MCS ELECTRONICS

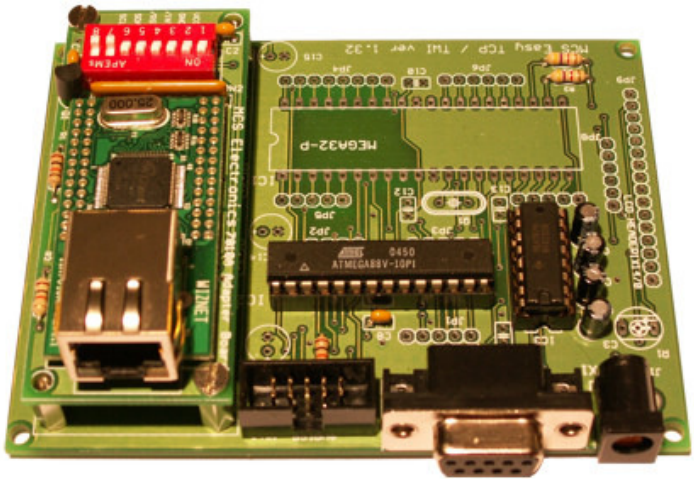
EMBEDDED SYSTEMS
BASIC COMPILERS
DEVELOPMENT

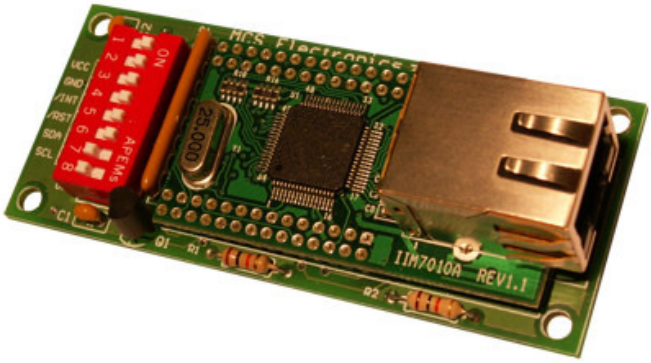
Version

1.6

MCS ELECTRONICS

Making Things Easy





Easy-TCP/IP
I²C / TWI interface

- Reduces the use of I/O pins (Only 4 pins needed for TCP/IP)
- Cost saving, by reducing components
- I²C / TWI enables low cost microcontrollers to interact with the Internet

Easy-TCP/IP

I²C / TWI interface

MCS ELECTRONICS

Easy-TCP/IP I²C / TWI Interface Guide

www.mcselec.com

Table of Contents

Introduction

A few years ago MCS Electronics introduced Easy-TCP/IP, which gave you a way of interfacing AVR microcontrollers to the Internet in a fast, cost effective way. Easy-TCP/IP is now being used in the world of embedded electronics.

We are now introducing the Easy-TCP/IP TWI – a TCP/IP interface, which uses only a few I/O pins of your microcontroller,and requires minimal change in your original program code. Easy-TCP/IP TWI was developed in response to our users' feedback.

The key advantage of the Easy-TCP/IP TWI is that it only uses four microprocessor pins, i.e., two I/O pins for I²C serial communications, and one interrupt pin and a reset pin. Thus **only 4 I/O** pins are needed to implement a full TCP/IP interface. The previously released TCP/IP board used about 16 I/O lines, making it only suitable for relatively large AVR controllers. If you are upgrading an application to TCP/IP, you are much more likely to have 4 I/O pins available than 16. And you do not need much more room for the changes needed in the program code.

The reduction of I/O pins of the Easy-TCP/IP TWI makes it slower than the EASY-TCP, but for a lot of embedded applications high speed is not a requirement. It's up to you to decide whether you choose speed, Easy-TCP or a compact and economic Easy-TCP/IP TWI. Using fewer pins requires that reading and writing be serial rather than parallel, which makes the TWI interface slower than the parallel interface.

This guide will help you implement Easy-TCP/IP TWI. It assumes you have some experience with the Easy-TCP/IP board. If you have not, please also read the East-TCP/IP manual, available from www.mcselec.com. The exact location is : http://www.mcselec.com/index.php?option=com_docman&task=doc_download&gid=97&Itemid=54

Note that Easy TCP/IP-TWI requires the 7010A-Adapter Board.
The Easy TCP/IP Motherboard is only intended to test this 7010A-Adapter Board.
You can also use an NM7010A module or the W3100A chip in your design.

Chapter 1

1. Getting started

This chapter explains how to get started with Easy-TCP/IP TWI.

If you have no experience with TCP/IP please read the TCP/IP manual available from www.mcselec.com. The TCP/IP manual describes general issues of TCP/IP and how to configure and test the EASY-TCP/IP series in your LAN.

The Easy-TCP/IP TWI board is meant to give you a way of experimenting with TCP/IP and BASCOM. If you wish to build your own complete solution we suggest you implement the NM7010A Adapter Board in your own system. The NM7010A is available from www.mcselec.com

1.1 Assembling your PCB

This chapter will take you thru the assembling of your EASY-TCP/IP TWI adapter and mother -board. Soldering the components can be done best in the order of the part list found on page 7.

On the PCB you will find pads like this: 

When soldering IC's or connectors, the square pad marks pin 1. For capacitors the square pad marks the positive (+) pole.

