLCR[ARP]에 의한 ARP-Process 에서 Destination 의 응답이 없는 경우 발생 하며, SLIR[TOUT] 이나  $Sn\_IR [TIMEOUT]$ 으로 확인한다.

$PING_{TO}$  은 SOCKET-less Command 의 SLICR[PING] 또는 SLICR[PING6]에 의한 ARP-Process 에서 Destination 의 응답이 없는 경우 발생하거나 ARP-Process 이후 Destination 로부터 PING Reply 가 없는 경우 발생하며 SLIR [TOUT]으로 확인한다.

$ND_{TO}$  는 TCP6 또는 TCPD Mode 에서 Sn\_CR[CONNECT6]에 의한, UDP6, UDPD 또는 IPRAW6 Mode 에서 Sn\_CR[SEND6], 그리고 SOCKET-less Command 의 SLICR[ARP6], SLICR[NS], SLICR[RS] 에 의한 ND 과정에서 발생하며 SLIR [TOUT]이나 Sn\_IR[TIMEOUT]으로 확인한다.

### 6.7.2 TCP Retransmission

TCP Retransmission 은 SYN, FIN, DATA Packet 등을 전송한 후, Destination 으로부터 해당 ACK Packet 을 수신하지 못한 경우 발생한다. SOCKET 은 Destination 으로부터 ACK Packet 을 수신할 때까지 지정된 RTR 시간마다 이전 Packet 을 재전송하고, 지정된 RCR 횟수를 초과하여 전송한 경우 해당 Packet 에 대한 TIMEOUT 을 발생시킨다.

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 :  $TIMEOUT_{VAL} \times 2^{(M+1)} > 65535$  and  $0 \leq M \leq TIMEOUT_{CNT}$  에서의 최소값  
 $TIMEOUT_{VAL} = Sn\_RTR$   
 $TIMEOUT_{CNT} = Sn\_RCR$   
 $TIMEOUT_{MAXVAL} = TIMEOUT_{VAL} \times 2^M$

Ex) RTR = 2000(0x07D0), RCR = 8(0x0008)

$$\begin{aligned} TCP_{TO} &= (0x07D0+0x0FA0+0x1F40+0x3E80+0x7D00+0xFA00+0xFA00+0xFA00+0xFA00) \times 0.1ms \\ &= (2000 + 4000 + 8000 + 16000 + 32000 + ((8 - 4) \times 64000)) \times 0.1ms \\ &= 318000 \times 0.1ms = 31.8s \end{aligned}$$

$TCP_{TO}$  은 Sn\_CR 의 CONNECT, CONNECT6, SEND, SEND6, DISCON Command 에 의해 발생하며, Sn\_IR[TIMEOUT] 으로 확인한다.

## 6.8 Others Functions

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

SYS\_CLK 은 25MHz 나 100MHz 로 동작 할 수 있으며, SYCR1[CLKSEL]이나, PHYCR1[RST], PHYCR1[PWDN] 의 설정에 의해 Switching 된다. Clock Switching 이 발생할 경우 SYS\_CLK 이 안정화 될 때까지 대기한다. (참조 [8.4.1 Reset Timing](#))

SYCR1[CLKSEL]	PHYCR1[RST]	PHYCR1[PWDN]	SYS_CLK(MHz)
0	0	X	25
0	1	0	100 (Default)
0	1	1	25
1	X	X	25

### 6.8.2 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, PHYLOCKR 을 Unlock 한다.

Ex) PHY Operation Mode 설정

```

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

    /* Set PHYCR0 100/10BASE & Full/Duplex */
    phy_mode = '000' // Auto Negotiation
    //phy_mode = '100' // 100BASE-TX FDX
    //phy_mode = '101' // 100BASE-TX HDX
    //phy_mode = '110' // 10BASE-TX FDX
    //phy_mode = '111' // 10BASE-TX HDX
    PHYCR0[2:0] = phy_mode;

    /* PHY Reset Process */
    PHYCR1[RST] = '1';
    Wait TPRST; // refer to 8.4.1 Reset Timing

    /* PHYCR0 & PHYCR1 Lock */
    PHYLOCKR = 0x00; // for Lock, write any value
    /* wait until PHY Link is up */
    while(PHYSR[LNK] != '0');

    /* read PHYSR */
    If( (PHYSR[5:3] == phy_mode) ) SUCCESS;
    else FAIL;
}
    
```

### 6.8.3 Ethernet PHY Parallel Detection

Link Partner 가 Auto-negotiation 을 지원하지 않을 경우, 내장 Ethernet PHY 는 Parallel Detection 을 통해 Link 를 형성한다.

