

Table of Contents

Introduction	2
Connecting Power.....	8
Connecting Ethernet Port.....	8
Connecting Serial Port.....	8
Switch SW1 Settings	8
Factory Default Settings	10
Configure SE-100M-EV	11
Install Java Configuration Utility.....	12
Serial and Data Packing Settings.....	14
TCP/IP Network Settings.....	14
Device Name Settings.....	15
Access Control Settings	15
Save Configuration	17
Save Configuration to File.....	17
Overview	19
Overview	21
Serial Console Mode.....	21
Telnet Console Mode.....	22
Forgot Password	23
Command Syntax.....	23
Echo Syntax.....	28

1

Introduction

Overview

The SE-110S is a stand along Serial-to-Ethernet embedded module which adds network and Web server function for legacy RS-232/422/485 devices. The SE-110S provides a transparent data channel between serial device and the TCP/IP network, so that the serial device can be accessed by using standard TCP/IP protocols. The SE-110S can behave as a TCP server, a TCP client or an UDP node on a Ethernet network to fulfill versatile application requirements.

Programmable Digital I/O

Besides, the SE-110S also provides 4 TTL digital I/Os. A pre-defined ASCII command set is provided for users to control these I/Os through LAN/WAN.

Product Specifications

Form Factor

Type: stand alone module with pin header connector

WxLxH: 57x40x15 mm

Network Interface

Type: 10/100BaseT, auto-detect

Protocols: TCP, UDP, HTTP, Telnet, IP, ICMP, ARP

IP addressing: DHCP, Static IP

Operation Modes

TCP/Server, TCP/Client and UDP mode

Serial Interface (TTL-level)

Signals: TxD, RxD, RTS, CTS, DTR, DSR, DCD, GND

Baud: 1,200 to 38,400 bps

Parity: None, Even, Odd

Data bits: 7,8

Stop bit: 1,2

Flow control: None, RTS/CTS, XON/OFF

Programmable Digital I/O

PIO0~PIO3: TTL level compatible

Configuration Methods

Web console, Telnet Console and Serial Console

Windows utility (included in CD)

General

Power input: 5VDC

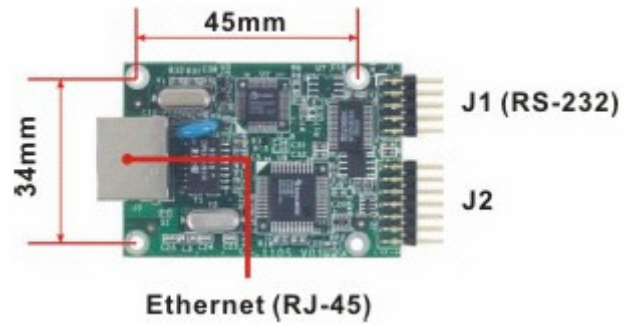
Power consumption: 100mA@+5VDC

Operation temp.: 0~70C, 5~95% RH

Storage temp.: -20~85C, 5~95% RH

Warranty: 2 years

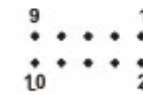
Dimensions



Pin Assignment

J1 (RS-232)

PIN	GPIO
1	DCD
2	Rx
3	Tx
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	---
10	---

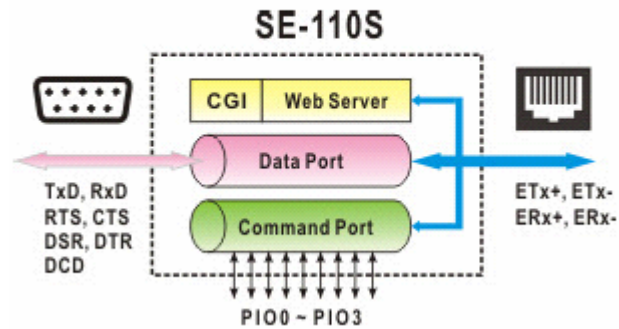


J2

PIN	GPIO
1	GND
2	SW1
3	Reset
4	PIO3
5	Ready
6	PIO2
7	100M_LED
8	PIO1
9	10M_LED
10	PIO0
11	+5V
12	GND
13	+5V
14	GND



Block Diagram



2

Getting Started

Connecting Power

Connecting +5VDC power line to pin 11 or 13 and ground to pin 12 to 14 of J2 connector.

