# UNO-1251G Embedded Linux User Manual V1.02



Enabling an Intelligent Planet

## CONTENTS

1. About This Manual	2
2. UNO-1251G Features Overview	3
3. UNO-1251G Embedded Linux	4
4. Installation	5
4.1 Micro SD Card Installation Guide	5
4.2 Boot the Linux	5
5. Accessing the Console of UNO-1251G	6
5.1 Accessing the Serial Debug Console	6
5.2 Remote Console by Telnet	6
6. Switch Init Level	7
6.1 Runlevel Arrangements	7
6.2 Runlevel Switch	7
7. Linux System Configuration and Use	8
7.1 Service Configuration	8
7.1.1 Starting/Stopping the Services	8
7.1.2 Starting Service with System Boot Up	8
7.2 Network configuration	9
7.2.1 Overview	9
7.2.2 Configuration via console	9
7.3 GRE Tunnel Configuration	10
7.3.1 GRE Tunnel Configuration On UNO-1251G	11
7.3.2 GRE Tunnel Configuration On HOST	12
7.4 Ipsec Configuration	14
7.4.1 Ipsec Configuration On UNO-1251G	14
7.4.2 Ipsec Configuration On HOST	15
7.5 Support MC8090 3G Modules	16
7.5.1 Controlling MC8090 3G Modules	17
7.5.2 PPP Dial-up Configuration	17
7.6 Support 968EMC0061(MC7304) 4G Modules	20
7.6.1 Controlling MC7304 4G Modules	20
7.6.2 PPP Dial-up Configuration	20
7.7 Support EWM-C109F601E/EWM-C109F6G1E (LISA-U200) 3G Modules	23
7.7.1 Controlling LISA-U200 3G Modules	24
7.7.2 PPP Dial-up Configuration	24
7.8 Support EWM-W150H02E(RT5390) Wireless Modules	27
7.8.1 Scan the nearby available AP	27
7.8.2 Using password-less wifi connection	28
7.8.3 Using WPA wifi connection	28
8. Software Management	30
8.1 OPKG Package Manager	30
8.2 Installation a New Software	30

8.3 More about OPKG	
8.4 Example	
9. Development	32
9.1 Advantech AM3517 Development Kit Overview	32
9.2 Installation	
9.3 Development of C/C++ Programs	
9.4 Development of GUI Programs with QT 4.7.2 and QT 4.8.3 Library	32
9.5 Deployment of the Application	
9.5.1 Starting Your Custom Application on System Boot up	33
9.5.2 Adding a New Application/Directory to Matrix	
9.6 Development of Kernel Driver modules	33
9.7 Source of Example	34
9.7.1 Serial Port Programming	34
9.7.2 OLED Programming	34
10. Appendix	
10.1 U-boot Configuration	
10.1.1 Enter U-boot mode	
10.1.2 Loading Kernel from micro SD	
10.1.3 Loading Kernel from Tftpd	
10.1.4 Mounting File system	36
10.2 APIs of ADVOLED Driver	37
10.3 RTC Battery Automatic detection	
10.4 Switch of Serial Port	
10.5 VNC remote login	
10.6 Get current IP address	40



## 1. About This Manual

Thank you for using Advantech products. This manual is a supplementation to UNO-1251G user manual, and it emphasizes on Linux for UNO-1251G. For more detail information about UNO-1251G hardware, jumper setting, serial port connections and so on, please refer to the UNO-1251G user manual.