**\*CAUTION** 10F/10H 와 같이 Duplex Mode 가 일치하지 않는 경우 Network 성능 저하의 원인 될 수 있다.

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

### 6.8.4 Ethernet PHY Auto MDIX

Ethernet PHY 를 Auto-negotiation(PHYCR0[MODE2] = '0')로 사용할 경우 Auto-MDIX 를 지원하며, Symmetric Transformer(Figure 43 Transformer Type)를 사용한다.

Auto-negotiation 를 사용하지 않는 경우는 Auto-MDIX 를 지원하지 않으므로 Cross UTP Cable 을 사용한다.

**\*CAUTION** : Link 된 두 Node 간 어느 한 Node 라도 Auto-MDIX 를 지원할 경우, Straight 혹은 Cross UTP Cable 둘 다 사용 가능하다.

### 6.8.5 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] = '0'로 설정할 경우 Ethernet PHY 는 Normal Mode 로 진입하고, SYS\_CLK 은 SYCR1[CLKSEL] 설정에 따라 결정된다. 참조 4.1.5 SYCR1 (System Config Register 1)

```

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 8.4.1 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 8.4.1 Reset Timing

/* wait until Clock is switched 25 to 100MHz*/
Wait TLF; // refer to 8.4.1 Reset Timing
}
```

### 6.8.6 Ethernet PHY's Registers Control

Ethernet PHY의 Register 들은 일반적으로 MDC/MDIO(Management Data Clock / Input Output) Interface 를 통해 Access 된다. W6100 은 MDC/MDIO Controller 를 내장하고 있으며, HOST 는 PHYDIVR, PHYRAR, PHYDOR, PHYDIR, PHYACR 을 통해 Control 할 수 있다.

아래의 Figure 35 는 MDC/MDIO Write Control Flow 를 나타낸다.

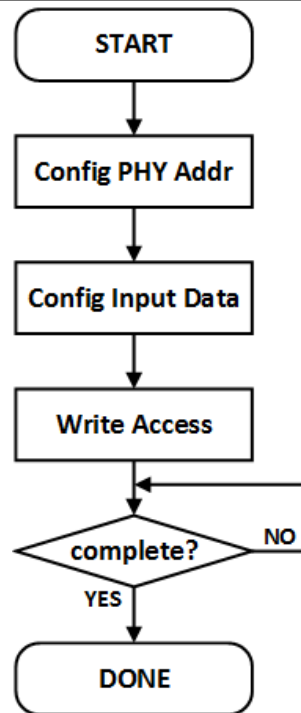


Figure 35 MDC/MDIO Write Control Flow

- **Config PHY Register Address**

PHYRAR 에 Write Access 하려는 PHY Register Address 를 저장한다.

```

{
START :
  /* set PHY Register Address into PHYRAR */
  PHYRAR = 0x00; /* BMCR Address is 0x00 */
}
  
```

- **Config Input Data**

PHYDIR0 & PHYDIR1 를 통해 PHY Register 에 Write 할 16bits Data 를 저장한다.

PHYDIR1 에는 상위 8bits Data 를 저장하고 PHYDIR0 에 하위 8bits Data 를 저장한다.

```

{
  Data = 0x8000; /* set RST bit in BMCR */

  PHYDIR1 = (Data & 0xFF00) >> 8; /* set upper 8bits Data */
  PHYDIR0 = Data & 0x00FF; /* set lower 8bits Data */
}
  
```

- **Write Access / Complete?**

PHYACR 을 '0x01'로 설정하면, PHYRAR 에 지정된 PHY Register 를 PHYDIR 을 통해 Write Access 한다. 완료 후 자동 Clear 된다.

```

{
  PHYACR = 0x01; /* set Write Access */
  while(PHYACR != 0); /* wait until MDC/MDIO Control is complete */
}
  
```

아래의 Figure 36 은 MDC/MDIO Read Control Flow 를 나타낸다.