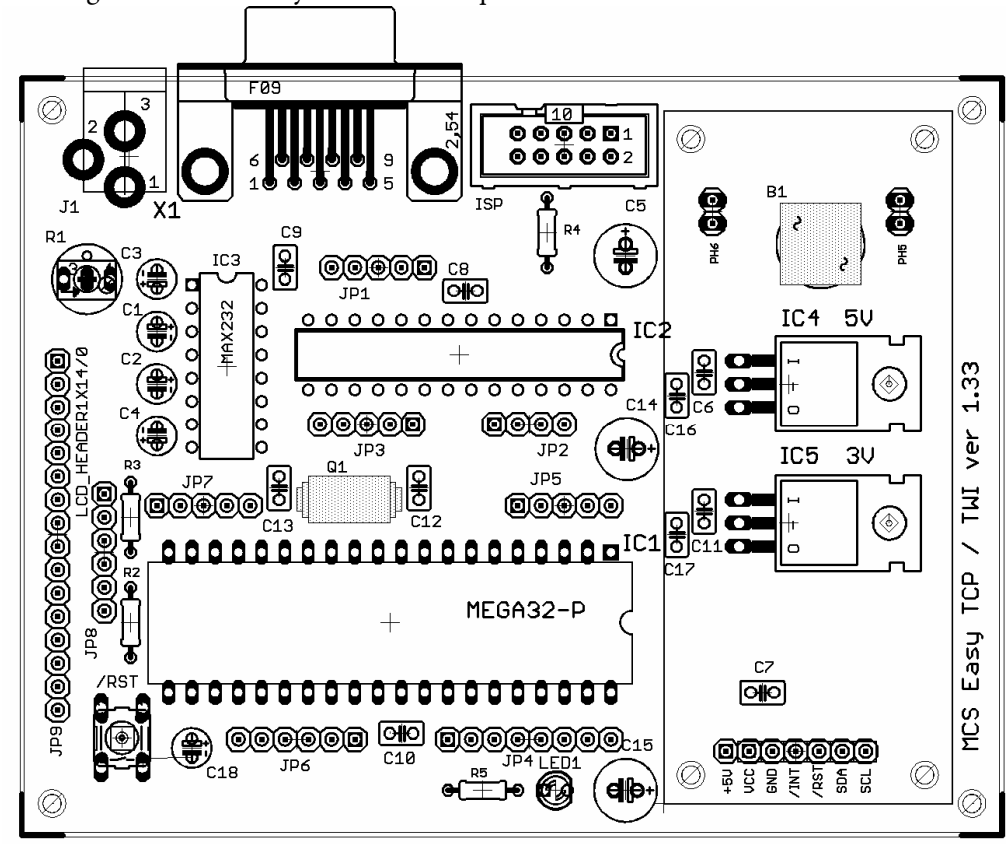
CAUTION

We advise you to use IC sockets rather than soldering the IC's directly to the PCB. You may want to use low quality IC sockets for the ATMegas, since it is easier to (re-) insert IC's into low quality sockets than into high quality (turned) sockets.

Do not insert the IC's at this time.

MOTHERBOARD LAYOUT (version 1.33)

To find the right position of the components you may want to reference this drawing of the board layout. It is also printed on the PCB itself.



PARTS LIST EASY-TCP/IP Motherboard (Controller)

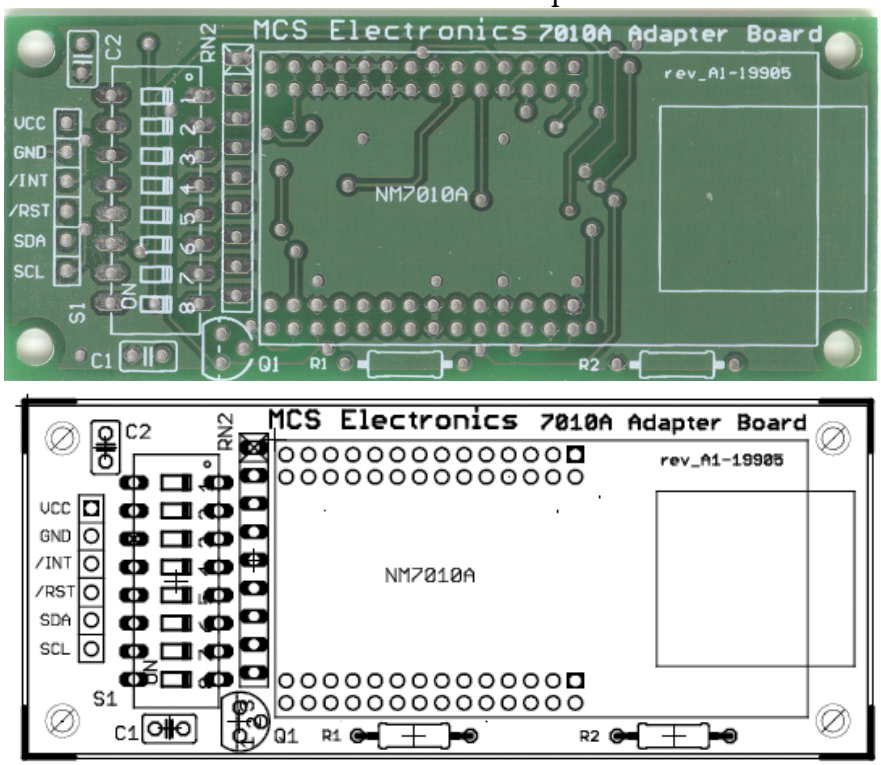
Component	Description	Value
R1	POTENTIOMETER . Only needed when LCD display is used.	1K
R2, 3	RESISTOR 1/8 W	4K7
R4	RESISTOR 1/8 W	10K
C1, C2, C3, C4	ELCO RADIAL	1uF/16V
C5	ELCO RADIAL	100uF/25V
C6, C8, C9, C10, C11, C16,	CERAMIC CAPACITOR	100nF

C17		
C7, C14, C15	ELCO Radial	10 uF/16V
C12, 13	CERAMIC CAPACITOR. Optional, only needed when you use an external oscillator.	22pF
Q1	CRYSTAL . Optional. Only needed when you do not want to use the micro processor internal oscillator.	8 MHz
IC4	VOLTAGE REGULATOR 5V	7805
IC5	**VOLTAGE REGULATOR 3V3 Only when using NON-SMD Adapter**	**LM2937ET- 3.3**
IC1	ATMEGA32 AVR. Or Mega163, Mega644, etc.	ATMEGA32
IC2	ATMEGA48 AVR or Mega88, Mega8, Mega168	ATMEGA88
	The board can be used with a 28 pin micro or a 40 pin micro, but not both at the same time. So only 1 microprocessor may be inserted!	
IC3	MAX232	MAX232
B1	BRIDGE RECTIFIER	B40R
JP1, JP3, JP5, JP7	PIN HEADER	1X5
JP2	PIN HEADER	1X4
JP4	PIN HEADER	1X8
JP6, JP8	PIN HEADER	1X6
JP9	PIN HEADER	1X16
JP10	AVR ISP HEADER	2X5
Adapter board	PIN HEADER FEMALE	1X6
J1	POWER CONNECTOR	
X1	DB9 CONNECTOR FEMALE	DB9FL
Misc	4 spacers and bolts	M3x6
PH5, PH6	2 pin header	
C18	Optional 100 nF capactiro	100 nF
RST	Reset switch	
R5	Resistor 330	330
LED1	LED 3mm Red	3 mm RED

Since there are 2 versions of the Adapter board, we describe them both.

Do not solder the ATMega's at this time, continue soldering the 7010 Adapter board. You are not to solder the micros at all.