Connecting Ethernet Port

Connect a RJ45 Ethernet cable to the Ethernet port of SE-110S. The 10M_LED or 100M_LED pin will be output of CMOS high signal to show the network speed if Ethernet cable is connected to the network.

Connecting Serial Port

Use CBL-F10M9-20 10-pin to DB9 serial cable to connect J1 of SE-110S. The CBL-F10M9 can be purchased separately from Artilla or home made by user. The serial port (J1) is a standard RS-232 serial port. (line driver is included)

SW1 Pin Setting

Connecting SW1 pin to ground will force SE-110S to enter serial

console mode. In serial console mode, user can use text command to configure SE-110S. Please refer to the command syntax section for the information to configure SE-110S

ATTENTION



In serial console mode, Ready pin signal will flash high and low and the Ethernet function will be disabled. All the new configuration will be effective after the next system boot.

Reset Pin Setting

To implement a hardware reset button, user can simply connect Reset pin and Ground pin with a push button.

Ready Pin Setting

When SE-110S firmware is ready, the Ready signal will be an output of TTL level high

PIO Pin Setting

DIO0 to DIO3 are TTL compatible Programmable DIO. All the DIO channel are internally pulled up to +5VDC with a 4.7K Ohm resistor.

The DIO can be controlled by Telnet Console or Serial Console and

command are described in Command Syntax section.

Factory Default Settings

The factory default settings are:

IP Address: 192.168.2.127

Data Port: 4000 (configurable)

Telnet Port: 5001

Web Port: 80

Baud rate: 19200

Data Format: N,8,1 (Parity, Data bits, Stop bits)

Flow Control: None

Interface: RS-232

Configure SE-110S

SE-110S provides four ways to configure the settings. They are:

1. Java Configuration Utility
2. Web Configuration Utility
3. Serial Console
4. Telnet Console

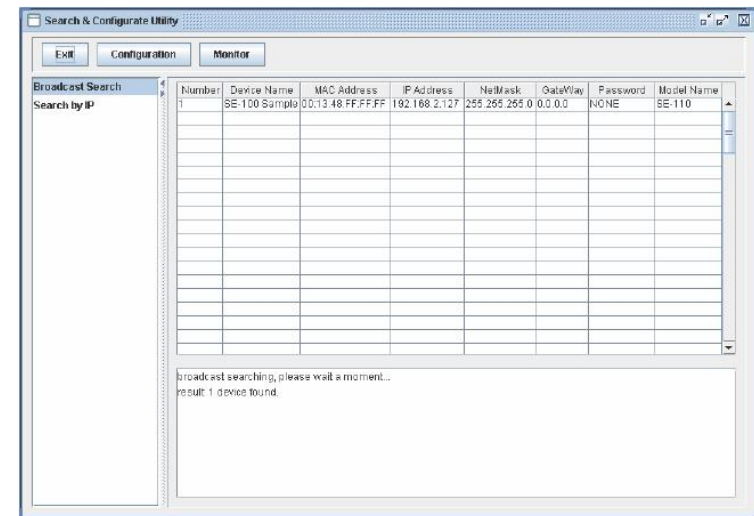


To use the Java configuration utility software, you need to install the Java 2 Platform, Standard Edition (J2SE) version 1.4.2 or later. J2SE is free and available at

<http://java.sun.com>

Install Java Configuration Utility

To install the Java Configuration Utility, you can simply copy **manager.jar** to the desired folder. Double click on the icon will start the **manager.jar**



Once this program started, it will perform a broadcast search to find the SE-110S device in the network. If you cannot find the SE-110S module and you know the IP address of the module, you can click **Search by IP** item and specify the IP address to find the SE-110S

SE-110S. The default IP address of SE-110S is **192.168.2.127**. Click on the row of the SE-110S settings, the configuration window will open as follow:



ATTENTION *When the SW1 is set to Console mode, SE-110S cannot be discovered by the Configuration Utility and the configurations are controlled by serial console.*

When enter the configuration mode, the READY signal will be flashing high and low and TCP Data Port will be disabled and data transfer between serial port and Ethernet port will be stopped.

