Aport-212PG

Programmable Device Server

User Guide

Version 1.0

Copyright © Artila Electronics Co., Ltd. All Rights Reserved.

Table of Contents

1. Introduction 1

1.1 Specification 1

1.2 Packing List 2

1.3 Optional Accessory 2

2. Layout 3

3. Pin Assignment and Definition 4

3.1 Power Connector 4

3.2 Serial Port Connector 4

3.3 LED Status 4

3.4 Factory Default Settings 4

4. Artila Utility Software 6

4.1 Install Manager Utility Software 6

4.2 Broadcast Search 6

4.3 Install Software Toolchain 7

4.4 Install Eclipse IDE 7

4.5 Start Your First Project 8

1. Introduction

Aport-212PG is a programmable serial to Ethernet gateway which includes Cortex-M3 CPU, 64KB SRAM and 512KB flash. Aport-212PG is designed for users who are looking for a tiny but mighty computing platform which has FreeRTOS and lwIP pre-installed. The tool chain, Sourcery CodeBench Lite can be downloaded from mentor website or you can also use Keil from ARM. A tiny web server is also available for web-based application such as ajax. A demo web page source code is available for user's reference.

The key features are as follow:

- NXP LPC1768 ARM Cortex-M3 100MHz
- 512KB on-chip flash, 64KB SRAM
- Two configurable RS-232/422/485 serial ports
- One 10/100Mbps Ethernet ports
- One serial console port
- Support lwIP and BSD socket library
- Support tiny Web server and AJAX application
- Windows configuration utility included
- Support Telnet and serial console command
- Toolchain: Sourcery CodeBench Lite (download from www.mentor.com)

1.1 Specification

- **System:**
    - CPU: NXP LPC1768 Cortex-M3 100MHz
- **Serial port:**
    - Port1: RS-232/422/485
    - Port2: RS-232/485
    - Baud rate: 1200 to 921600 bps
    - Flow control: None/Hardware/Xon_Xoff
    - Data bit: 5 to 8
    - Stop bit: 1 to 2
    - Protection: 15KV ESD
- **Ethernet:**
    - 10/100 Mbps, RJ45
    - Protection: 1500V Magnetic isolation
- **Serial console port:**
    - RS-232: 115200 baud rate, None flow control, 8 bits data, 1 stop bit
- **Power:** 9~40VDC power jack and terminal block
- **Dimension:** 108 x 78 x 25 mm (H x W x D)
- **Operating Temperature:** 0~70°C
- **Storage Temperature:** -20~85°C

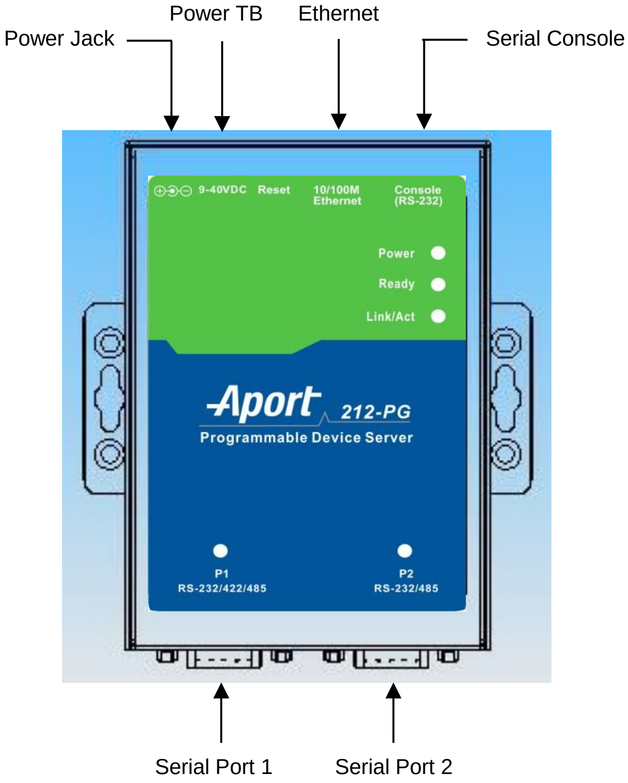
1.2 Packing List

- Aport-212PG programmable device server
- Software toolchain (download from Artila FTP)
- Manager Utility (download from Artila FTP)