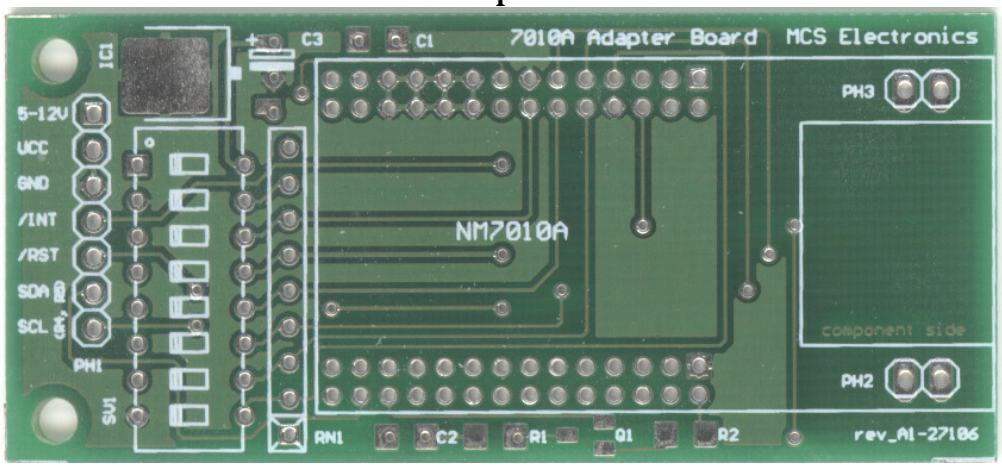
NON-SMD Adapter board

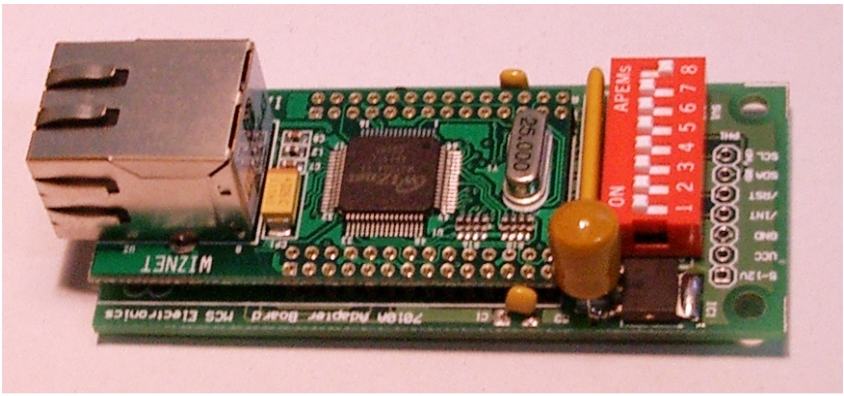


PARTS LIST EASY-TCP /IP 7010A NON-SMD Adapter Board

Component	Description	Value
R1,2	RESISTOR 1/4W	10K
C1, 2	CERAMIC CAPACITOR	100n
RN2	RESISTOR NETWORK	8 x 10kOHM
Connector	PIN HEADER **BEWARE: SOLDER ON TOP LAYER**	1X6
S1	DIP SWITCH	DIP07 or DIP08
Q1	PNP TYPE TRANSISTOR	BC307/BC557
M1	TCP/IP MODULE	NM7010A

SMD Adapter board





PARTS LIST EASY-TCP /IP 7010A SMD Adapter Board

Component	Description	Value
R1,2	RESISTOR 1/4W SMD	10K
C1, 2	CERAMIC CAPACITOR	100 nF
RN2	RESISTOR NETWORK	8 x 10kOHM
Connector	PIN HEADER **BEWARE: SOLDER ON TOP LAYER**	1X7
S1	DIP SWITCH	DIP08
Q1	PNP TYPE TRANSISTOR SMD	BC857
M1	TCP/IP MODULE	NM7010A
IC1	3v3 regulator SMD	BA03

C3	Tantalium elco	22uF/10V
PH3, PH4	2 pin header	

Do not place the adapter board on the main board yet.

Note 1 : The adapter board is connected to the Motherboard with a 6/7-pin male header. This header must be soldered on the board so it points downward. Solder it from the top of the board so that it can be inserted into the Motherboard female header, which will be below it.

BEFORE YOU CONNECT THE POWER

Once you have soldered all parts, check the PCBs. Remove any small solder dots between tracks, check you have soldered all the pins and that the solder joints look shiny (no cold solder joints). All contacts should look good.

Connect a power supply to the Motherboard. The power should be in the range from 7 – 12VDC. The higher the voltage, the more power must be dissipated by the regulator. So 7V is better than 12V but both will work.

The polarity is not important since a diode bridge is used. Measure the voltage at pin 3 of IC4. It should be 5V. Also measure the voltage at pin 2 of IC5. It should be 3.3V. The pinout of the 3.3V regulator is the same as the 7805 so be careful that the pinout is correct if you plan to use a different 3.3V regulator. When you use the SMD-Adapter board, it is not needed to insert the 3V3 regulator !

If the voltage is not correct, disconnect the power and check the board.

It is good practice to use a nut and bolt to fasten each regulator to the board to provide a heat sink.

If the voltages check out OK, disconnect the power and place ONE of the ATMega controllers. (You can use ATMega32 or ATMega88 or chips with the same pin outs) Also place the MAX232. At this point, it is a good idea to use a simple program to output something like “1” then “2” on the serial port in a do loop with a 1 second wait between transmissions to ensure that everything is working OK on the motherboard.

Check that the 7010A adapter board has no solder dots then connect the 7010A adapter to the main board.

The SMD-Adapter board has 7 pins. It has an on board 3V3 regulator. It can provide power to your own circuit too. Since the regulator needs an power source of 5V or more, this adapter board has an additional pin. It is named 5-12V on the PCB. When you set DIP switch 1, you connect the 3V3 regulated power to the VCC pin. Do this only when you do not supply the 3V3 Volt via the VCC pin !
Thus when you use the NON-SMD Adapter board, you need to insert the 3V3 Voltage regulator on the motherboard so the Adapter board will get 3V3.
But when you use the SMD-Adapter board which has it's own regulator, you must take care that you do not close DIP switch 1, AND use the 3V3 regulator of the motherboard. It is advised to not insert the 3V3 regulator on the motherboard in that case. But you can also leave the DIP switch 1 off, so the outputs of the two regulators are not connected to each other.

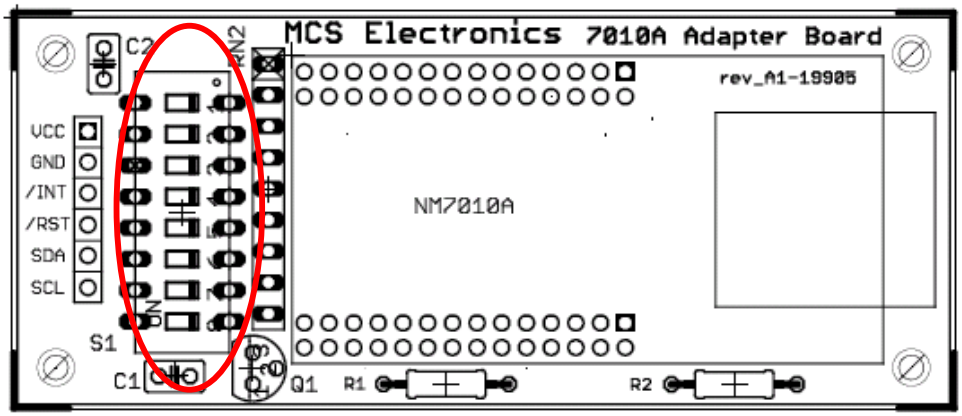
Chapter

2

2. Configuring the 7010A Adapter

This chapter explains how to configure the 7010A Adapter.

The only thing that needs to be configured is the 7010A's slave address. This can be done with DIP switch S1.



The TWI/I2C slave must have a unique address. The LS bit is used to indicate Read or Write and cannot be set. Therefore, it is not connected to the DIP switch. As DIP switches with seven switches are hard to find, the PCB is designed so that you can use a DIP switch with 7 or 8 switches. The switch number indicated with "8" is not connected. When you choose to insert a DIP with 7 switches, make sure that position 1 matches position 1 on the PCB and that position 8 on the PCB remains open/unused.