Advantech UNO-1251G Manual

## 2. UNO-1251G Features Overview

- > CPU: TI Coretex A8 AM3517 600MHz
- ➢ RAM: 256MB/512MB
- Storage: 1 x micro SD 1GB (bootloader and image)
- 1 x micro SD card slot (External storage)
- ➢ USB: 1 x USB 2.0 port
- LAN: 2 x 10/100 Mbps RJ45 Ethernet ports
- Display: Active Area 0.91" 128 x 32 Pixels OLED
- > Expansion: 1 x mPCle slot with USB single
- Serial Ports: UNO-1251G 1 x RS485, 1 x RS422/485, 1 x RS232 serial port
- ➢ Other: 1 x CAN Bus





## 3. UNO-1251G Embedded Linux

- 1. Boot loader: u-boot-2011.09
- > 2. Kernel version: Linux 2.6.37
- ➢ 3. Support filesystem list: ext2, ext3, ext4, fat32, aufs, unionfs, squashfs
- 4. C library: glibc version: 2.12.2
- 5. QT library: QT/Embedded 4.7.2, QT/Embedded 4.8.3
- ➢ 6. GUI-based application launcher
- > 7. Example application with available source code
- 8. Device:

Serial Ports (RS232/RS422/RS485) Ethernet RTC micro SD slot mPCIe with USB single slot USB CAN OLED

## 4. Installation

Perform the following installation steps recommended in **Ubuntu14.04 and the previous** version.

### 4.1 Micro SD Card Installation Guide

This guide is for those looking to create a three-partition card, booting from a FAT partition that can be read by the OMAP3 ROM bootloader and Linux/Windows, then utilizing an ext3 partition for the Linux root file system, and the last partition is for user's data.

First, take out the SD card from the machine. Open the SD cover of the device, you will see the SD card as shown below:

#### Note. The SD size required is minimum 1024 MB.

Connect your **SD** card to a Linux machine. Open a terminal console on your Linux machine and run the following command:

hostPC\$ sudo fdisk -I

Check the connection for your **SD** card. Look for /dev/sd\*\*\* device which matches the memory card size. e.g /dev/sdb/ (Disk /dev/sdb: 503 MB, 503316480 bytes)

Download the script "mkbootcard.sh" card to your Linux machine (You can also get this script from the package of embedded Linux for UNO-1251G, please refer to section 4.2.1).

Grant execution access to the script using the following command:

hostPC\$ sudo chmod u+x mkbootcard.sh

To create the partitioning, run the following command:

hostPC\$ sudo ./mkbootcard.sh /dev/sdb

Then, your SD card has 2 partitions: boot, rootfs. Sdb1 is boot partition, sdb2 is rootfs partition. The shell will auto install bootloader & kernel file "u-boot.img" "uImage" and "MLO" to the partition labeled "boot" on the SD card. And it will auto decompress install the rootfs to the partition labeled "rootfs" on the SD card.

### 4.2 Boot the Linux

After the completion of the installation steps above, insert the SD card back into the device marked as "SD 1" or "SD\_BOOT", and install the SD cover. Then you can boot your device to Linux.

## 5. Accessing the Console of UNO-1251G

### 5.1 Accessing the Serial Debug Console

Use "**minicom**" in the Linux or "**HyperTerminal**" in the Windows to connect "**Debug Port**" of UNO-1251G to the PC.Connect COM as the below figure shows. Make sure the Baud Rate is 115200 bps.

## 5.2 Remote Console by Telnet

If your UNO-1251G is connected to the LAN, you can access the console by telnet. It avoid to connect the debug serial cable and do such complicated configuration.

Run the following command on your host PC:

hostPC\$ sudo telnet TARGET\_SYSTEM\_IP

You can find the target's IP address by using "ifconfig" command.



Enabling an Intelligent Plan

## 6. Switch Init Level

### 6.1 Runlevel Arrangements

Advantech Embedded linux has 6 runlevels. Of which runlevels 2-5 are set to different software operating environment mode. At present, only runlevel 5 support VNC server ( xfce desktop ).

Runlevel 2: Automatically start user-defined qt-framebuffer application.

Runlevel 3: Start Matrix GUI App List (Not support remote extension displays)
 Runlevel 4: Automatically start user defined X11 application.
 Runlevel 5: Start xfce4 desktop (support VNC server)



System is boot from runlevel 5 by default. And you can set the runlevel which you want by follow the section 6.2.