1.3 Optional Accessory

- CB-RJ2CON-100 (91-RJCON-100): Console Cable (RJ45 to DB9 Female, 100cm)
- DK-35A (36-DK35A-000): DIN RAIL Mounting Kit
- PWR-12V-1A (31-62100-000): 110~240VAC to 12VDC 1A Power Adaptor

2. Layout


Power Jack
Power TB
Ethernet
Serial Console
9-40VDC
Reset
10/100M Ethernet
Console (RS-232)
Power
Ready
Link/Act
Aport 212-PG
Programmable Device Server
P1 RS-232/422/485
P2 RS-232/485
Serial Port 1
Serial Port 2

3. Pin Assignment and Definition

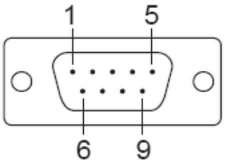
3.1 Power Connector

Connecting 9~40VDC power line with the Aport-212PG terminal block or the power jack. If the power is properly supplied, the Power LED will keep solid yellow color.

3.2 Serial Port Connector

Serial Port uses a Male DB9 connector and it includes RS-232, RS-422 or RS485 signal and pin assignments are described as follow:

Pin No.	RS-232	RS-422	RS-485
1	DCD*	TXD-	-
2	RXD	TXD+	-
3	TXD	RXD+	DATA+
4	DTR*	RXD-	DATA-
5	GND	GND	GND
6	DSR*	-	-
7	RTS	-	-
8	CTS	-	-
9	-	-	-



3.3 LED Status

The LED provides the Aport-212PG operation information. The LED status is described as follow:

- **Power LED:** Power LED keeps ON if power (+9VDC to +40VDC) is correctly input to Aport-212PG.
- **Ready LED:** Ready LED keeps ON when Aport-212PG firmware is ready for operation.
- **Link/Act LED:** Link and Activity LED will turn ON when the Ethernet cable is connected. When there is network data traffic, this LED will flash.
- **RX/TX LED:** The RX/TX LED is a dual color LED that indicates the serial data traffic. The Yellow LED stands for receiving data and Green LED means transmitting data.