The W3100 chip needs pull up resistors to specify a one in its address. When the DIP switch is in the ON position, it connects the W3100 address line to ground, thus making it 0. When the switch is open, the pull up resistor will make the address line a "1".

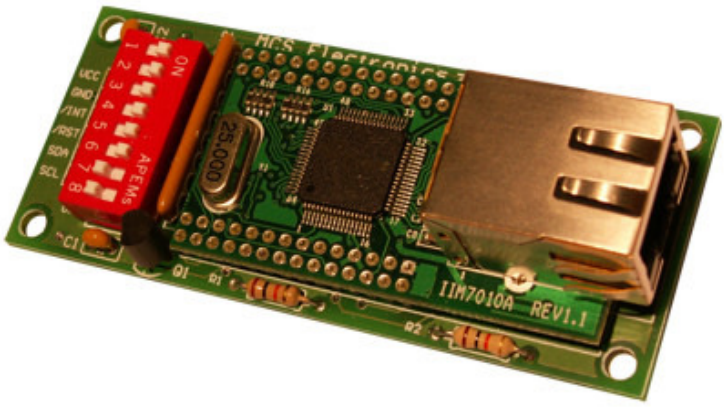
We start by selecting the address 0. We need to close all switches, which means that the switches must be pulled to the right.when the PCB is in the same position as shown above.

The “On” on the silk screen of the PCB might not match the “On” of the DIP switch. But make sure that the pin numbers of the DIP match the PIN numbers of the silk.

In the example programs, address 128(dec) is used. This is &H80(hex) or 10000000(bin).

As the MSB of the address matches pin 7 of the DIP, this switch must be pulled to the left so the address line will become “1”.

	NON-SMD	**SMD-Adapter**	
DIP PIN	**Address line**	**Address line**	**Remark**
1	A1	3V3 to VCC	A0 of the I2C can never be selected as it is the R/W bit. It is however part of the I2C slave address.
2	A2	A1	
3	A3	A2	
4	A4	A3	
5	A5	A4	
6	A6	A5	
7	A7	A6	
8	Not connected	A7	



The shown address is A7=1 , i.e., &B10000000 = &H80 = 128(dec). The switch at position 8 may be in any position since it is not connected.

This address must be specified with the CONFIG TCPIP command.

```
Config Tcpip = Int0 , Mac = 12.128.12.34.56.78 , Ip = 192.168.0.8 , Submask =
255.255.255.0 , Gateway = 0.0.0.0 , Localport = 1000 , Tx = $55 , Rx = $55 , Twi
= &H80 , Clock = 400000
```

This is in fact the only difference between a program that uses TWI or the high speed address mode.

You can now start experimenting with the EASY-TCP/IP TWI board. Make sure you keep your copy of BASCOM up to date in order to get new samples.

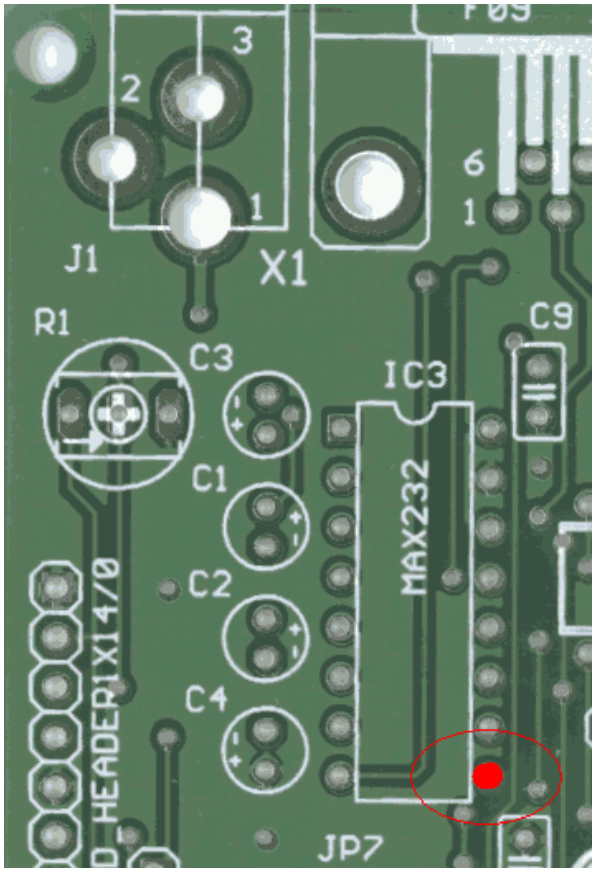
A new sample uses the new SNTP() function. To get a precise time.

```
'-----------------------------------------------------------------------------------------
'name                     : sntp.bas   RFC 2030
'copyright                : (c) 1995-2005, MCS Electronics
'purpose                  : test SNTP() function
'micro                    : Mega32
'suited for demo          : yes
'commercial addon needed  : no
'-----------------------------------------------------------------------------------------

'this demo also uses TWI/I2C with PCF8574A to show that the I2C is also available for general use
$lib "i2c_twi.lbx"

'$regfile = "m162def.dat"                                  'specify the micro used
$regfile = "m32def.dat"                                    ' specify the micro used

$crystal = 8000000                                         ' crystal frequency
$baud = 19200                                              ' baud rate
$hwstack = 80                                              ' default use 80 for the hardware stack
$swstack = 80                                              ' default use 80 for the SW stack
$framesize = 80                                            ' default use 80 for the frame space

Const Sock_stream = $01                                    ' Tcp
Const Sock_dgram = $02                                     ' Udp
Const Sock_ipl_raw = $03                                   ' Ip Layer Raw Sock
Const Sock_macl_raw = $04                                  ' Mac Layer Raw Sock
Const Sel_control = 0                                      ' Confirm Socket Status
Const Sel_send = 1                                         ' Confirm Tx Free Buffer Size
Const Sel_recv = 2                                         ' Confirm Rx Data Size

'socket status
Const Sock_closed = $00                                    ' Status Of Connection Closed
Const Sock_arp = $01                                       ' Status Of Arp
Const Sock_listen = $02                                    ' Status Of Waiting For Tcp Connection
Setup
Const Sock_synsent = $03                                   ' Status Of Setting Up Tcp Connection
Const Sock_synsent_ack = $04                               ' Status Of Setting Up Tcp Connection
Const Sock_synrecv = $05                                   ' Status Of Setting Up Tcp Connection
Const Sock_established = $06                               ' Status Of Tcp Connection Established
Const Sock_close_wait = $07                                ' Status Of Closing Tcp Connection
Const Sock_last_ack = $08                                  ' Status Of Closing Tcp Connection
Const Sock_fin_wait1 = $09                                 ' Status Of Closing Tcp Connection
Const Sock_fin_wait2 = $0a                                 ' Status Of Closing Tcp Connection
Const Sock_closing = $0b                                   ' Status Of Closing Tcp Connection
Const Sock_time_wait = $0c                                 ' Status Of Closing Tcp Connection
Const Sock_reset = $0d                                     ' Status Of Closing Tcp Connection
Const Sock_init = $0e                                      ' Status Of Socket Initialization
Const Sock_udp = $0f                                       ' Status Of Udp
Const Sock_raw = $10                                       ' Status of IP RAW
```