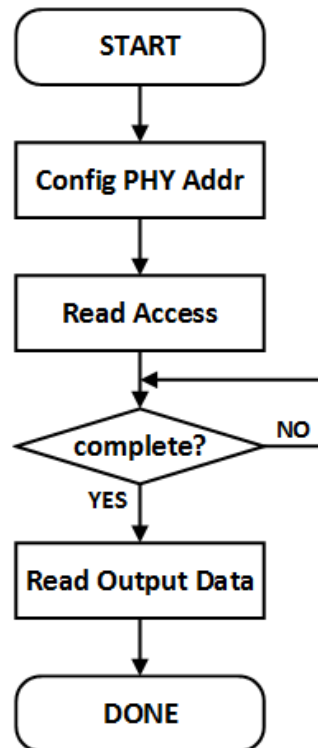


Figure 36 MDC/MDIO Read Control Flow

- **Config PHY Register Address**

PHYRAR 에 Write Access 하려는 PHY Register Address 를 저장한다.

```

{
START :
  /* set PHY Register Address into PHYRAR */
  PHYRAR = 0x01; /* BMSR Address is 0x01 */
}
  
```

- **Read Access / Complete?**

PHYACR 을 '0x02'로 설정하면, PHYRAR 에 지정된 PHY Register 를 PHYDOR 을 통해 Read Access 한다. 완료 후 자동 Clear 된다.

```

{
  PHYACR = 0x02; /* set Read Access */
  while(PHYACR != 0); /* wait until MDC/MDIO Control is complete */
}
  
```

- **Read Output Data**

Read Access 가 완료되면 PHY Register 값은 PHYDOR0,1 에 저장된다.

PHYDOR1 은 상위 8bits Data 가 저장되고, PHYDOR0 에는 하위 8bits Data 가 저장된다.

```

{
  Data = (PHYDOR1 & 0x00FF) << 8; /* get upper 8bits Data */
}
  
```

```
Data = Data + (PHYDOR0 & 0x00FF); /* get lower 8bits Data */  
}
```

### 6.8.7 Ethernet PHY 10BASE-T<sub>e</sub> Mode

W6100 Ethernet PHY 는 10BASE-T<sub>e</sub> Mode 로 동작할 수 있으며 설정방법은 아래와 같다.

```
{  
    /* PHYCR0&PHYCR1 Unlock */  
    PHYLCKR = 0x53;  
  
    /* Enable Auto-negotiation */  
    PHYCR0[MODE2:MODE0] = '000';  
  
    /* set PHY Te Mode */  
    PHYCR1[TE] = '1';  
  
    /* PHY Reset Process */  
    PHYCR1[RST] = '1';  
    Wait TPRST; // refer to 8.4.1 Reset Timing  
}
```



## 7 Clock & Transformer Requirements

### 7.1 Quartz Crystal Requirements.

Table 10 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	W6100 Reset			60	ms
Trans-conductance( $g_m$ )			8.43		mA/V
Gain Margin ( $gain_{margin}$ )	$gain_{margin} = g_m / g_{m_{crit}}$	6.99			dB

$C_0^{(1)}$  : The Packaging Parasitic Shunt Capacitance.

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

$C_{L1}, C_{L2}$  : 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)  $g_{m_{crit}} = 4 \times (ESR + R_{Ext}) \times (2\pi F)^2 \times (C_0 + C_L)^2$

$ESR^{(1)}$  : Maximal equivalent series resistance. eq)  $ESR = R_m \times (1 + C_0/C_L)^2$

$R_{Ext}$  : 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.

- $C_0, C_L, ESR$  and  $DL$  are provided by the crystal manufacturer.
- The W6100 has no feedback resistor. Therefore, it must be inserted outside.

\* Crystal 회로는 아래와 같은 형식으로 모델링 된다.

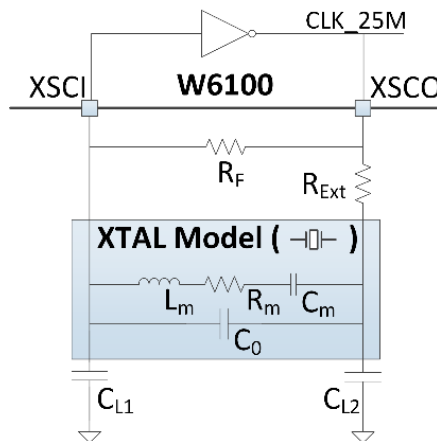


Figure 37 Quartz Crystal Model

Table 11 Crystal Recommendation Characteristics

Parameter	Range
Frequency	25 MHz
Frequency Tolerance (at 25°C)	±30 ppm

Shunt Capacitance	7pF Max
Drive Level	500uW
Load Capacitance	8pF
Aging (at 25°C)	±3ppm / year Max

