SMIC

Connect your devices to TCP/IP network

NetEyeBox(D) V.2.0

(Ethernet to DI/PI/DO+RS485)

DI/PI/DO + RS485 Product

Part Number : 3301-Ethernet



SM Information and Communication

Homepage : http://www.smic21.com
Support : +82-2-6292-2100 khg@smic.co.kr, smic21@smic21.com
Address :A-705 Sktwintech Bld. Gassandong, GeumcheonGu, Seoul Korea Zip 153-023

SMIC

Connect your devices to TCP/IP network

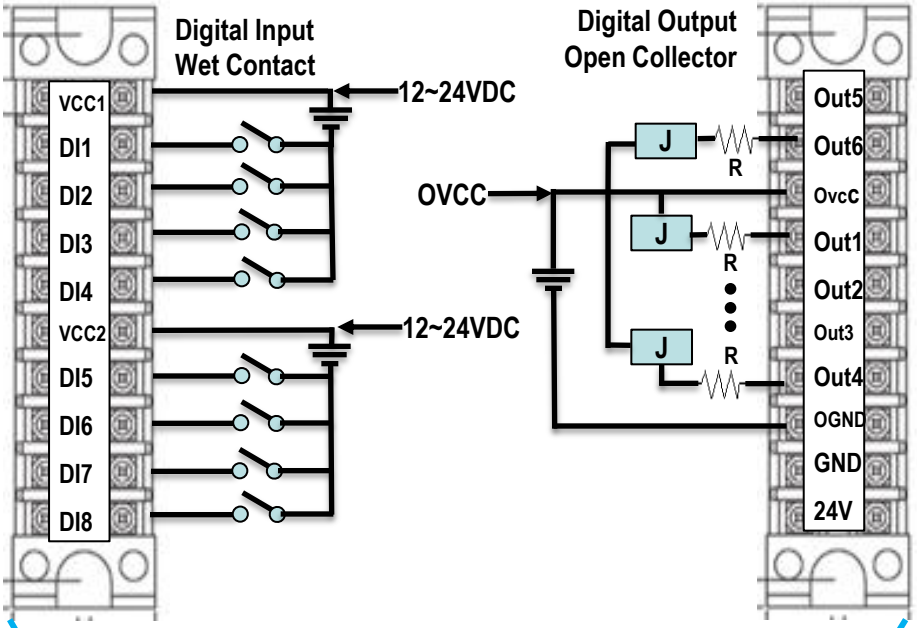
Device Feature

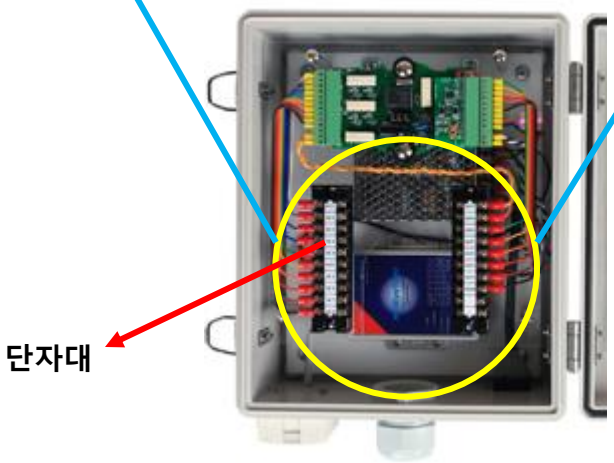
I/O channel number-----8DI(PI)6DO+1 Serial(RS485), OptionRS232
Input range----------+12V~+36V
DI/PI Input signal----------Wet Contact, Falling Edge
DOI Output signal----------Open Collector
Counter frequency----------100Hz@8channels;1000Hz@1channel
Counter length----------32-bit
Output BUS---------- WiFi 802.11g
Output Protection----------Lightening, static
Power input----------9~12V(AC/DC)
Power consumption----------<0.6W
Ambient temperature:
Operation---------- -20~70℃(-13~185°F)