### 6.2 Runlevel Switch

You can set different runlevel to make your device boot to a different mode mentioned in section 6.1. Please follow the following steps to change the system runlevel:

**Step 1**: Access the console of device (refer section chapter 5).

**Step 2**: Run the following command on UNO-1251G:

advantech# cp /etc/inittab-rl/inittab.rl**X** /etc/inittab

Note. X should be one number of 2-5 which you want to change the runlevel to.

Step 3: Reboot the device, and the system will reboot to the mode you set.

Enabling an Intelligent Plane

## 7. Linux System Configuration and Use

## 7.1 Service Configuration

UNO-1251G has built seven common network services: tftp service, ftp service, ssh service, telnet service, http service, (s)ntp service and snmp service. And the IPv6 protocol can be supported by ftp, ssh, and telnet. Also, UNO-1251G has build three network protocols SMTP, POP3, IMAP into filesystem. You can test these three network protocols SMTP, POP3, IMAP by using claws-mail client receive/send email in runlevel 5.

#### 7.1.1 Starting/Stopping the Services

To start a service, run the following command on UNO-1251G:

advantech# /etc/init.d/SERVICE\_NAME start

Note. SERVICE\_NAME should be one of the following list:

xinetd - TFTP server agent vsftpd - FTP server sshd - SSH server telnetd - TELNET server lighttpd - HTTP server ntpd-(S)NTP server snmpd-SNMP server

Similar, use the following command to stop the service:

advantech# /etc/init.d/SERVICE\_NAME stop

If you want to start/stop the tftp service, please modify /etc/xinetd.d/tftp, the changes will take effect after reboot.

/etc/xine	etd.d/tftp	
# enable	tftp server by default on boo	ot
	disable	<i>= no</i>
# disab	le tftp server by default on b	pot
#	disable	= yes

The username/password for these servers are listed below:

<b>/</b> sftpd	username:root	password:NULL
Telnet	username:root	password:NULL
Sshd	username:root	password:NULL

The default directory of tftp server is /tftpboot, ftp server is /home/ftp.

#### 7.1.2 Starting Service with System Boot Up

If you want the service to start when you boot up the UNO-1251G, you need to do some configuration. Run the following command on UNO-1251G:



nabling an Intelligent Plan

advantech# cp /etc/init.d/SERVICE\_NAME /etc/rcX.d/S10SERVICE\_NAME

Note. X should be one number of 2-5 which the current runlevel is. (refers to Chapter 6)

For example, if you want the telnet service to start by default on the runlevel 5(xfce4 desktop), run the following command:

advantech# cp /etc/init.d/telnetd /etc/rc5.d/S10telnetd

When you want to cancel the default startup, just remove the S10\* file. The services telnetd, lighttpd and sshd will start when the device boots up by default.

### 7.2 Network configuration

#### 7.2.1 Overview

You can get an IP address via dhcp, also you can configure a static IP address for UNO-1251G.

#### 7.2.2 Configuration via console

The UNO-1251GL does not have VGA, so no Advantech NIC Configuration utility on it, but you can access the console by telnet. Run the following command on UNO-1251G:

DHCP:

Start the DHCP process

advantech# /etc/init.d/dhcpc eth0 start

advantech# /etc/init.d/dhcpc eth1 start

If you want to reserve the setting after rebooting the device, set as below

advantech# cp /usr/src/example/netcfg/temp.dhcp.netcfg.eth0 /etc/adv.d/netcfg.eth0

advantech# cp /usr/src/example/netcfg/temp.dhcp.netcfg.eth1 /etc/adv.d/netcfg.eth1

STATIC:

Stop the DHCP process

advantech# /etc/init.d/dhcpc eth0 stop

advantech# /etc/init.d/dhcpc eth1 stop

Set the static IP as below

advantech# /sbin/ifconfig eth0 172.21.73.122 netmask 255.255.255.0

advantech# /sbin/route delete default

advantech# echo 'nameserver 172.21.128.10' > /etc/resolv.conf

advantech# echo '/sbin/route add default gw 172.21.73.253 dev eth0' > /etc/gwserver/gateway.eth0

advantech# chmod a+x /etc/gwserver/gateway.eth0

advantech# /etc/gwserver/gateway.eth0



advantech# /sbin/ifconfig eth1 172.21.73.121 netmask 255.255.255.0

advantech# /sbin/route delete default

advantech# echo 'nameserver 172.21.128.10' > /etc/resolv.conf

advantech# echo '/sbin/route add default gw 172.21.73.253 dev eth1' > /etc/gwserver/gateway.eth1

advantech# chmod a+x /etc/gwserver/gateway.eth1

advantech# /etc/gwserver/gateway.eth1

If you want to reserve the setting after rebooting the device, set as below :

Firstly , open and modify /usr/src/example/netcfg/temp.static.netcfg.eth0 or

/usr/src/example/netcfg/temp.static.netcfg.eth1.

For example, set ip ( eth0 192.168.10.100 ), netmask ( netmask 255.255.255.0 ), nameserver ( nameserver 192.168.128.10 ) and gateway ( gw 192.168.10.253 ) for eth0 :

```
#!/bin/sh
```

.....

```
if [ -f "/var/tmp/eth0.link" ];then
```

ip addr flush eth0

/sbin/ifconfig eth0 192.168.10.100 netmask 255.255.255.0

funcWaitShellReady

.....

/sbin/route delete default

echo 'nameserver 192.168.128.10' > /etc/resolv.conf

echo 'route add default gw 192.168.10.253 dev eth0' > /etc/gwserver/gateway.eth0

chmod a+x /etc/gwserver/gateway.eth0

/etc/gwserver/gateway.eth0

```
fi
```

fi

#### Then:

advantech# cp /usr/src/example/netcfg/temp.static.netcfg.eth0 /etc/adv.d/netcfg.eth0

advantech# cp /usr/src/example/netcfg/temp.static.netcfg.eth1 /etc/adv.d/netcfg.eth1

### 7.3 GRE Tunnel Configuration

UNO-1251G fully supported GRE tunnel. Before setup a GRE tunnel, you must do some settings at



two sides.

I will provide an example that how to setup a GRE tunnel between UNO-1251G and HOST (Fedora 17). UNO-1251G: 172.21.73.116

HOST: 172.21.73.79

#### 7.3.1 GRE Tunnel Configuration On UNO-1251G

1. To create a GRE-type tunnel on UNO-1251G, run the following commands.

advantech# ip tunnel add tun1 mode gre remote 172.21.73.79 local 172.21.73.116 ttl 255

advantech# ip link set tun1 up

advantech# ip addr add 10.0.1.1 dev tun1

advantech# ip route add 10.0.2.0/24 dev tun1

2. When you want to create a GRE Tunnel with system booting up on UNO-1251G. You should create gre.sh and write above commands into it, then run gre.sh with system booting up.

advantech# vi /etc/init.d/gre.sh

gre.sh:

#!/bin/sh

ip tunnel add tun1 mode gre remote 172.21.73.79 local 172.21.73.116 ttl 255

ip link set tun1 up

ip addr add 10.0.1.1 dev tun1

ip route add 10.0.2.0/24 dev tun1

Run the following commands. The gre.sh will run with system booting up.

advantech# chmod +x /etc/init.d/gre.sh

advantech# In -s /etc/init.d/gre.sh /etc/rcS.d/S99Gre

In the above, we create a GRE tunnel device called tun1 on UNO-1251G, and set its remote address to 172.21.73.79. Tunneling packets will be originating from 172.21.73.116 (local IP address), and their TTL field will be set to 255. The tunnel device is assigned IP address 10.0.1.1 with netmask 255.255.255.255, also we add ip route 10.0.2.0/24 into tunnel device tun1.