## 7.2 Oscillator requirements.

Table 12 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 Voltage		1.08V	1.2V	1.32V	
Aging (at 25°C)		±3 / year Max			ppm

## 7.3 Transformer Characteristics

Table 13 Transformer Characteristics

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

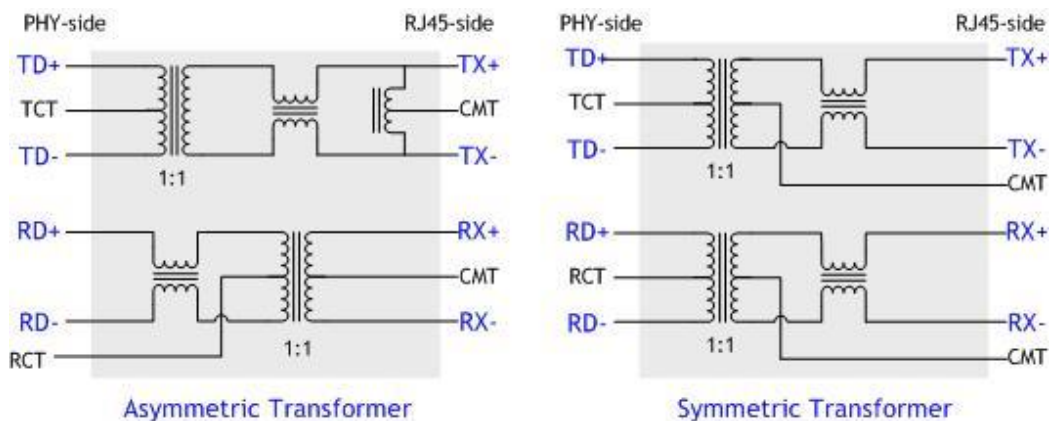


Figure 38 Transformer Type

## 8 Electrical Specification

### 8.1 Absolute Maximum ratings

Table 14 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.*

### 8.2 Absolute Maximum ratings (Electrical Sensitivity)

Table 15 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 16 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

## 8.3 DC Characteristics

Table 17 DC Characteristics

(Test Condition:  $T_a = -40$  to  $85^\circ\text{C}$ )

Symbol	Parameter	Test Condition	Min	Typ	Max	Unit
$V_{DD}$	Supply voltage	Apply VDD, AVDD	2.97	3.3	3.63	V
$V_{IH}$	High level input voltage		2.0	-	-	V
$V_{IL}$	Low level input voltage		-		0.8	V
$V_{T+}$	Schmitt trig Low to High Threshold point	All inputs except Analog PINs	0.8	1.1	-	V
$V_{T-}$	Schmitt trig High to Low Threshold point	All inputs except Analog PINs	-	1.6	2.0	V
$T_J$	Junction temperature		-40	25	125	$^\circ\text{C}$
$I_L$	Input Leakage Current			$\pm 1$	$\pm 10$	$\mu\text{A}$
$R_{PU}$	Pull-up Resistor		40	75	190	Kohm
$R_{PD}$	Pull-down Resistor		40	75	190	Kohm
$V_{OL}$	Low level output voltage	IOL = 4.0mA ~ 16mA All outputs except XO			0.4	V
$V_{OH}$	High level output voltage	IOH = 4.0m ~ 6mA, All outputs except XO	2.4			V
$I_{DD1}$	Supply Current (Normal operation mode)	VDD=3.3V, AVDD=3.3V, $T_a = 25^\circ\text{C}$		132		mA
$I_{DD2}$	Supply Current (Power Down mode)	PHY Power Down mode, VDD=3.3V, AVDD=3.3V, $T_a = 25^\circ\text{C}$		13		mA