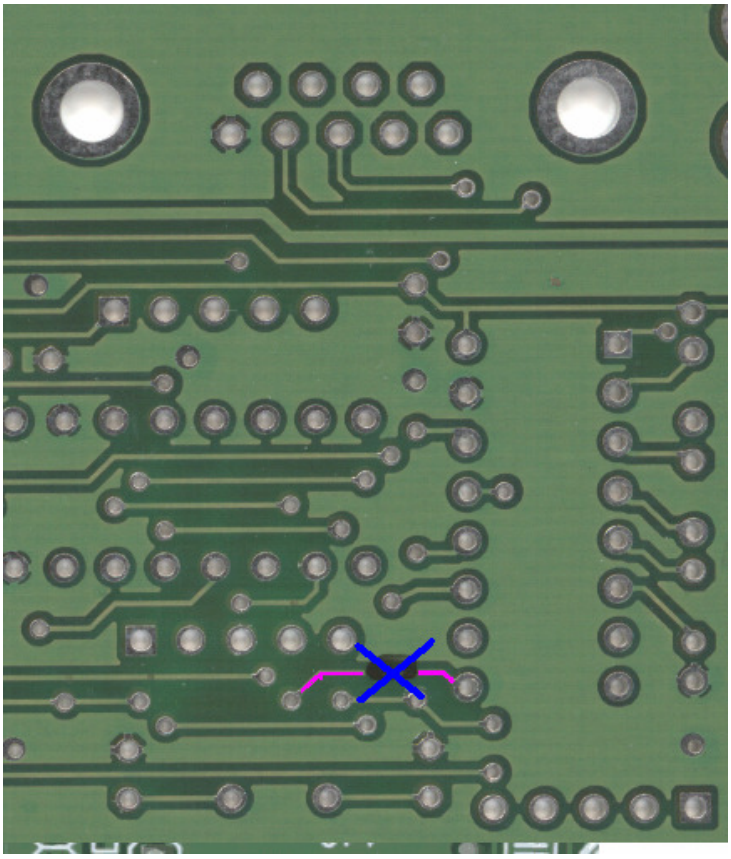
```
Print "Init TCP"            ' display a message
Enable Interrupts           ' before we use config tcpip , we need to enable the interrupts
Config Tcpip = Int0 , Mac = 12.128.12.34.56.78 , Ip = 192.168.0.8 , Submask = 255.255.255.0 , Gateway
= 0.0.0.0 , Localport = 1000 , Tx = $55 , Rx = $55 , Twi = &H80 , Clock = 400000

Dim Var As Byte                                         'for i2c test
Dim Ip As Long                                          ' IP number of time server
Dim Idx As Byte                                         ' socket number
Dim Lsntp As Long                                       ' long SNTP time
'-------------------------------------------------------------------------------------------
'When you use UDP, you need to dimension the following variables in exactly the order shown!
Dim Peersize As Integer , Peeraddress As Long , Peerport As Word
'-------------------------------------------------------------------------------------------

Print "SNTP demo"

'assign the IP number of a SNTP server
Ip = Maketcp(193.67.79.202 )                            'assign IP num  ntp0.nl.net port 37

'we will use Dutch format
Config Date = Dmy , Separator = -


'we need to get a socket first
'note that for UDP we specify sock_dgram
Idx = Getsocket(idx , Sock_dgram , 5000 , 0)            ' get socket for UDP mode, specify port
5000
Print "Socket " ; Idx ; " " ; Idx

'UDP is a connectionless protocol which means that you can not listen, connect or can get the status
'You can just use send and receive the same way as for TCP/IP.
'But since there is no connection protocol, you need to specify the destination IP address and port
'So compared to TCP/IP you send exactly the same message but with the addition of the IP and PORT
'The SNTP uses port 37 which is fixed in the tcp asm code

Do

   'toggle the variable
   Toggle Var
   Waitms 1000

   'now send the value of var to the PCF8574A
   I2csend &H70 , var , 1
   'this demonstrates that you can use the TWI for both the IIM7010 and other chips on the bus
   'A problem could arise when both chips need to be addressed
   'And as the IIM7010 is serviced via an interrupt, the main i2c traffic could slow down the TCP
traffic


   Lsntp = Sntp(idx , Ip)                               ' get time from SNTP server
   'notice that it is not recommended to get the time every sec
   'the time server might ban your IP
   'it is better to sync once or to run your own SNTP server and update that once a day

   'what happens is that IP number of timer server is sent a diagram
   'it will put the time into a variable lsntp and this is converted to BASCOM date/time format
   'in case of a problem the variable is 0
   Print Date(lsntp) ; Spc(3) ; Time(lsntp)
Loop

End
```

The Easy TCP/TWI Motherboard uses the DTR line to control the RESET of the micro processor. This way you can use the MCS Bootloader to reset the board automatic when you update/load the program.

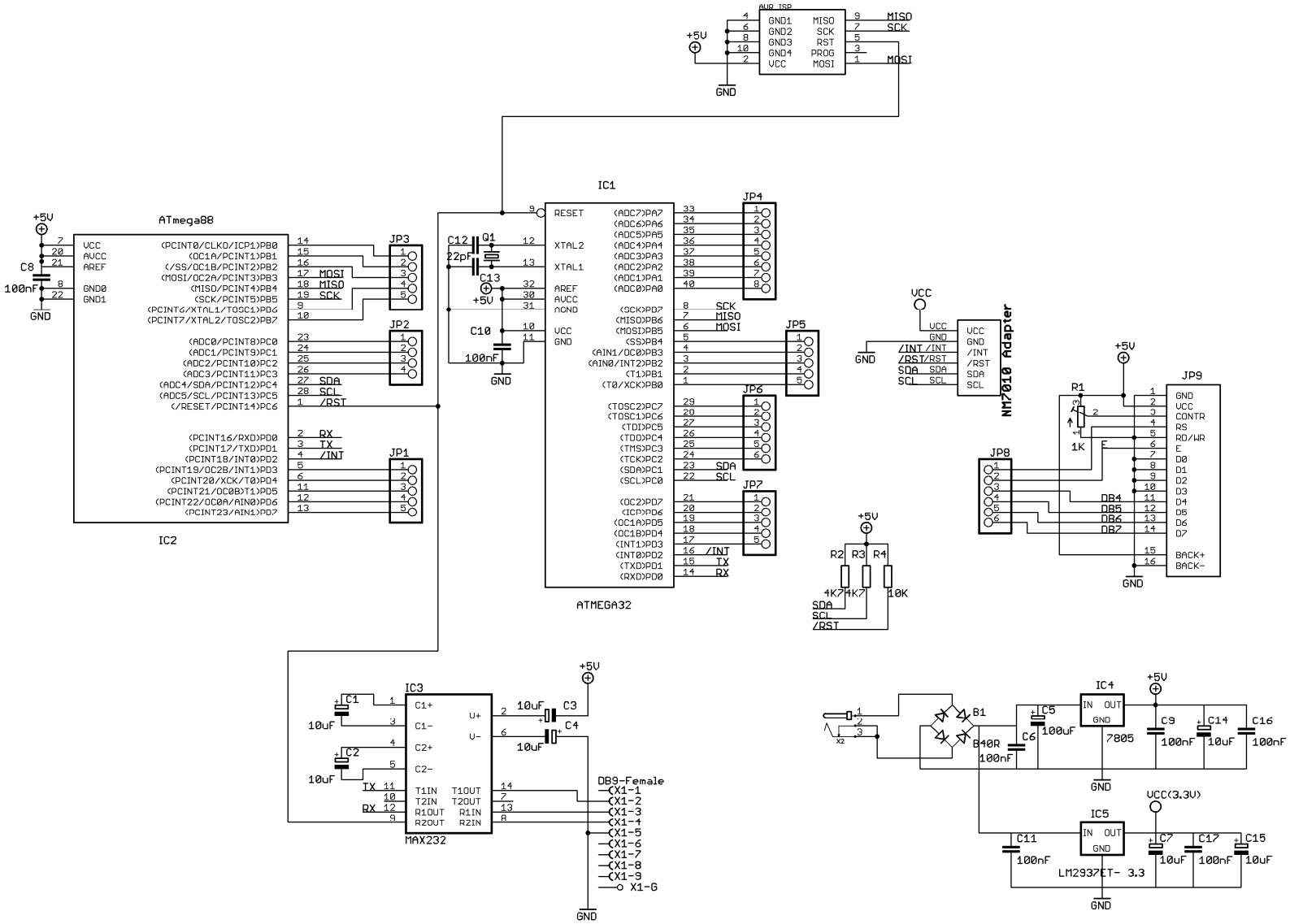
But when you do not use the MCS Bootloader, the micro will be reset when DTR is not made low. Third party terminal emulators might also set DTR high. When you want to remove this option, you need to cut a track of the PCB. You can disconnect pin 9 of the MAX-232. Or you can cut the track on the bottom of the PCB. When you bend pin 9 of IC3, you do not need to change the PCB and all options are still usable.