Application:

- Remote data acquisition
- Process monitoring
- Industrial process control
- Energy management
- Supervisory control
- Security systems
- Laboratory automation
- Building automation
- Product testing
- Direct digital control

8DI(Counter)/6DO Connection



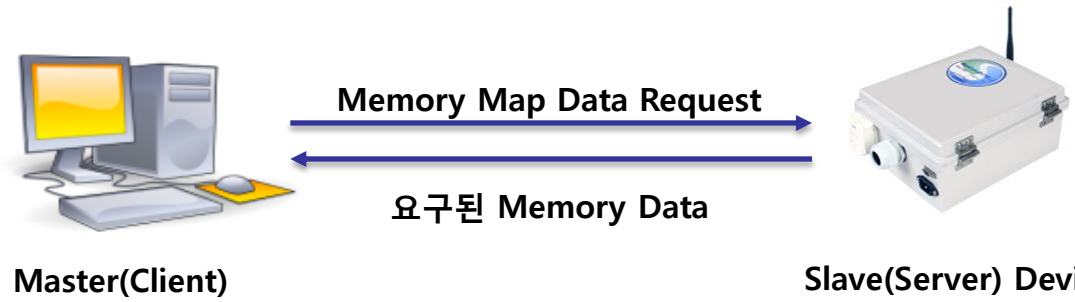


6. NetEyeBox(D) Data Communication

NetEyeBox(D) 의 TCP/IP 접속이 성공하면 Modbus Protocol 를 통하여 Modbus Register Map의 내용을 읽거나 쓸 수 있다.

● Modbus Protocol

가장 많이 사용되고 잘 정립된 Protocol 이다. 주로 다양한 PLC와 통신모듈 DI/DO/AI등 여러 장비에 신뢰성 있는 통신 방법을 제공한다. 내용은 아주 간단하다 약속된 프로토콜과 Memory Map이 제공된다.. Hardware는 Master와 Slave로 구분된다.(TCP/IP 용어로는 Client와 Server라고 한다). Data를 처음 보내는 측이 Master이고 Client이다. Hardware내부의 저장 가능한 Memory에 Memory Map이 저장되고 있고 약속된 Data을 보내면 그에 따라 Memory Map에서 대응되는 데이터를 송신하는 구조로 되어 있다.



Modbus Protocol의 Data Packet는 모두 Hexa Value를 사용하며 다양한 명령어 Type을 가지고 있지만 주로 READ와 WRITE만 사용된다. READ는 Type 0x03이고 WRITE는 Type 0x06이다. Master 는 Slave Device의 ID와 명령어의 Type(READ or WRITE), Memory Map의 address와 가져올 Data Packet (Register 수로 표현됨)의 수 그리고 Error Detection을 위한 2bytes의 CRC를 보낸다.이에 대하여 Slave는 같은 형태로 요청된 Data Packet으로 응답 한다. Slave가 여러 개 붙어도 TCP/IP의 Socket한 라인을 통하여 모두와 통신 할 수 있으므로 확장 성이 우수하다 수백 channel의 통제도 가능하다.

- 전형적인 Modbus Protocol의 Packet 구조.

BYTES	설명 (모두 HEXA Value로 설정합니다.)
Bytes1	RS485의 Device ID
Bytes2	READ or WRITE 기능
Bytes3	읽거나 쓰게 될 Memory Map의 시작 주소 Address
Bytes4	시작 주소 Address 의 High byte
Bytes5	시작 주소 Address의 Low byte
Bytes6	읽거나 쓰게 될 Register(1word)의 수
Bytes7	CRC high bytes
Bytes8	CRC low bytes