Serial and Data Packing Settings

The group of setting is to configure the serial interface and data packing settings. Data Packing setting is are as follow:

Length: Pack the length of serial data before forwarding data to Ethernet port

Timeout: the period of time to forward data to Ethernet Port

Delimiter: Wait for the Delimiter Character before forwarding data to Ethernet port. The Delimiters are maximum two bytes Hex format ASCII code. If you use Carriage Return (CR) and Line Feed (LF) as delimiters, you can specify 0D0A or 0d0a.

TCP/IP Network Settings

The group of TCP/IP Network settings are configuring the IP Address, TCP port and operation mode as follow:

OpMode: TCP operation mode setting

TCP Port: TCP Port number of SE-110S

Destination IP: The IP address of remote host which SE-110S will actively connect to (TCP Client mode only)

Connect At: Startup means TCP connection is established when system starts (TCP Client mode only)

Timeout: When this option is checked, TCP connection will be disconnected if there is no serial data activity before timeout

IP Mode: Configure the IP Address to be Static IP or Dynamic IP by DHCP

IP Address: Set the Static IP Address

Subnet Mask: Subnet Mask setting

Gateway: Gateway address setting

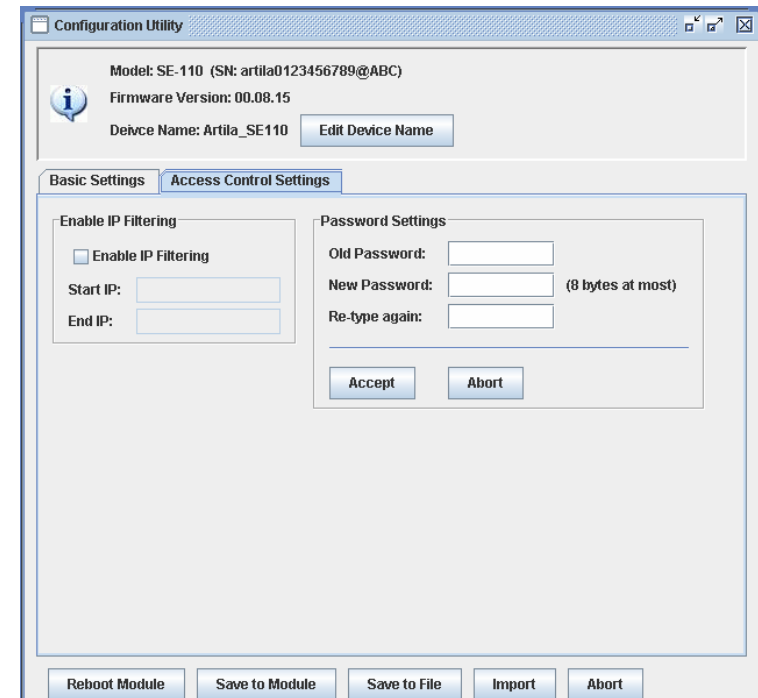
Device Name Settings

Click Edit Device Name button can edit the device name



Access Control Settings

SE-110S provides IP address filtering method and password authentication for access control



Enable IP Filtering: When this option is checked, user can specify the starting IP address and ended IP address which are allowed to access SE-110S TCP port in order to prevent unauthorized access.

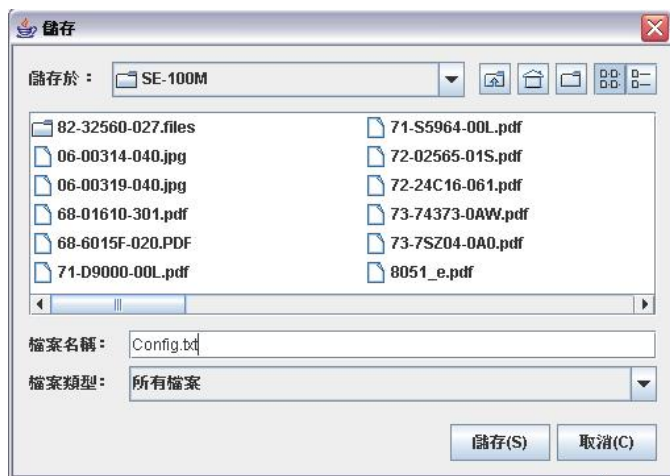
Password Settings: Enter the new password will enable the password authentication. Password is required to login Java Configuration and Web configuration Pages in the next entry.