3.4 Factory Default Settings

IP Address: 192.168.2.127

Netmask: 255.255.255.0

Serial Port: RS-232

Baud rate: 115200

Data: No parity, 8 bits, 1 stop bit

Flow control: None

Serial Console port: RS-232

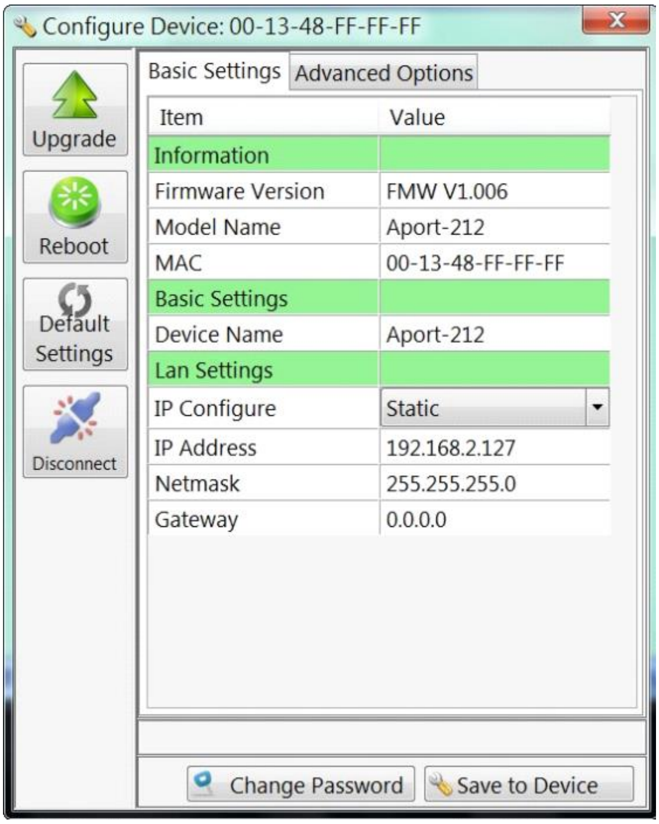
Baud rate: 115200

Data: No parity, 8 bits, 1 stop bit

Flow control: None

Web console: http://192.168.2.127:5003

Telnet console: telnet 192.168.2.127 5001


Configure Device: 00-13-48-FF-FF-FF
Upgrade
Reboot
Default Settings
Disconnect
Basic Settings
Advanced Options
Change Password
Save to Device

Item	Value
Information	
Firmware Version	FMW V1.006
Model Name	Aport-212
MAC	00-13-48-FF-FF-FF
Basic Settings	
Device Name	Aport-212
Lan Settings	
IP Configure	Static
IP Address	192.168.2.127
Netmask	255.255.255.0
Gateway	0.0.0.0

4. Artila Utility Software

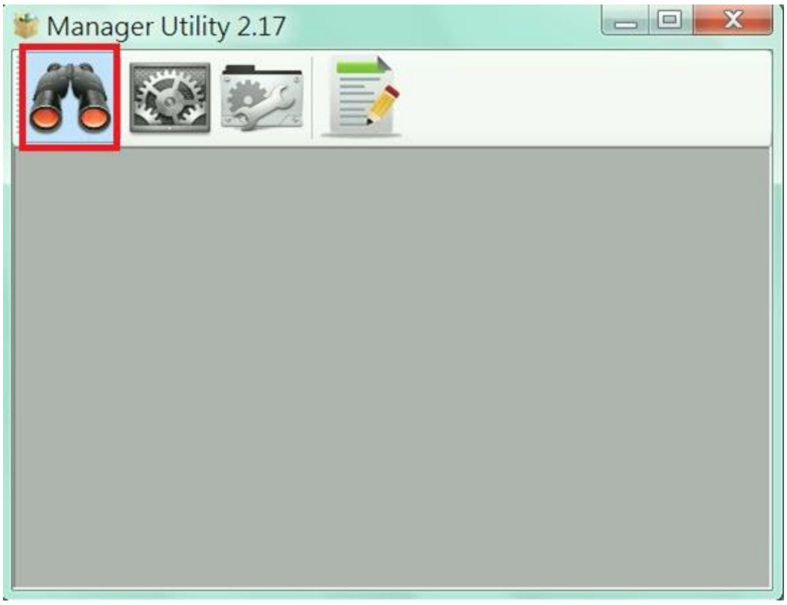
4.1 Install Manager Utility Software

You can find many useful software utilities from Artila FTP. You need to install Manager Utility first in order to configure the Aport-212PG. To install the Manager Utility, please find the ***ManagerUtilitysetup.exe*** as shown following:

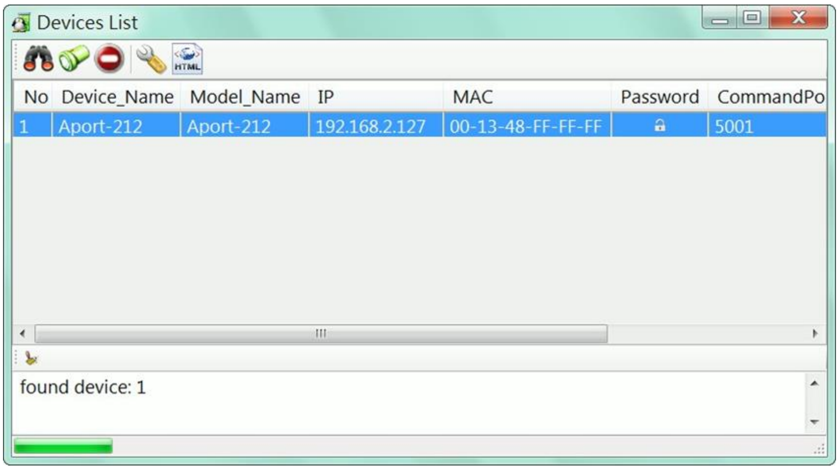
ManagerUtilitySetup
ManagerUtility Setup
Artila Electronics Co., Ltd

4.2 Broadcast Search

Once start Manager utility, you can click telescope icon to search the Aport-212PG in the network.