## 8.4 AC Characteristics

### 8.4.1 Reset Timing

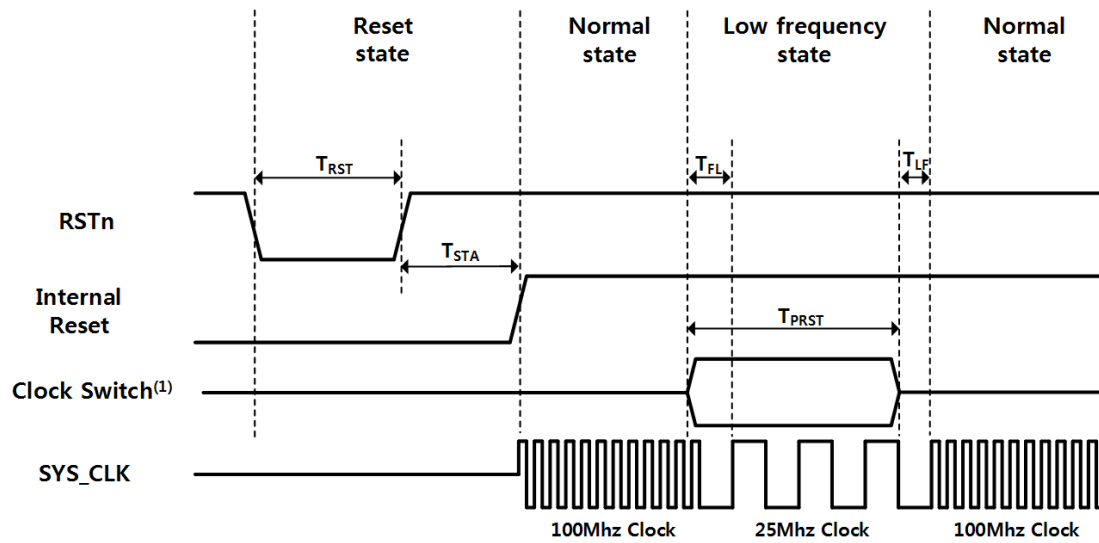


Figure 39 Reset Timing

Table 18 Reset Table

Symbol	Description	Min	Typ	Max
$T_{RST}$	Reset Time	350 ns	580 ns	1.0 $\mu$ s
$T_{STA}$	Stable Time	-		60.3 ms
$T_{FL}$	Fast to Low Time by SYCR1[CLKSEL]	100 ns		-
	Fast to Low Time by PHYCR1[RST] or PHYCR1[PWDN]	300 ns		
$T_{PRST}$	PHY Auto Reset Time	0.6 ms		-
	PHY Power Down Time	200 $\mu$ s		
	Clock Switch Time	200 ns		
$T_{LF}$	Low to Fast Time by SYCR1[CLKSEL]	100 ns		-
	Low to Fast Time by PHYCR1[RST] 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  $T_{FL}$ , 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