NetEye3XXX Series Default Parameters

Address of device: 254 개 ID

Baudrate: 19200 Default

Channel: enable all channels

Filtering: 200us~300mls for frequency input

CRC Error Correcting Details

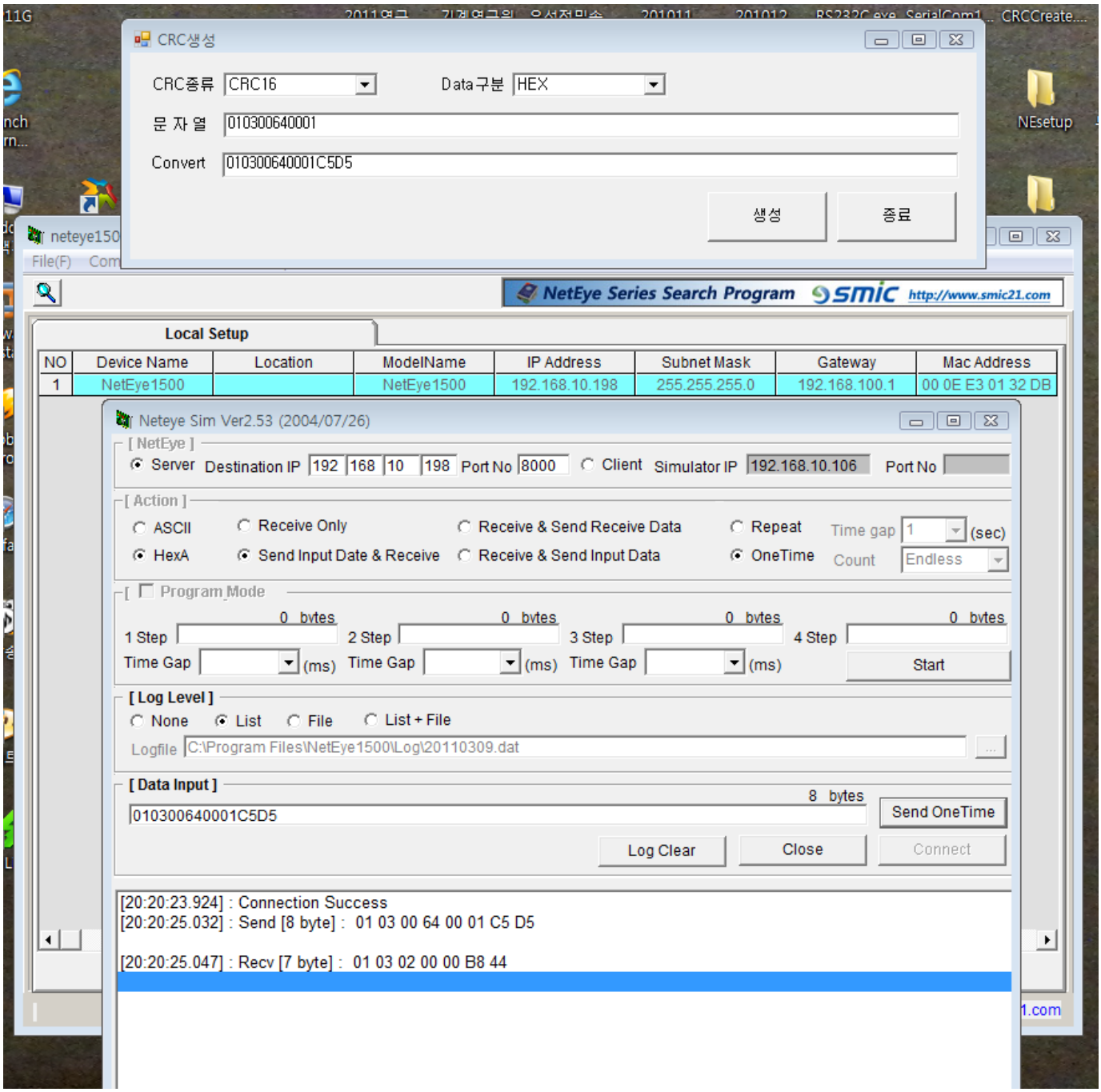
❖ **Check field calculation Sample Code in C language)**