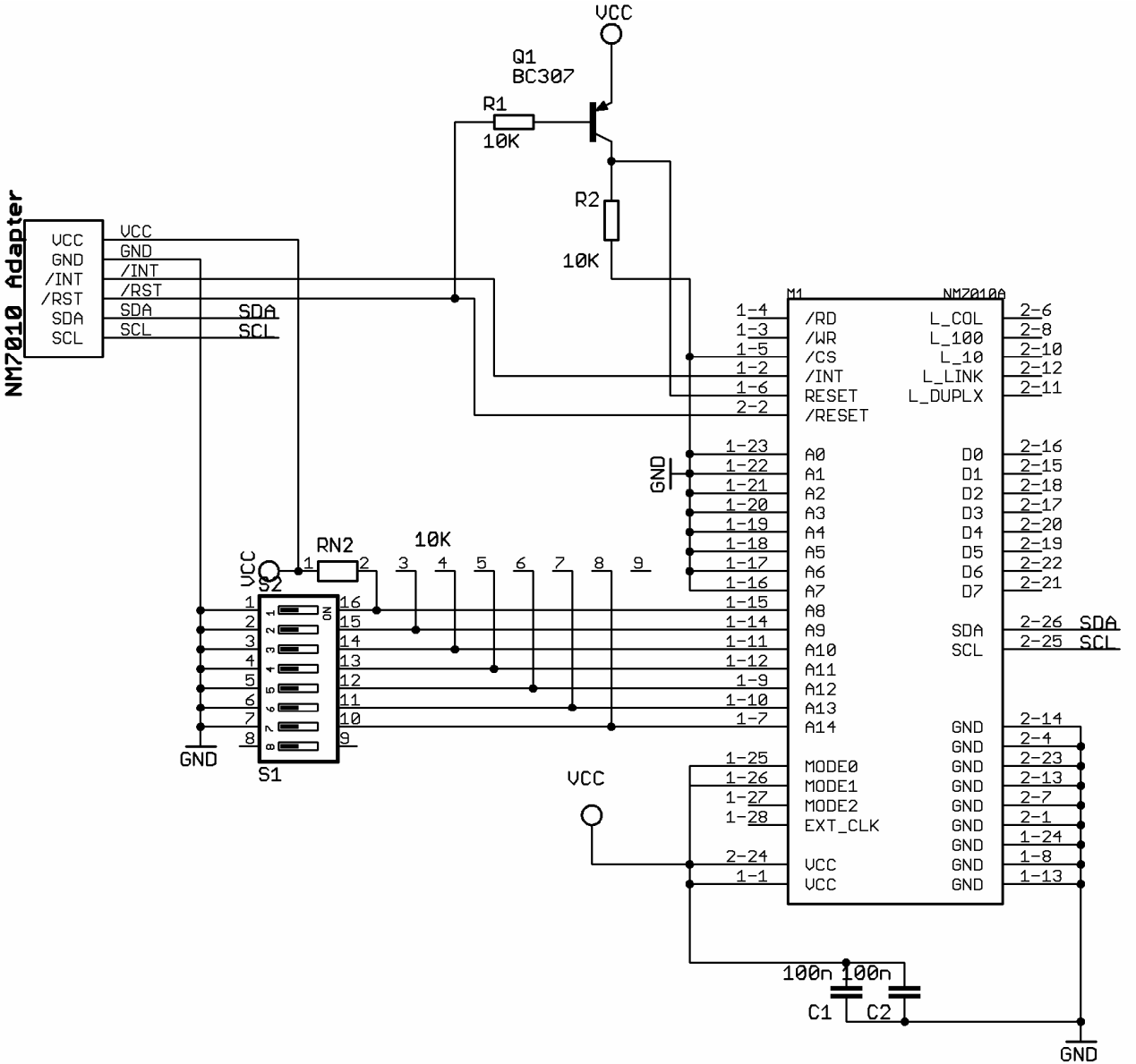
Option to remove pin 9



Option to remove track



Mother Board


NM7010 Adapter
VCC
GND
/INT
/RST
SDA
SCL
Q1
BC307
R1
10K
R2
10K
RN2
10K
S1
S2
M1
NM7010A
/RD
/WR
/CS
/INT
RESET
/RESET
A0
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
A11
A12
A13
A14
MODE0
MODE1
MODE2
EXT_CLK
L_COL
L_100
L_10
L_LINK
L_DUPLX
D0
D1
D2
D3
D4
D5
D6
D7
SDA
SCL
GND
100n
C1
C2