## 8.4.2 BUS ACCESS TIMING

### 8.4.2.1 READ TIMING

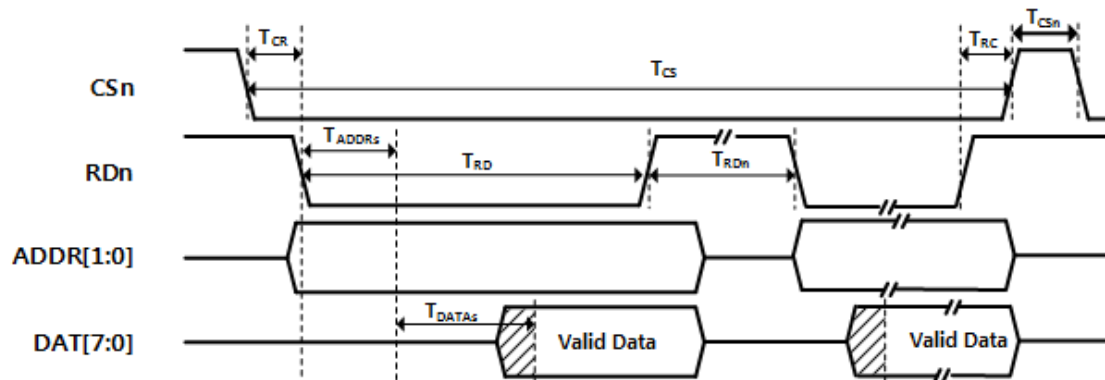


Figure 40 Bus Read Timing

Table 19 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	

### 8.4.2.2 WRITE TIMING

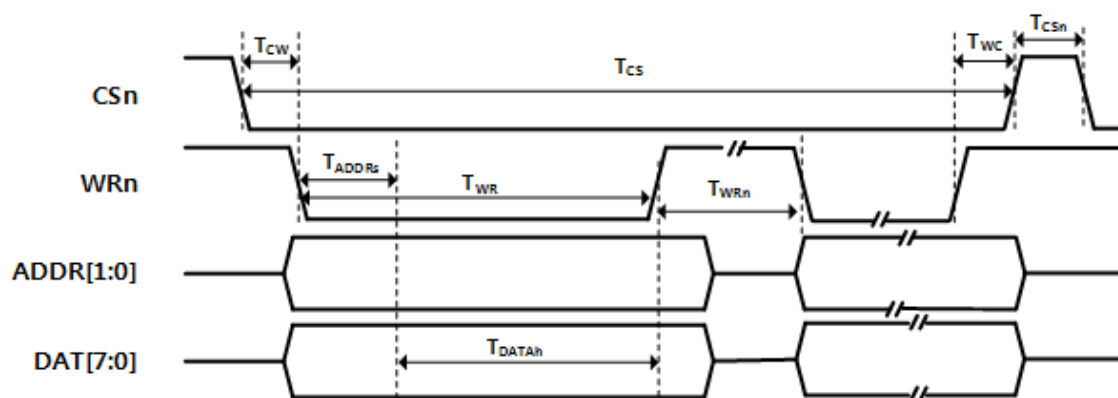


Figure 41 BUS Write Timing

Table 20 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	

### 8.4.3 SPI ACCESS TIMING

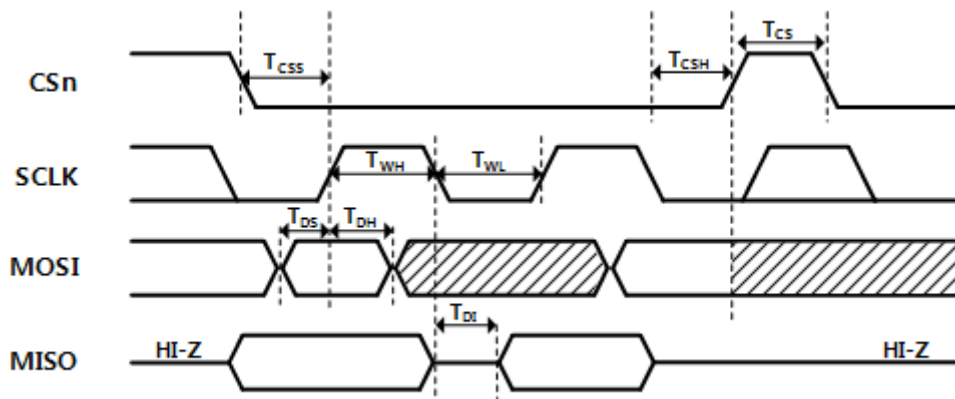


Figure 42 SPI Access Timing

Table 21 SPI Access Timing

Symbol	Description	Min	Max	Units
$F_{SCLK}$	SCLK Clock Frequency		70	MHz
$T_{CSS}$	CSn Setup Time	3 SYS_CLK		ns
$T_{CSH}$	CSn Hold Time	2 SYS_CLK		ns
$T_{CS}$	CSn High Time	2 SYS_CLK		ns
$T_{WH}$	SCLK High time	6		ns
$T_{WL}$	SCLK Low Time	6		ns
$T_{DS}$	Data Setup Time	3		ns
$T_{DH}$	Data In Hold Time	3		ns
$T_{DI}$	Data Invalid Time	7		ns