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <conio.h>
unsigned short CRC16(unsigned char *, int);
void main()
{
unsigned char SendData[10], *pMsg;
unsigned char Len;
unsigned short crc16=0xffff;
pMsg=&SendData[0];
Len=0x06;
memset(&SendData[0], 0x01, 1);
memset(&SendData[1], 0x03, 1);
memset(&SendData[2], 0x00, 1);
memset(&SendData[3], 0x64, 1);
memset(&SendData[4], 0x00, 1);
memset(&SendData[5], 0x01, 1);
crc16= CRC16(pMsg, Len);
printf("CRC16 is %x\n", crc16);
system("pause");
}
unsigned short CRC16(unsigned char*addr, int num) {
unsigned short CRC = 0xFFFF;
unsigned short CRCT= 0x0000;
int i;
while (num--) {
CRC ^= *addr++;
for (i = 0; i < 8; i++) {
  if (CRC & 1)
  {
    CRC >>= 1;
    CRC ^= 0xA001;
  }
  else  {  CRC >>= 1;  }
}
}
CRCT=CRC;
CRCT=CRCT<<8;
CRC=CRC >> 8;
CRC=CRC | CRCT;
return CRC;
}
```

Setting 순서

1. NetEye 1500 Setup Program을 실행한다.
2. NetEye1500 Manual을 www.smic21.com에서 Down 받아 Setting 한다.
3. I/O Device Parameter 변경 시 NeEye1500 Serial Port를 RS485로, Baudrate는 19200 bps, N, 8, 1로 Setting하여 통신되도록 한 후 I/O Device의 Parameter (Device ID, Baudrate 등을 변경 할 수 있다.
4. Program으로 통신 테스트 하기 전에 장비를 테스트하기 위하여 www.smic21.com 에서CRCCreate 프로그램을 down 받고 NetEye1500 Setup Program의 Tool Menu 에서 Simulator 를 이용하여 I/O Device 통신을 테스트한다.

***다음은 NetEyeS3302(16DI/Counter)를 기준으로 한 테스트이며 모든 NetEyeBox 시리즈에 동일하게 적용된다.**



4. CRCcreat.exe Program: CRC16 check sum data를 2 bytes 생성 시킨다.
5. NetEye Simulator: 모든 NetEye Setup Program의 Simulator를 통하여 NetEyeS3302와 통신하기 위하여 Modbus Protocol에 의한 Request Data를 보낸다.

6. 아래는 초기에 Device ID를 확인하는 방법이다.

Device ID 확인 방법

1) PC-> NE3302

ADDR	내용	Value(HEX)	비고
0	고정	ff	Broadcast ID
1	Read/Write	03	Read Mode
2	ADDR-H	00	NES3302의 ADDR Register
3	ADDR-L	06	
4	Data Count -H	00	Reading Register 의 갯수
5	Data Count -L	01	
6	CRC-1	71	계산식
7	CRC-2	D5	계산식

2) PC <- NE3302 응답

ADDR	내용	Value(HEX)	비고
0	고정	ff	
1	Read/Write	03	Read Mode
2	Count	02	Data Count (ADDR9~10)
3	고정	00	
4	Device Address	초기치 : FE(254)	NES3302의 설정된 ADDR
5	CRC-1	0	계산식
6	CRC-2	0	계산식

현재는 FE로 설정되어 있음

Device ID 변경 방법

3) PC-> NES3302

ADDR	내용	Value(HEX)	비고
0	ID ADDR	FE	현재 NES4300S의 ID ADDR
1	Read/Write	06	Write Mode
2	ADDR-H	00	NE3302의 ADDR 의 Register
3	ADDR-L	06	
4	변경 ADDR -H	00	
5	변경 ADDR -L	01~FE	NES3302의 변경할 ID ADDR
6	CRC-1	BC	계산식
7	CRC-2	04	계산식

현재는 01~FE 까지로 설정 할 수 있다.