WIZnet
www.wiznet.co.kr

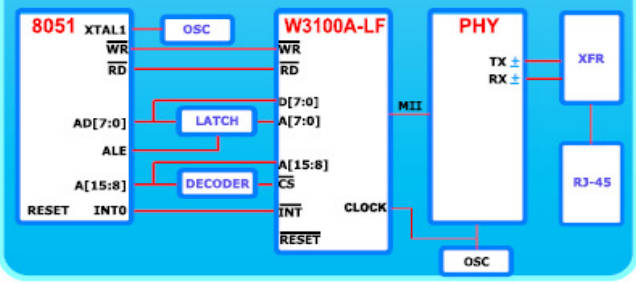
? purchase inquiry
sales@wiznet.co.kr

Hardwired TCP/IP Chip, W3100A-LF

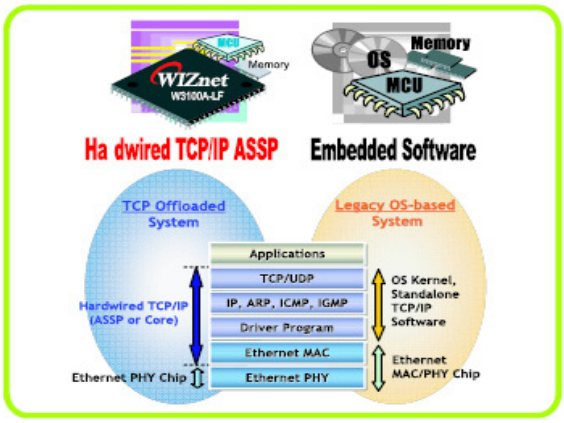
Features

- TCP,UDP,IP,ICMP,ARP,MAC hardwired logics included
- Supports 4 independent channels simultaneously
- Up to 12Mbps data transmission speed
- Dynamic buffer allocation for each channel
- MCU bus interface and I^2C serial interface for MCU
- Standard MII interface for physical layer
- 16KBytes data buffer embedded
- 10/100 Base-T auto detection
- 3.3V internal operation, 5V tolerant IOs
- 64 Pin LQFP package (RoHS)
- Simple socket API interface
- DHCP,TELNET,FTP,HTTP,DNS,SNMP source codes are provided with EVB
- API and all source codes are written in C

Sample Schematic

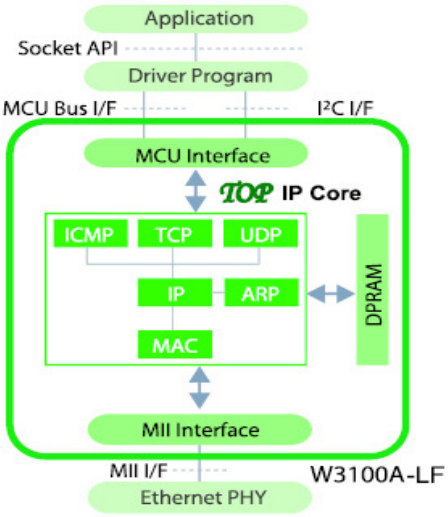


Benefits of Hardwired TCP/IP



Easy-to-Use: easiest memory-like way to access the Internet
Full hardware processing of Internet protocols
MCU offload : get 8-bit MCU connected without networking overhead

Block Diagram



Core Technology

TOP (TCP/IP Offload Platform) is Market-proven ASSP and Silicon-proven Custom ASIC.

TOP is a Fully Hardwired TCP/IP algorithm, which guarantees line speed by on-the-fly processing architecture that is independent of CPU.
TOP eliminates CPU overhead by offloading TCP/IP processing and hence enhences overall system performance esp. in multimedia streaming application.

WIZnet
www.wiznet.co.kr

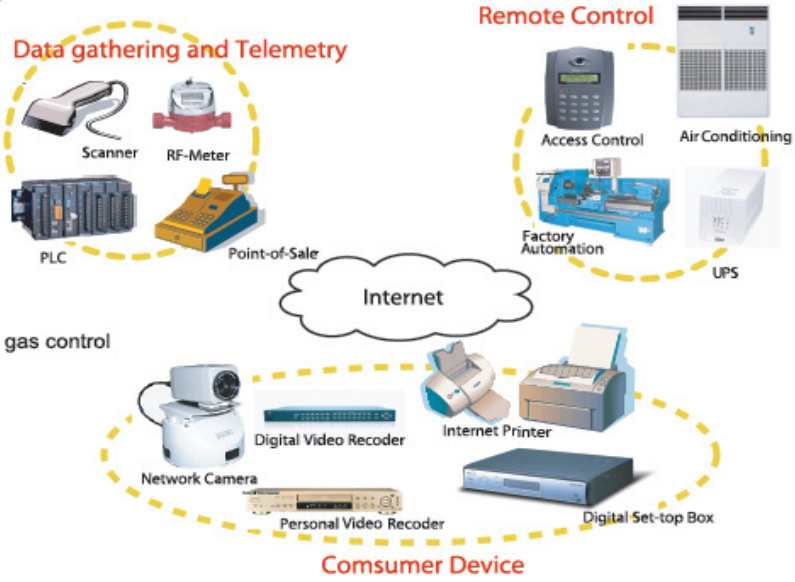
Hardwired TCP/IP Chip, W3100A-LF

Application

Data gathering and Telemetry
Serial-to-Ethernet
CAN-to-Ethernet
RF-to-Ethernet
Point-of-Sale

Remote Conrtol
Access Control
Factory automation
Air conditioning, temperature humidity gas control
Power distribution

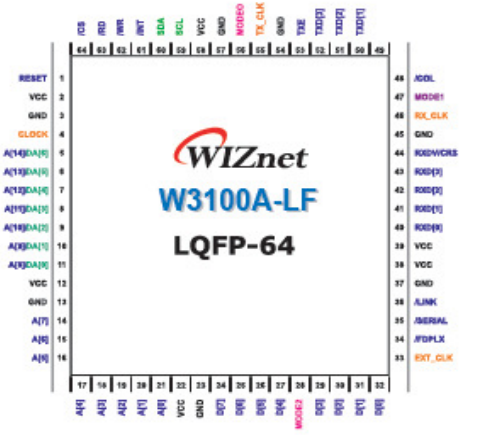
Consumer Devices
Digital video recorder
Personal cideo recorder
Digital Set-top box
Internet Printer
Network camera



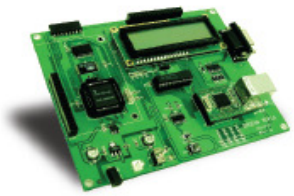
Socket API

Functions	Operation
Sysinit	Initialize W3100A-LF
Socket	Create a socket
Connect	Attempt to open a TCP connection
Listen	Accept a TCP connection
Send	Transmit data through a TCP socket
Recv	Receive data through a TCP socket
Sendto	Transmit data through a UDP or RAW socket
Recvfrom	Receive data through a UDP or RAW socket
Close	Close a socket

Pin Assignment



Evaluation Board



EVB8051
(ATMEL 89C51)

EVBAVR
(ATMEL AVR)