### 8.4.4 Transformer Characteristics

Table 22 Transformer Characteristics

Parameter	Transmit End	Receive End
-----------	--------------	-------------

Turn Ratio	1:1	1:1
Inductance	350 uH	350 uH

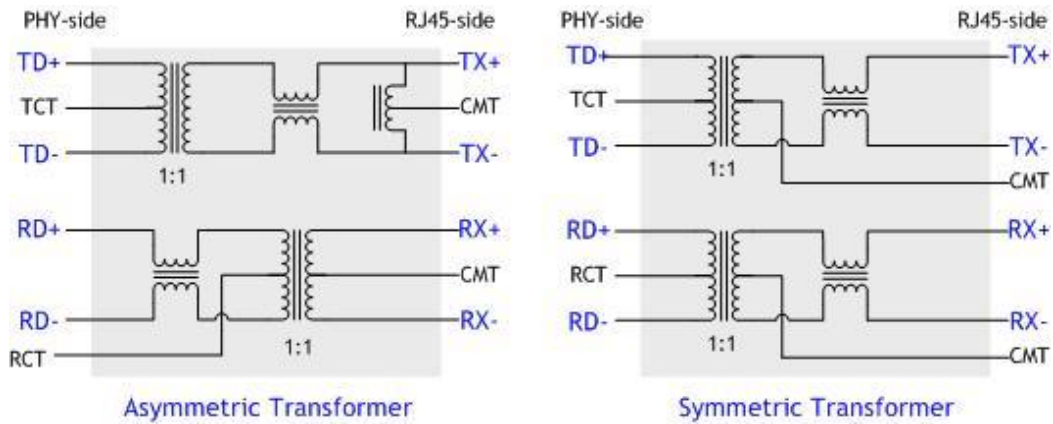


Figure 43 Transformer Type

### 8.4.5 MDIX

W6100 는 Auto-Negotiation Mode 인 경우, Auto-MDIX 를 지원한다.

## 8.5 POWER DISSIPATION

Table 23 Power Dissipation

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

Condition	Min	Typ	Max	Unit
100M Link	-	98	115	mA
10M Link	-	112	265	mA
10M-Te Link	-	75	190	mA
100M Unlink(실측)	-	50	199	mA
10M Unlink(실측)	-	26	170	mA
10M-Te Unlink(실측)	-	26	130	mA
Un-Link (Auto-negotiation mode)(실측)	-	50	199	mA
Power Down mode	-	14	20	mA