```
[21:14:04.077] : Send [8 byte] :  01 03 00 64 00 01 C5 D5
[21:14:04.092] : Recv [7 byte] :  01 03 02 FF FF B9 F4
```

7. 상기의 통신내용은 Device ID 01의 NetEyeS3302 device에서 Hexa 0x64 즉 100번째 register 에서 Register 1개를 읽는 내용이다.
8. Device 01로 부터 Register 1개 2bytes FF FF를 읽은 결과 이다. 100 번지 register는 Digital Input의 상태를 나타낸다.
9. 100 번지 register의 2 bytes는 16 channel의 Digital Input 상태를 나타낸다. In-active 일 때 1, Active 즉 입력전압이 인가 되었을 때 0 이 bit 별도 할당된다.

```
[21:19:34.641] : Send [8 byte] :  01 03 00 64 00 01 C5 D5
[21:19:34.656] : Recv [7 byte] :  01 03 02 FF FE 78 34
```

10. 상기의 내용은 읽혀진 2 bytes가 FF FE 는16 Digital Input Channel중 첫 번째 채널 이 0으로 set 되어진 것을 의미 한다.

```
[21:29:58.906] : Send [8 byte] :  01 03 00 64 00 03 44 14
[21:29:58.922] : Recv [11 byte] :  01 03 06 FF FE 00 00 00 10 1D 62
```

11. 상기의 내용은 읽혀진 2 bytes가 FF FF 는 16 Digital Input Channel중 첫 번째 채널 에 Count 되어진 숫자가 Hexa 10 즉 16개 누적 된 것을 의한다.
12. 다른 채널도 register값 address와 0x64, 0x65등으로 바꾸어서 다른 채널의 상태를 읽어 볼 수 있다.

13. Counter Reset

- 100번째 register 00 64 에서 3 개의 address를 읽는다.
- DI 16 ch 이 FF FE 이고 첫 번째 ch의 Counter값이 Hexa 00 10 즉 16이다.

[21:29:58.906] : Send [8 byte] : 01 03 00 64 00 03 44 14

[21:29:58.922] : Recv [11 byte] : 01 03 06 FF FE 00 00 00 10 1D 62

- 102번째 register 00 66 을 0 으로 reset 한다.
- 0 으로 reset 한 Ack를 받는다.

[21:43:57.919] : Send [8 byte] : 01 06 00 66 00 00 69 D5

[21:43:57.981] : Recv [8 byte] : 01 06 00 66 00 00 69 D5

- 100번째 register 00 64 에서 부터 4개의 address를 읽어본다.
- DI 16 channel상태와 첫 번째 ch의 Counter값이 0으로 reset된 것을 볼 수 있다.

[21:46:39.191] : Send [8 byte] : 01 03 00 64 00 04 05 D6

[21:46:39.207] : Recv [13 byte] : 01 03 08 FF FE 00 00 00 00 00 00 C5 1C

3301-Ethernet(8DI/Counter/6DO) Register Map.

*** 모든 값들은 Program시에 HEXA Value로 바꾸어 사용합니다.**

<table>
<tr><th rowspan="2">Address</th><th rowspan="2">Bytes</th><th colspan="2">Value range</th><th rowspan="2">Description</th><th rowspan="2">Property</th></tr>
<tr><th>Min</th><th>Max</th></tr>
<tr><td>0-3</td><td>4</td><td>1</td><td>4294967295</td><td>Serial number,unique for each product</td><td>R</td></tr>
<tr><td>4-5</td><td>2</td><td>100</td><td>65535</td><td>Firmware version number</td><td>R</td></tr>
<tr><td>6</td><td>1</td><td>1</td><td>254</td><td>Device address</td><td>R/W</td></tr>
<tr><td>7</td><td>2</td><td>3301</td><td>3301</td><td>Product model</td><td>R</td></tr>
<tr><td>8</td><td>1</td><td>1</td><td>255</td><td>Hardware version</td><td>R</td></tr>
<tr><td>9</td><td>2</td><td>12</td><td>1152</td><td>Baudrate setting
<table><tr><th>Value</th><th>Buadrate</th></tr><tr><td>12</td><td>1200</td></tr><tr><td>24</td><td>2400</td></tr><tr><td>48</td><td>4800</td></tr><tr><td>96</td><td>9600</td></tr><tr><td>192*</td><td>19200</td></tr><tr><td>384</td><td>38400</td></tr><tr><td>576</td><td>57600</td></tr><tr><td>1152</td><td>115200</td></tr></table>For example:write 96 to register 9 to set the baudrate 9600.</td><td>R</td></tr>