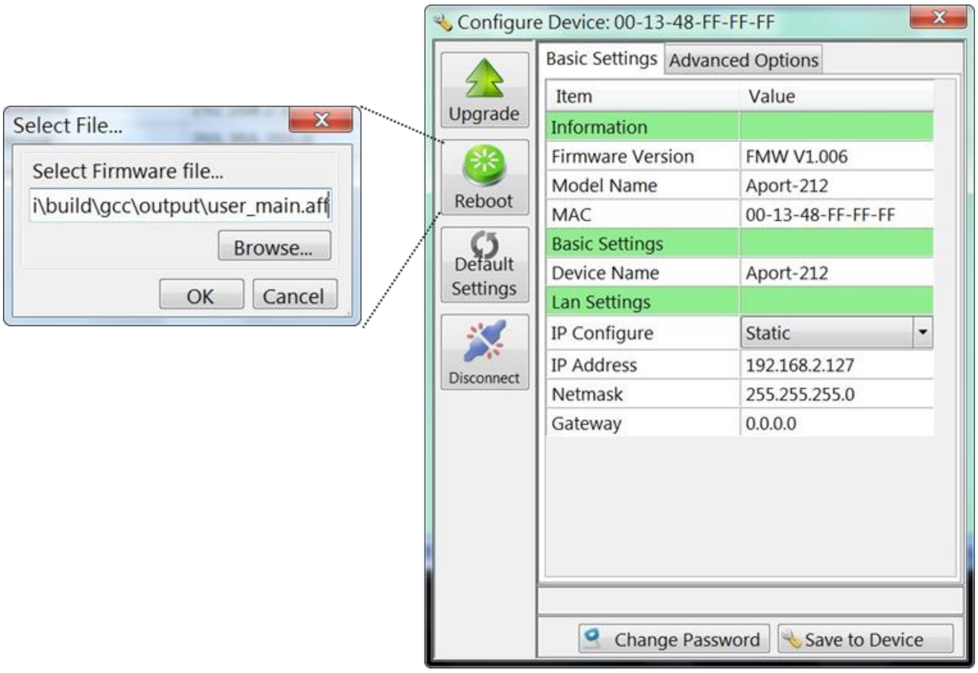
Click the device to configure its settings.



No	Device_Name	Model_Name	IP	MAC	Password	CommandPo
1	Aport-212	Aport-212	192.168.2.127	00-13-48-FF-FF-FF	🔒	5001

found device: 1

Click the upgrade to upload the new firmware ***user_main.aff.***



Select File...
Browse...
OK
Cancel
Configure Device: 00-13-48-FF-FF-FF
Upgrade
Reboot
Default Settings
Disconnect
Basic Settings
Advanced Options
Change Password
Save to Device

Select Firmware file...

i\build\gcc\output\user_main.aff

Item	Value
Information	
Firmware Version	FMW V1.006
Model Name	Aport-212
MAC	00-13-48-FF-FF-FF
Basic Settings	
Device Name	Aport-212
Lan Settings	
IP Configure	Static
IP Address	192.168.2.127
Netmask	255.255.255.0
Gateway	0.0.0.0