Save Configuration

After the all the settings are configured, click to the **Save to Module button** to save the settings to the module. The new settings will be effective after the reboot of module by clicking the **Reboot Module** button. If the IP address had been modified, please do broadcast search or search by IP again to find the SE-110S module.

Save Configuration to File

You can also save current configuration to file by click the **Save to File** button. A new window to specify the path of the file will pop out as follow:



The configuration is a text file and uses extension of txt. If you use Text editor to open the configuration command, you will notice that the configuration file contains the ASCII command set of the configuration. You can also edit these commands by the text editor to configure the settings and import it to the SE-110S by using the Import button.

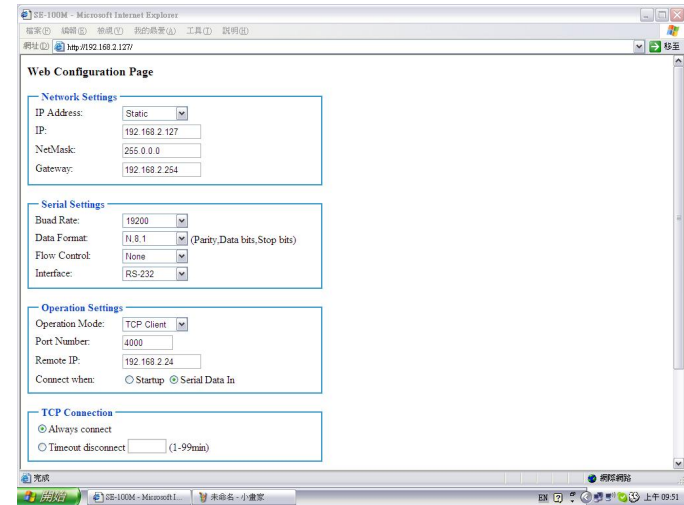


4

Web Configuration

Overview

SE-110S can also be configured by Web Browser. The build-in Web server and Web configuration pages makes SE-110S configurable anywhere via a Web browser such as IE and Firefox. To open the Web configuration pages, you can simply type the IP address to the Web Address input such as 192.168.2.127. The Login windows will ask you enter password if the password option is enabled. After password confirmed, the web configuration page will show up as follow:



The configuration is very similar to the Java configuration utility. After completing the settings, click submit button then all the settings will save to the module and the module will reboot automatically to use the new configuration.

5

Serial/Telnet Command

Overview

SE-110S provides a set of ASCII command to configure it through a serial and Ethernet port. User can use serial terminal or Telnet command to configure SE-110S when Web browser and Java utility are available. In addition, these functions provide user the most convenient way to develop their own configuration utility software simply to use the ASCII command.

Serial Console Mode

To enter the Serial console mode, you need to short SW1 signal to ground and then serial port will function as a console port. The READY signal will keep flash that indicates SE-110S is in serial console mode. Please set the serial data setting of the Serial Terminal software (such as Hyper Terminal) to be the same as the SE-110S serial port.

Baud rate: 19200

Data Format: N,8,1 (Parity, Data bits, Stop bits)

Flow Control: None

Interface: RS-232

Now you can use serial terminal software to send the ASCII command to SE-110S.



In Serial Console Mode, all the network functions will be disabled

Once the configuration is completed, remember to disconnect SW1 to ground and reboot the module by Reset Command or push reset button.

Telnet Console Mode

SE-110S uses Port number **5001** as telnet console port.

Remember to turn on the LOCALECHO before opening the telnet console port. You must login first before sending command to SE-110S. If password is enabled, you need to use the password to log in. The password is encrypted using Tiny Encryption Algorithm (TEA) and the keys are sixteen bytes with low case character from “a” to “p”. The login command format is as follow:

\$LOG[16 bytes of TEA Encrypted Password]

or

\$LOG

if there are no password settings

Forgot Password

If you forgot the Password, you can switch SW1 to Factory Default setting mode and use default IP Address **192.168.2.127** and enter the Web console pages. Once the configuration is completed, remember to switch SW1 back to Normal operation mode and reboot the module by Reset Command or push reset button. Password is not required for serial console mode. Therefore you can also use serial console to reset the password.