<tr><td>10-99</td><td>-</td><td>-</td><td>-</td><td>Reserved</td><td>-</td></tr>
<tr><td>100</td><td>1</td><td>0</td><td>255</td><td>Status for digital input channel 1 through 8, 0 = contact active,1 = contact inactive.Bit0 correspond to channel1,bit1 correspond to channel 2 and so on.</td><td>R</td></tr>
<tr><td>101</td><td>1</td><td>0</td><td>255</td><td>Open-collector output,0 = active,1 = inactive.Bit0 correspond to output 1,bit1 correspond to channel 2 etc.</td><td>R/W</td></tr>
<tr><td>102</td><td>2</td><td>0</td><td>65535</td><td>High word for counter input 1</td><td>R/W</td></tr>
<tr><td>103</td><td>2</td><td>0</td><td>65535</td><td>Low word for counter input 1,value of counter = （102）*65536 + （103）</td><td>R/W</td></tr>
<tr><td>104</td><td>2</td><td>0</td><td>65535</td><td>High word for counter input 2</td><td>R/W</td></tr>
<tr><td>105</td><td>2</td><td>0</td><td>65535</td><td>Low word for counter input 2,value of counter = （104）*65536 + （105）</td><td>R/W</td></tr>
</table>

Continue…

SMIC

3301-Ethernet Register Map.

*** 모든 값들은 Program시에 HEXA Value로 바꾸어 사용합니다.**

Address	Bytes	Value range		Description	Property
		Min	Max		
106	2	0	65535	High word for counter input 3	R/W
107	2	0	65535	Low word for counter input 3,value of counter = （106）*65536 + （107）	R/W
108	2	0	65535	High word for counter input 4	R/W
109	2	0	65535	Low word for counter input 4,value of counter = （108）*65536 + （109）	R/W
110	2	0	65535	High word for counter input 5	R/W
111	2	0	65535	Low word for counter input 5,value of counter = （110）*65536 + （111）	R/W
112	2	0	65535	High word for counter input 6	R/W
113	2	0	65535	Low word for counter input 6,value of counter = （112）*65536 + （113）	R/W
114	2	0	65535	High word for counter input 7	R/W
115	2	0	65535	Low word for counter input 7,value of counter = （114）*65536 + （115）	R/W
116	2	0	65535	High word for counter input 8	R/W
117	2	0	65535	Low word for counter input 8,value of counter = （116）*65536 + （117）	R/W
118	1	1	100	Respond delay for serial communication, the units is ms and default is 10ms	R/W
119	1	1	255	Filter time for counter input, the units is 10us and the default is 200us	R/W
120	1	0	255	Disable/enable input,0 = disable and 1 = enable.Bit0 correspond to input1, Bit1 correspond to input 2 and so on.	R/W

3302-Ethernet (16DI/Counter) Register Map.

*** 모든 값들은 Program시에 HEXA Value로 바꾸어 사용합니다.**

Modbus register list: **Note: * means default value**

Address	Bytes	Value range		Description	Property
		Min	Max		
0-3	4	1	4294967295	Serial number,unique for each product	R
4-5	2	100	65535	Firmware version number	R
6	1	1	254	Device address	R/W
7	2	3302	3302	Product model	R
8	1	1	255	Hardware version	R
9	2	12	1152	Baudrate setting Value / Buadrate: 12/1200; 24/2400; 48/4800; 96/9600; 192*/19200; 384/38400; 576/57600; 1152/115200 For example:write 96 to register 9 to set the baudrate 9600.	R
10-99	-	-	-	Reserved	-