4.3 Install Software Toolchain

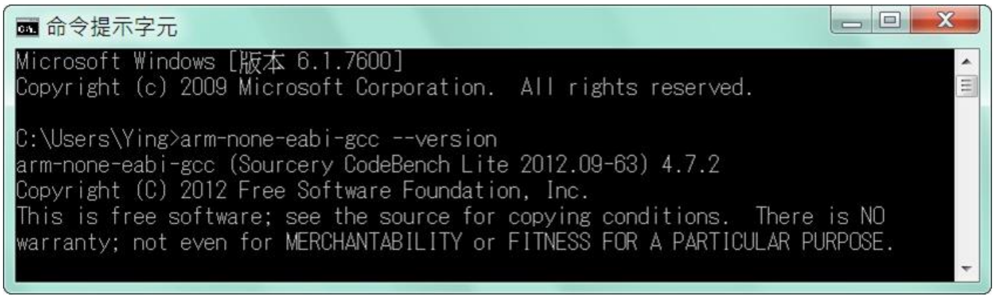
The toolchain, Sourcery CodeBench Lite ARM EABI Release is available at:

http://www.mentor.com/embedded-software/sourcery-tools/sourcery-codebench/editions/lite-edition/

Configure the environment to add the path of the toolchain. After installing toolchain, a new path will be added to Windows Environment i.e.:

Sourcery_CodeBench_Lite_for_ARM_EABI\bin

Restart the computer to make the new environment effective. After installation, you can test toolchain as follow:



```
Microsoft Windows [版本 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\Ying>arm-none-eabi-gcc --version
arm-none-eabi-gcc (Sourcery CodeBench Lite 2012.09-63) 4.7.2
Copyright (C) 2012 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

4.4 Install Eclipse IDE