Command Syntax

The command syntax is

[Delimiter][Command][Carriage Return/CR][Line Feed/LF]

Delimiter (one byte): The command begins with a delimiter of dollar sign \$ and if SE-110S receives the correct command it will response with an echo which begins with the delimiter of a percentage sign %.

Command: The command are ASCII string which contains three elements: {Read/Write} {Function} {Parameter} as described as follow:

Command Category	R	W	Function (2 bytes)	Parameters (command related)
Basic	<i>G</i>	<i>S</i>	<i>BN</i> : Device Name	Max. 14 bytes
	<i>G</i>		<i>BM</i> : Model Name	Default
		<i>S</i>	<i>BP</i> : Password	Max. 8 bytes
	<i>G</i>		<i>BV</i> : Firmware ver...	Default
Serial	<i>G</i>	<i>S</i>	<i>SB</i> : Baud Rate	<i>1200</i> <i>2400</i> <i>4800</i> <i>9600</i> <i>19200</i> <i>38400</i>
	<i>G</i>	<i>S</i>	<i>SD</i> : Data Format (Parity,Data,Stop)	<i>N72</i> <i>E71</i> <i>O71</i> <i>N81</i> <i>E72</i> <i>O72</i> <i>E81</i> <i>O81</i> <i>N82</i>
	<i>G</i>	<i>S</i>	<i>SF</i> : Flow Control	<i>NONE</i> <i>RTS/CTS</i> <i>XON/XOFF</i>
	<i>G</i>	<i>S</i>	<i>SI</i> : Interface	<i>RS-232</i>

				RS-422 RS-485
Network	G	S	LI: IP Address	e.g. \$\$LI192.168.2.127
	G	S	LN: Netmask	e.g. \$\$LN255.0.0.0
	G	S	LG: Gateway	e.g. \$\$LG192.168.2.254
	G	S	LC: IP Mode	0: Static IP 1: DHCP
	G		LM: MAC	Default
	G		LS: LAN Status	Link fail Link OK,100M Link OK, 10M
	G	S	LW: TCP windows size	0:128, 1:256, 2:512, 3:1024 e.g. \$\$LW0 means TCP window size 128 bytes
Operation	G	S	OM: Operation Mode	TS: TCP Server TC: TCP Client
	G	S	OC: TCP connection timeout option	AC: always connect TO: Disconnect if timeout
	G	S	OU: Timeout clock	0~65535 (unit: ms) e.g. \$\$OU500
	G	S	OD: Data Packing	Three parameters: { [0]:Delimiter Option

				[1]: Timeout Option [2]: Length Option }={I:Enable,0:Disable} e.g. \$\$OD110
	G	S	OS: Delimiter	Max. two bytes characters e.g. \$\$SOS0D0A (CR/LF)
	G	S	OT: Timeout	0~65535 (unit: ms) e.g. \$\$SOT500
	G	S	OL: Data Length	0~1024 e.g. \$\$SOL20
Access Control	G	S	OF: IP Filtering	0: Disable 1: Enable
	G	S	OI: Authorized IP	e.g. \$\$OI192.168.2.127:192.168.2.180 (Starting IP:Ended IP)
TCP Server	G	S	VP: Listen Port	1~65535 e.g. \$\$SVP4000
TCP Client	G	S	CI: Destination IP	e.g. \$\$CI192.168.1.211
	G	S	CP: Destination Port	e.g. \$\$SCP4001
	G	S	CC: Connection Logic	0: Reserved for PPPoE 1: Establish connection when Serial Data in
System		S	YC: System mode	R: Reboot system F: Reset to default setting and reboot

			<i>D:</i> Set current setting as Default setting
	<i>S</i>	<i>EI:</i> Exit console	
DIO	<i>S</i>	<i>DM:</i> Set Digital I/O mode	<i>I:</i> Input <i>O:</i> Output e.g. <i>SSDMI:0,4,5</i> set DIO0, DIO4, DIO5 as input <i>SSDMO:1:H,2:L,3:H</i> set DIO1,DIO2,DIO3 as output and initial state are High (H) or Low (L)
	<i>G</i>	<i>DM:</i> Get Digital I/O mode	e.g. <i>SGDM</i> the response is <i>%GDMI,O,O,O,I,I,O,O</i>
	<i>S</i>	<i>DOH:</i> Set DO channel High Output	e.g. <i>SSDOH:1,3,7</i>
	<i>S</i>	<i>DOL:</i> Set DO channel Low Output	e.g. <i>SSDOL:2,6</i>
	<i>G</i>	<i>DO:X</i>	e.g. <i>SGDO:2</i> get DIO2 status
	<i>G</i>	<i>DI:</i> Get DI channel status	e.g. <i>SGDI</i> the response is <i>%GDI 0=1,4=0,5=1</i> It stands for DIO0=1,DIO4=0,DIO5=1

				e.g. <i>SGDI:4</i> Get DIO4 status
--	--	--	--	---------------------------------------