3. Now you can verify GRE Tunnel configuration on UNO-1251G. Run *"ifconfig"* command, then you can see GRE Tunnel device tun1.

advantech# ifconfig ...... tun1 Link encap:UNSPEC HWaddr AC-15-49-74-00-00-00-00-00-00-00-00-00-00 inet addr:10.0.1.1 P-t-P:10.0.1.1 Mask:255.255.255 UP POINTOPOINT RUNNING NOARP MTU:1476 Metric:1



RX packets:0 errors:0 dropped:0 overruns:0 frame:0

TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

Run *"ip addr show"* command on UNO-1251G. You can see the information of GRE Tunnel device tun1.

advantech# ip addr show

.....

10: tun1: <POINTOPOINT,NOARP,UP,LOWER\_UP> mtu 1476 qdisc noqueue state UNKNOWN

link/gre 172.21.73.116 peer 172.21.73.79

inet 10.0.1.1/32 scope global tun1

#### 7.3.2 GRE Tunnel Configuration On HOST

1. To create a GRE tunnel on HOST, run the following commands.

hostPC\$ sudo ip tunnel add tun1 mode gre remote 172.21.73.116 local 172.21.73.79 ttl 255

hostPC\$ sudo ip link set tun1 up

hostPC\$ sudo ip addr add 10.0.2.1 dev tun1

hostPC\$ sudo ip route add 10.0.1.0/24 dev tun1

2.When you want to create a GRE Tunnel with system booting up on HOST. You should create gre.sh and write above commands into it. Then run gre.sh with system booting up.

hostPC\$ sudo vi /etc/init.d/gre.sh

gre.sh:

#!/bin/sh

*ip tunnel add tun1 mode gre remote 172.21.73.116 local 172.21.73.79 ttl 255* 

ip link set tun1 up

ip addr add 10.0.2.1 dev tun1

ip route add 10.0.1.0/24 dev tun1

Run gre.sh with system booting up.

hostPC\$ sudo chmod +x /etc/init.d/gre.sh

hostPC\$ sudo echo "/etc/init.d/gre.sh" >> /etc/rc.d/rc.local

In the above, we create a GRE tunnel device called tun1 on HOST, and set its remote address to 172.21.73.116. Tunneling packets will be originating from 172.21.73.79 (local IP address), and their

TTL field will be set to 255. The tunnel device is assigned IP address 10.0.2.1 with netmask 255.255.255.255. Also, we add ip route 10.0.1.0/24 into tunnel device tun1.

3. Now you can verify GRE Tunnel configuration on HOST. Run *"ifconfig"* command, then you can see GRE Tunnel device tun1.

hostPC\$ sudo ifconfig

.....

*tun1*: *flags=209<UP,POINTOPOINT,RUNNING,NOARP> mtu* 1476

inet 10.0.2.1 netmask 255.255.255.255 destination 10.0.2.1 inet6 fe80::5efe:ac15:494f prefixlen 64 scopeid 0x20<link> unspec AC-15-49-4F-48-B7-00-10-00-00-00-00-00-00-00 txqueuelen 0 (UNSPEC) RX packets 0 bytes 0 (0.0 B)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 3 bytes 168 (168.0 B)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

Run "ip addr show" command on HOST. You can see the information of GRE Tunnel device tun1.

*hostPC\$ sudo ip addr show* 

.....

7: tun1: <POINTOPOINT,NOARP,UP,LOWER\_UP> mtu 1476 qdisc noqueue state UNKNOWN

link/gre 172.21.73.79 peer 172.21.73.116

inet 10.0.2.1/32 scope global tun1

inet6 fe80::5efe:ac15:494f/64 scope link

valid\_lft forever preferred\_lft forever

At this point, a GRE tunnel should be established between UNO-1251G and HOST.

Run the following commands to test GRE tunnel between UNO-1251G and HOST.

hostPC\$ sudo ping 10.0.1.1

PING 10.0.1.1 (10.0.1.1) 56(84) bytes of data.

64 bytes from 10.0.1.1: icmp\_req=1 ttl=64 time=1.82 ms

64 bytes from 10.0.1.1: icmp\_req=2 ttl=64 time=0.552 ms

If you want to tear down the GRE tunnel, run the following commands on UNO-1251G and HOST.