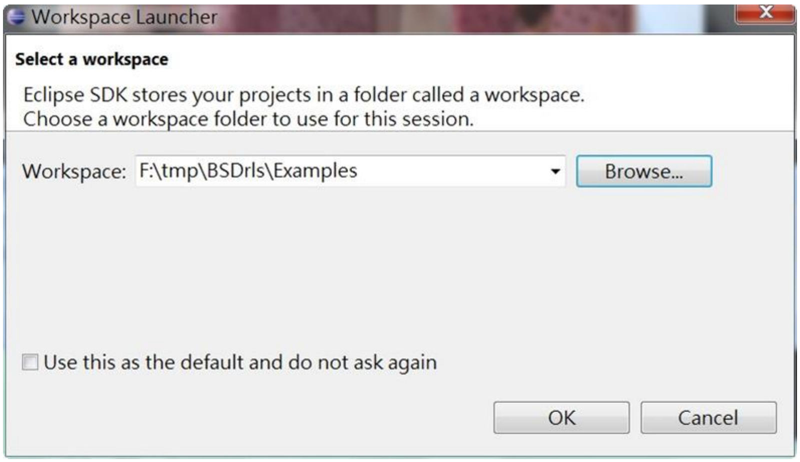
If you are interesting in using IDE to develop your program, the eclipse IDE is available at:

http://www.eclipse.org/downloads/

And choose C/C++ compiler option.

4.5 Start Your First Project

Run eclipse and select a workspace: ***BSDrls\Examples***. You can find the path of the example program on Artila FTP with path: ***BSDrls\Examples.***



Workspace Launcher

Select a workspace

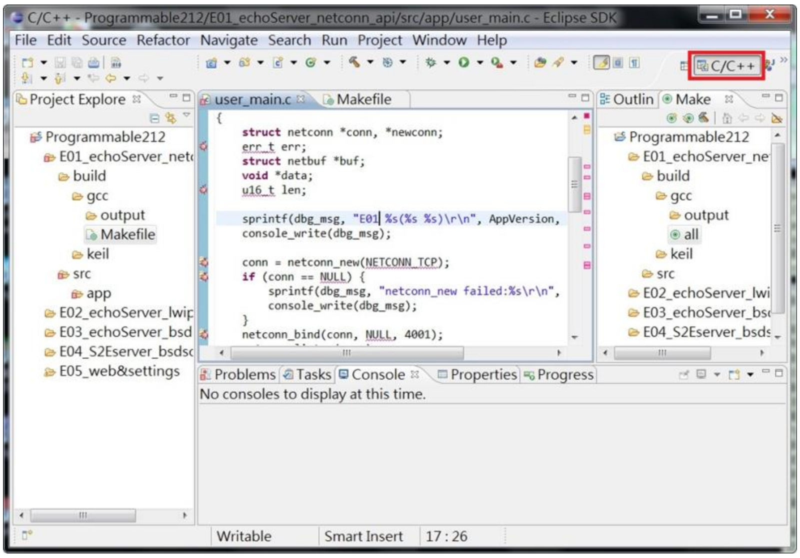
Eclipse SDK stores your projects in a folder called a workspace.
Choose a workspace folder to use for this session.

Workspace: F:\tmp\BSDrls\Examples

☐ Use this as the default and do not ask again

Browse...
OK
Cancel

Choose C/C++ in the Workbench.



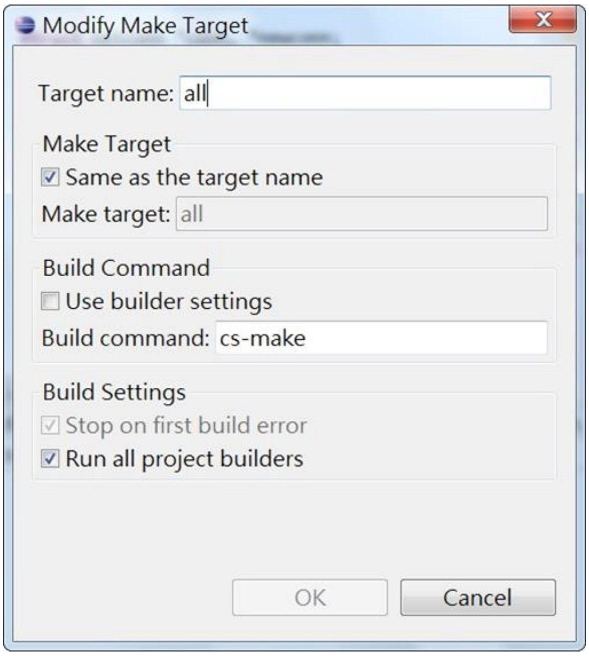
C/C++ - Programmable212/E01_echoServer_netconn_api/src/app/user_main.c - Eclipse SDK