Remark: *Italic font stands for Command String*

CR/LF (two bytes): Both command from host and echo from SE-110S are terminated with a Carriage Return (ASCII code Hex 0d) and Line Feed (ASCII code Hex 0a). Therefore please remember to add CR/LF at the end of command line in the terminal emulation program.



All command strings are in Upper Case

Echo Syntax

After SE-100M-EV console port received correct command string, it will response Echo String and the Echo Syntax is as follow:

[Delimiter][Function][Status]

Delimiter (One byte): A percentage sign “%”

Function (Two bytes): function string

Status: “OK” or “FAIL”

For example: Set the IP Address by sending command

SSLI192.168.2.125(CR)(LF)

Echo string

%SLIOK(CR)(LF)

means the new IP address is successfully transferred to the module and the new setting will be effective after reboot the module.

To get IP address setting, you can send a command as follow

\$GLI

then the Echo string will be

%GLI192.168.2.125

ATTENTION



Always remember to reboot the module by send command

\$SYCR to use the new configuration

Appendix A

Tiny Encryption Algorithm

1. Example of TEA.C

/******

The Tiny Encryption Algorithm (TEA) by David Wheeler and Roger Needham of the Cambridge Computer Laboratory. Placed in the Public Domain by David Wheeler and Roger Needham.

**** ANSI C VERSION (New Variant) ****

Notes:TEA is a Feistel cipher with XOR and addition as the non-linear mixing functions.

Takes 64 bits of data in v[0] and v[1]. Returns 64 bits of data in w[0] and w[1].

Takes 128 bits of key in k[0] - k[3].

TEA can be operated in any of the modes of DES. Cipher Block Chaining is, for example, simple to implement. n is the number of iterations. 32 is ample, 16 is sufficient, as few as eight may be OK. The algorithm achieves good dispersion after six iterations. The iteration count can be made variable if required.

Note this is optimised for 32-bit CPUs with fast shift capabilities. It can very easily be ported to assembly language on most CPUs. delta is chosen to be the real part of (the golden ratio $\sqrt{5/4} - 1/2 \sim 0.618034$ multiplied by 2^{32}). This version has been amended to foil two weaknesses identified by David A. Wagner (daw@cs.berkeley.edu): 1) effective key length of old-variant TEA was 126 not 128 bits 2) a related key attack was possible

although impractical.

```
*****/
#include "xtea.h"
void encipher(unsigned long *const v,unsigned long *const w,
const unsigned long *const k)
{
    register unsigned long    y=v[0],z=v[1],sum=0,delta=0x9E3779B9,n=32;
    while(n-->0)
        { y+= (z<<4 ^ z>>5) + z ^ sum + k[sum&3];
          sum += delta;
          z+= (y<<4 ^ y>>5) + y ^ sum + k[sum>>11 & 3];
        }
    w[0]=y; w[1]=z;
}
void decipher(unsigned long *const v,unsigned long *const w,
const unsigned long *const k)
{
    register unsigned long    y=v[0],z=v[1],sum=0xC6EF3720,
                              delta=0x9E3779B9,n=32;
    /* sum = delta<<5, in general sum = delta * n */
    while(n-->0)
        {
            z-= (y<<4 ^ y>>5) + y ^ sum + k[sum>>11 & 3];
            sum -= delta;
            y-= (z<<4 ^ z>>5) + z ^ sum + k[sum&3];
        }
    w[0]=y; w[1]=z;
}
```

```
}
2. The TEA.H
#ifndef __XTEA_H__
#define __XTEA_H__
void encipher(unsigned long *const v,unsigned long *const w,
const unsigned long *const k);
void decipher(unsigned long *const v,unsigned long *const w,
const unsigned long *const k);
#endif
```