# ip link set tun1 down

# ip tunnel del tun1

## 7.4 Ipsec Configuration

UNO-1251G fully supported Ipsec. Before setup a Ipsec channel, you must do some settings at two sides.

I will provide an example that how to setup a Ipsec channel between UNO-1251G and HOST (Fedora 17). UNO-1251G: 172.21.73.116

HOST: 172.21.73.79

#### 7.4.1 Ipsec Configuration On UNO-1251G

1. Open and modify /etc/setkey.conf.

#!/usr/sbin/setkey -f

# Configuration for 172.21.73.116

# Flush the SAD and SPD

flush;

spdflush;

# Attention: Use this keys only for testing purposes!

# Generate your own keys!

# AH SAs using 128 bit long keys

add 172.21.73.116 172.21.73.79 ah 0x200 - A hmac-md5

0xc0291ff014dccdd03874d9e8e4cdf3e6;

add 172.21.73.79 172.21.73.116 ah 0x300 - A hmac-md5

0x96358c90783bbfa3d7b196ceabe0536b;

# ESP SAs using 192 bit long keys (168 + 24 parity)

add 172.21.73.116 172.21.73.79 esp 0x201 -E 3des-cbc

0x7aeaca3f87d060a12f4a4487d5a5c3355920fae69a96c831;

add 172.21.73.79 172.21.73.116 esp 0x301 -E 3des-cbc

Oxf6ddb555acfd9d77b03ea3843f2653255afe8eb5573965df;

# Security policies

spdadd 172.21.73.116 172.21.73.79 any -P out ipsec

esp/transport//require

ah/transport//require;

spdadd 172.21.73.79 172.21.73.116 any -P in ipsec



esp/transport//require

ah/transport//require;

2. Run the following command to update SDD and SDA:

advantech# setkey -f /etc/setkey.conf

#### 7.4.2 Ipsec Configuration On HOST

1. Run the following command to install ipsec-tools.

hostPC\$ sudo yum install ipsec-tools

2. Open and modify /etc/setkey.conf. If there isn't etc/setkey.conf file, you can create it.

#!/usr/sbin/setkey -f

# Configuration for 172.21.73.79

# Flush the SAD and SPD

flush;

spdflush;

# Attention: Use this keys only for testing purposes!

# Generate your own keys!

# AH SAs using 128 bit long keys

add 172.21.73.116 172.21.73.79 ah 0x200 - A hmac-md5

0xc0291ff014dccdd03874d9e8e4cdf3e6;

add 172.21.73.79 172.21.73.116 ah 0x300 - A hmac-md5

0x96358c90783bbfa3d7b196ceabe0536b;

# ESP SAs using 192 bit long keys (168 + 24 parity)

add 172.21.73.116 172.21.73.79 esp 0x201 -E 3des-cbc

0x7aeaca3f87d060a12f4a4487d5a5c3355920fae69a96c831;

add 172.21.73.79 172.21.73.116 esp 0x301 -E 3des-cbc

0xf6ddb555acfd9d77b03ea3843f2653255afe8eb5573965df;

# Security policies

spdadd 172.21.73.116 172.21.73.79 any -P in ipsec

esp/transport//require

ah/transport//require;

spdadd 172.21.73.79 172.21.73.116 any -P out ipsec



esp/transport//require

ah/transport//require;

3. Run the following command to update SDD and SDA:

hostPC\$ sudo setkey -f /etc/setkey.conf

The successful load can be tested by displaying the SAD and the SPD, Run the following command and test it on UNO-1251G and HOST.

# setkey -D

# setkey -DP

After that, all network data between UNO-1251G(172.21.73.116) and HOST(172.21.73.79) will be encrypted.

Run the following command on host, ping from one peer to the other the traffic will be encrypted.

hostPC\$ sudo ping 172.21.73.116

Run *"tcpdump"* command on HOST that will show the following packets: Note: 172.21.73.116 is the IP of UNO-1251G. Also, you can use wireshark to capture the following packets.