```
{
    struct netconn *conn, *newconn;
    err_t err;
    struct netbuf *buf;
    void *data;
    u16_t len;

    sprintf(dbg_msg, "E01 %s(%s %s)\r\n", AppVersion,
    console_write(dbg_msg);

    conn = netconn_new(NETCONN_TCP);
    if (conn == NULL) {
        sprintf(dbg_msg, "netconn_new failed:%s\r\n",
        console_write(dbg_msg);
    }
    netconn_bind(conn, NULL, 4001);
```

No consoles to display at this time.

File Edit Source Refactor Navigate Search Run Project Window Help
C/C++
Project Explore
Programmable212
E01_echoServer_netc
build
gcc
output
Makefile
keil
src
app
E02_echoServer_lwip
E03_echoServer_bsd
E04_S2Eserver_bsdsc
E05_web&settings
user_main.c
Makefile
Outlin
Make
Programmable212
E01_echoServer_ne
build
gcc
output
all
keil
src
E02_echoServer_lwi
E03_echoServer_bs
E04_S2Eserver_bsds
Problems
Tasks
Console
Properties
Progress
Writable
Smart Insert
17 : 26

Modify the make file to compile the program as follow:


Modify Make Target
OK
Cancel

Target name: all

Make Target

☑ Same as the target name

Make target: all

Build Command

☐ Use builder settings

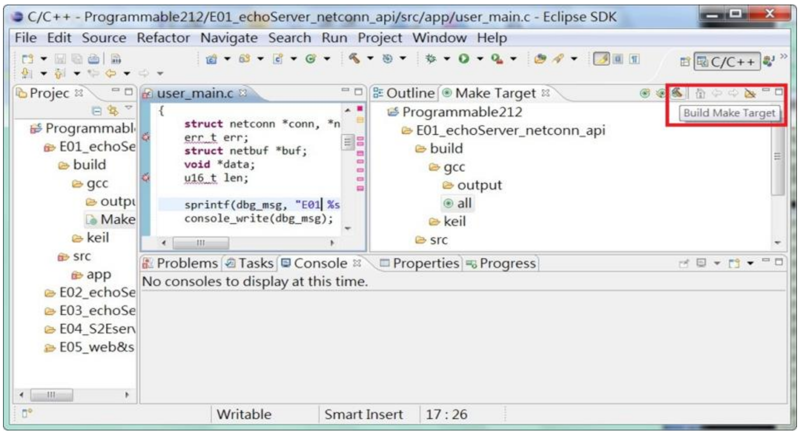
Build command: cs-make

Build Settings

☑ Stop on first build error

☑ Run all project builders

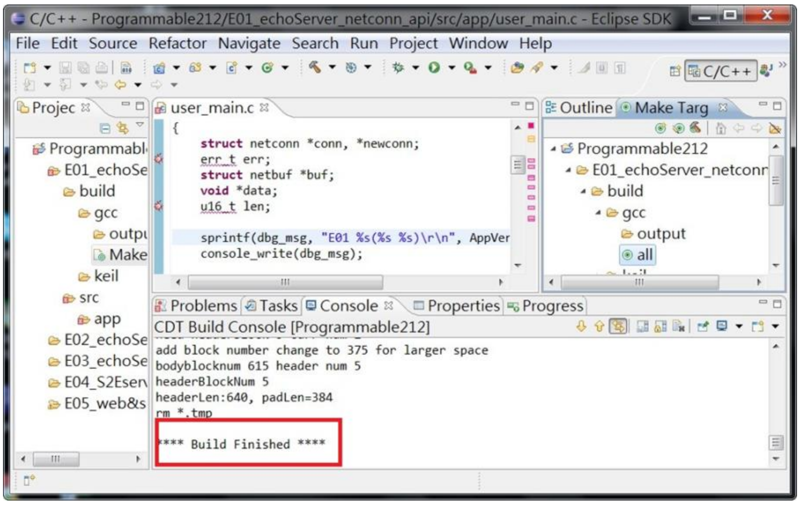
Use make file to build target.


C/C++ - Programmable212/E01_echoServer_netconn_api/src/app/user_main.c - Eclipse SDK
File Edit Source Refactor Navigate Search Run Project Window Help
C/C++
Projec
Programmabl
E01_echoSe
build
gcc
outpu
Make
keil
src
app
E02_echoSe
E03_echoSe
E04_S2Esen
E05_web&s
user_main.c
Outline
Make Target
Programmable212
E01_echoServer_netconn_api
build
gcc
output
all
keil
src
Build Make Target
Problems
Tasks
Console
Properties
Progress
Writable
Smart Insert
17 : 26

```
{
    struct netconn *conn, *n
    err_t err;
    struct netbuf *buf;
    void *data;
    u16_t len;

    sprintf(dbg_msg, "E01 %s
    console_write(dbg_msg);
```

No consoles to display at this time.

Once project is built, you will find the target execution file ***user_main.aff*** is generated and available at: ***E01_echoServer_netconn_api\build\gcc\output***


C/C++ - Programmable212/E01_echoServer_netconn_api/src/app/user_main.c - Eclipse SDK
File Edit Source Refactor Navigate Search Run Project Window Help
C/C++
Projec
Programmabl
E01_echoSe
build
gcc
outpu
Make
keil
src
app
E02_echoSe
E03_echoSe
E04_S2Esen
E05_web&s
user_main.c
Outline
Make Targ
Programmable212
E01_echoServer_netconn
build
gcc
output
all
Problems
Tasks
Console
Properties
Progress

```
{
    struct netconn *conn, *newconn;
    err_t err;
    struct netbuf *buf;
    void *data;
    u16_t len;

    sprintf(dbg_msg, "E01 %s(%s %s)\r\n", AppVer
    console_write(dbg_msg);
```

CDT Build Console [Programmable212]

```
add block number change to 375 for larger space
bodyblocknum 615 header num 5
headerBlockNum 5
headerLen:640, padLen=384
rm *.tmp

**** Build Finished ****
```