## 9 Package Information

### 9.1 LQFP48

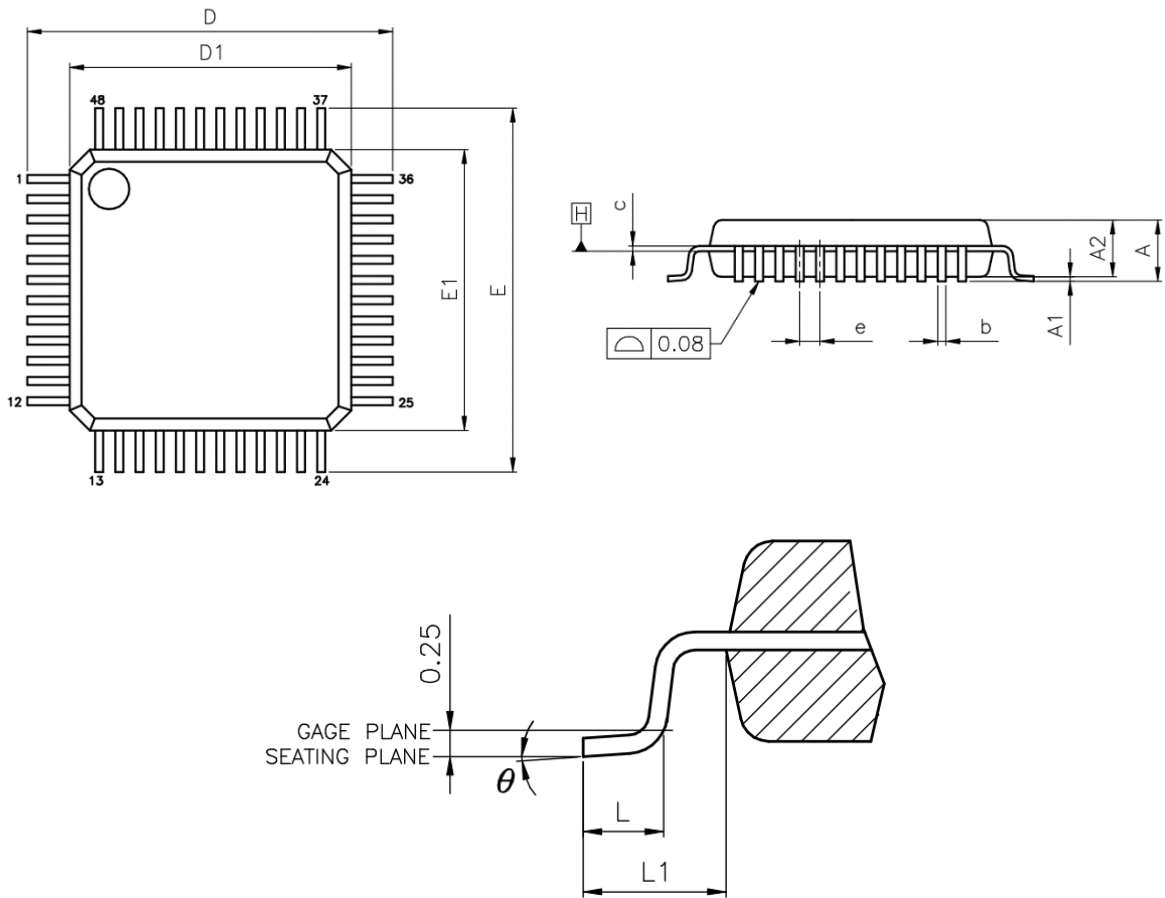


Table 24 LQFP48 VARIATIONS (ALL DEMINSIONS SHOWN IN MM)

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.

## 9.2 QFN48

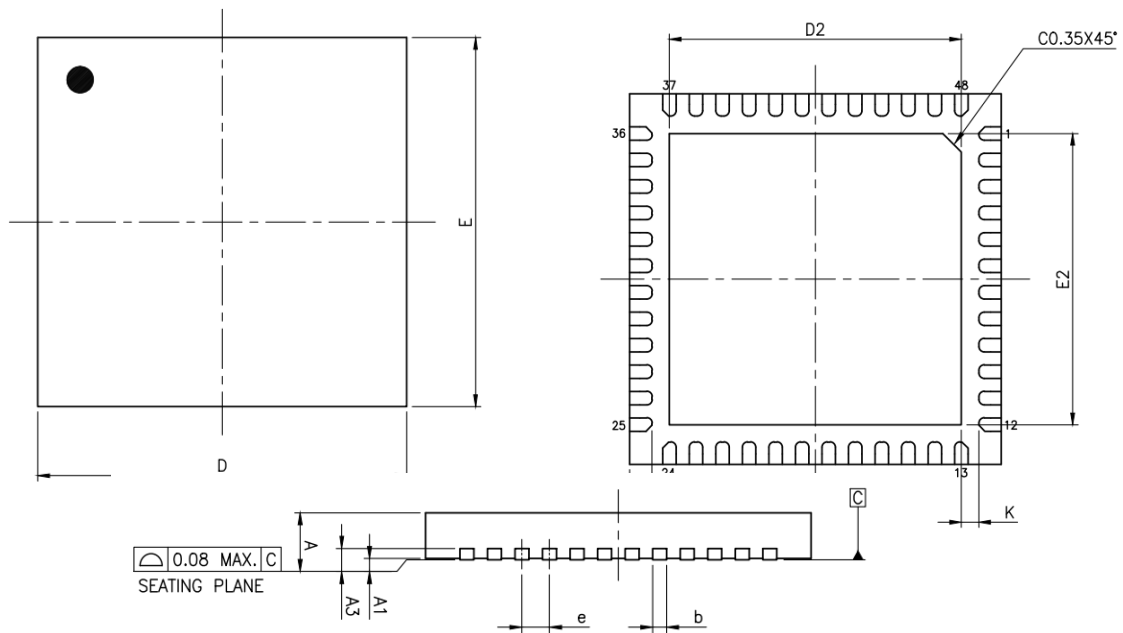


Table 25 QFN48 VARIATIONS (ALL DEMINSIONS SHOWN IN MM)

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. DIMENSION 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.

---

## 10 Document Revision History

Version	Date	Descriptions
Ver. 1.0.0	1FEB2019	Initial Release
Ver. 1.0.1	7MAR2019	1.Modified Power Dissipation (in <a href="#">8.5 POWER DISSIPATION</a> )
Ver. 1.0.2	15MAY2019	1. Added Maximum junction temperature (in <a href="#">8.1 Absolute Maximum ratings</a> )
Ver. 1.0.3	8OCT2019	Modified Hyperlink about “Clock Selection Guide” in page 13
Ver. 1.0.4	28OCT2019	Modified MACRAW Mode Value from ‘0100’ to ‘0111’

## Copyright Notice

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

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

Wiki : <https://wizwiki.net>

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

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