100	2	0	65535	Status for digital input channel 1 through 16, 0 = contact active,1 = contact inactive.Bit0 correspond to channel 1,bit1 correspond to channel 2 etc.	R
101	2	0	65535	High word for counter input 1	R/W
102	2	0	65535	Low word for counter input 1,value of counter = (101) *65536 + (102)	R/W
103	2	0	65535	High word for counter input 2	R/W
104	2	0	65535	Low word for counter input 2,value of counter = (103) *65536 + (104)	R/W

Continue…

SMIC

3302-Ethernet Register Map.

Address	Bytes	Value range		Description	Property
		Min	Max		
105	2	0	65535	High word for counter input 3	R/W
106	2	0	65535	Low word for counter input 3,value of counter = (105) *65536 + (106)	R/W
107	2	0	65535	High word for counter input 4	R/W
108	2	0	65535	Low word for counter input 4,value of counter = (107) *65536 + (108)	R/W
109	2	0	65535	High word for counter input 5	R/W
110	2	0	65535	Low word for counter input 5,value of counter = (109) *65536 + (110)	R/W
111	2	0	65535	High word for counter input 6	R/W
112	2	0	65535	Low word for counter input 6,value of counter = (111) *65536 + (112)	R/W
113	2	0	65535	High word for counter input 7	R/W
114	2	0	65535	Low word for counter input 7,value of counter = (113) *65536 + (114)	R/W
115	2	0	65535	High word for counter input 8	R/W
116	2	0	65535	Low word for counter input 8,value of counter = (115) *65536 + (116)	R/W
117	2	0	65535	High word for counter input 9	R/W
118	2	0	65535	Low word for counter input 9,value of counter = (117) *65536 + (118)	R/W
119	2	0	65535	High word for counter input10	R/W
120	2	0	65535	Low word for counter input 10,value of counter = (119) *65536 + (120)	R/W
121	2	0	65535	High word for counter input 11	R/W
122	2	0	65535	Low word for counter input 11,value of counter = (121) *65536 + (122)	R/W
123	2	0	65535	High word for counter input 12	R/W
124	2	0	65535	Low word for counter input 12,value of counter = (123) *65536 + (124)	R/W
125	2	0	65535	High word for counter input 13	R/W
126	2	0	65535	Low word for counter input 13,value of counter = (125) *65536 + (126)	R/W
127	2	0	65535	High word for counter input 14	R/W

3302-Ethernet Register Map.

128	2	0	65535	Low word for counter input 14,value of counter = (127) *65536 + (128)	R/W
129	2	0	65535	High word for counter input 15	R/W
130	2	0	65535	Low word for counter input 15,value of counter = (129) *65536 + (130)	R/W
131	2	0	65535	High word for counter input 16	R/W
132	2	0	65535	Low word for counter input 16,value of counter = (131) *65536 + (132)	R/W
133	1	1	100	Respond delay for serial communication, the units is ms and default is 10ms	R/W
134	1	1	255	Filter time for counter input, the units is 10us and the default is 200us	R/W
135	1	0	255	Disable/enable input,0 = disable and 1 = enable.Bit0 correspond to input1, Bit1 correspond to input 2 and so on.	R/W
136	1	0	255	Disable/enable input,0 = disable and 1 = enable.Bit0 correspond to input9, Bit1 correspond to input 10 and so on.	R/W