Whole source codes & schematics are fully available

WIZnet
www.wiznet.co.kr

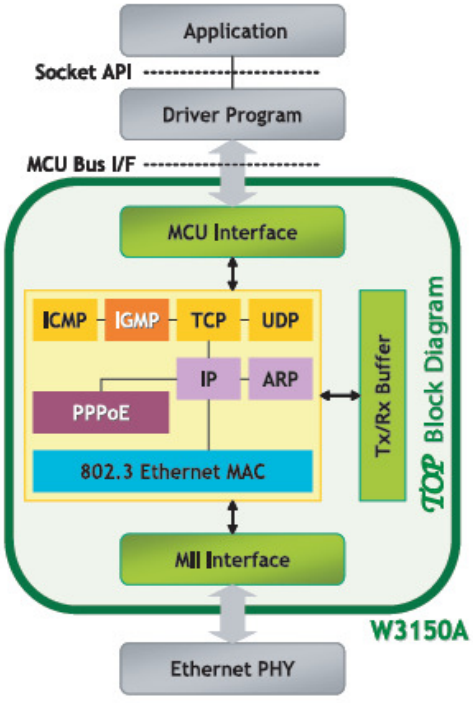
? purchase inquiry
sales@wiznet.co.kr

Hardwired TCP/IP Chip, W3150A

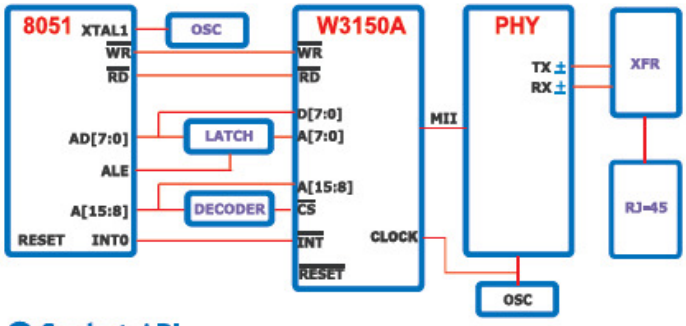
Features

- TCP, UDP, IP, ICMP, IGMP, ARP, PPPoE, MAC in Fully Hardwired Logic
- DHCP, TELNET, FTP, HTTP, DNS source codes are provided
- Supports 4 Independent Sockets Simultaneously
- 10/100 Base-T Ethernet (Auto Detection)
- High Throughput; 8051(max. 1Mbps), AVR(max. 8Mbps), ARM7(max. 20Mbps)
- Full-duplex Mode
- Standard MII Interface for Physical Layer
- 16Kbytes Data Buffer Embedded
- Simple Socket API Interface
- 3.3V Operation, 5V Tolerant I/Os
- Commercial Temperature Range from 0°C to 80°C
- All Source Codes are written in ANSI C totally free
- 64 pin LQFP Level 2 Lead-free Package

Block Diagram



Sample Schematic



Socket API

Functions	Operation
Sysinit	Initialize W3150A
Socket	Create a Socket
Connect	Attempt to Open a TCP Connection (Active Open)
Disconnect	Attempt to Close Terminate a TCP Connection
Listen	Accept a TCP Connection (Passive Open)
Send	Transmit Data through a TCP Socket
Recv	Receive Data through a TCP Socket
Sendto	Transmit Data through a UDP or RAW Socket
Recvfrom	Receive Data through a UDP or RAW Socket
Close	Close a Socket

Core Technology

TOP (TCP/IP Offload Platform) is Market-proven ASSP and Silicon-proven Custom ASIC.

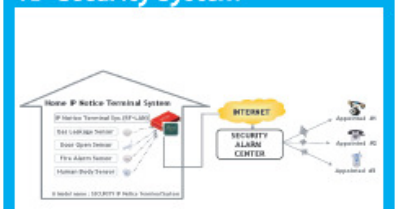
TOP is a Fully Hardwired TCP/IP algorithm, which guarantees line speed by on-the-fly processing architecture that is independent of CPU.
TOP eliminates CPU overhead by offloading TCP/IP processing and hence enhences overall system performance esp. in multimedia streaming application.

WIZnet
www.wiznet.co.kr

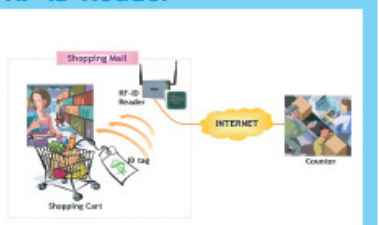
Whatever device needs embedded internet, W3150A is the TOP choice.

TOP TCP/IP Offload Platform
www.iomad.net

RF-Security System



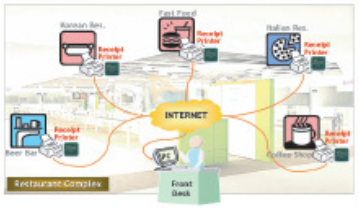
RF-ID Reader



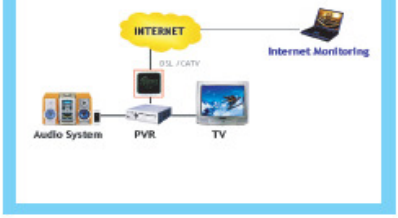
DVR

Serial-to-Ethernet

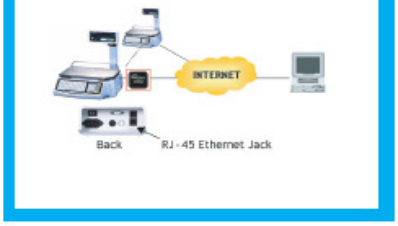
Internet Printer



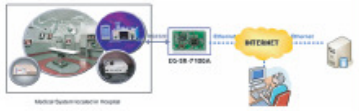
PVR



Electronic Scale

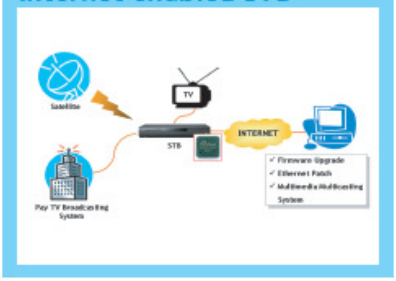


Medical Devices

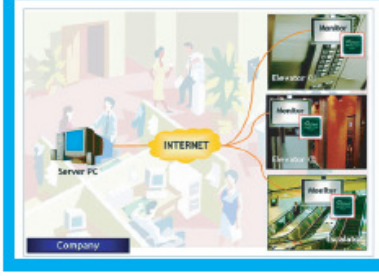


Air Conditioner

Internet enabled STB



LCD Monitor

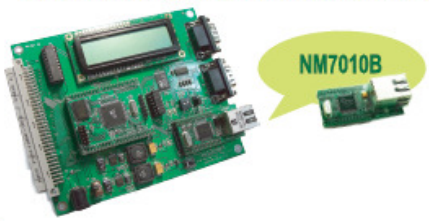


Evaluation Board for W3150A, "EVB-B1"

*** Feature**

- Full-duplex Data Transfer up to 8Mbps with Atmega128
- Internet Application Protocols: ANSI C Source Codes such as DHCP, etc.
- By using External 96 pin connecters, AVR's all pins can be used for developer

Product		Ordering Code	
EVB-AVR-W3150A	Base Board	EVB-B1	MB-EVB-X1
	MCU Module		PM-A1
	Network Module		NM7010B



*** Contents**

- Base Board + MCU Module + Network Module
- CD (Source codes, Schematics, Manuals)
- 12V 500mA Power Adapter
- Serial cable, UTP cable
- AVR-ISP cable