*hostPC\$ sudo tcpdump -i p1p1 -n -vv | grep 172.21.73.116* 

tcpdump: listening on p1p1, link-type EN10MB (Ethernet), capture size 65535 bytes

172.21.73.79 > 172.21.73.116: AH(spi=0x00000300,sumlen=16,seq=0x12):

ESP(spi=0x00000301,seq=0x12), length 88

172.21.73.116 > 172.21.73.79: AH(spi=0x00000200,sumlen=16,seq=0x12):

ESP(spi=0x00000201,seq=0x12), length 88

172.21.73.79 > 172.21.73.116: AH(spi=0x00000300,sumlen=16,seq=0x13):

ESP(spi=0x00000301,seq=0x13), length 88

### 7.5 Support MC8090 3G Modules

UNO-1251G fully supported MC8090 3G Modules. When you insert the SIM card in MC8090 3G Modules, you will see /dev/ttyUSB0~ttyUSB4 after system boot up. Before use it, you need to do some settings on UNO-1251G.

I will give an example to show that how to use MC8090 3G Modules on UNO-1251G.

- Wireless Modules: MC8090 3G Modules
- SIM Card: China Mobile CMNET

#### 7.5.1 Controlling MC8090 3G Modules

You can use a Terminal program (for example, microcom, minicom) to open /dev/ttyUSB3(MC8090). The serial port can be found in the dmesg output, the serial port settings are usually 115200 bits per second, 8 data bits, No parity and 1 stop bit (115200 8N1). Then, you must send AT command to the wireless module to open the full functionality of module, and check the signal strength of the device before Dial-up in 7.5.2 chapter .

root@advantech:/dev# microcom -t 5000 -s 115200 /dev/ttyUSB3

AT+CFUN=1

ОК

AT+CSQ

+CSQ: 12,99

If this doesn't work, you either have the wrong port, or the wrong settings.

Now, you can use following AT command to dial. *"\$TELEPHONE\_NUMBER"* is a telephone number. Of course, you can also use other AT commands.

Note: /dev/ttyUSB3(MC8090) serial port are have been tested.

ttyUSB3# ATD**\$TELEPHONE\_NUMBER;** 

#### 7.5.2 PPP Dial-up Configuration

Open and modify /usr/sbin/ppp-on. You only need to modify the red font part. "*APN*" is the APN name for the connection. "*TELEPHONE*" is the telephone number for the connection. "*ACCOUNT*" is the account name for logon. "*PASSWORD*" is the password for this account. "*SERIAL\_PORT*" is serial port (for example /dev/ttyUSB3(MC8090) ). "*BAUD\_RATE*" is the serial port of baud rate (115200). And you need add the DNS of the Operators of you used or the DNS of public service platform into /usr/sbin/ppp-on before Initiating the connection.

**Note:** The DNS of public service platform is: **114.114.114**(114 DNS) and **8.8.8.8**(Google Public DNS).

ppp-on:

```
#!/bin/sh
#
# Script to initiate a ppp connection. This is the first part of the
# pair of scripts. This is not a secure pair of scripts as the codes
# are visible with the 'ps' command. However, it is simple.
#
# These are the parameters. Change as needed.
APN=CMNET # The APN for the connection
```



<b>TELEPHONE</b> =*99***1# # The telephone number for the connection
<b>ACCOUNT</b> =george # The account name for logon (as in 'George Burns')
<b>PASSWORD</b> =gracie # The password for this account (and 'Gracie Allen')
LOCAL_IP=0.0.0.0 # Local IP address if known. Dynamic = 0.0.0.0
REMOTE_IP=0.0.0.0 # Remote IP address if desired. Normally 0.0.0.0
NETMASK=255.255.255.0 # The proper netmask if needed
#
#Save the DNS Server IP of ppp0
if [ ! -f /etc/dnsserver/res.conf.ppp0 ]; then
touch /etc/dnsserver/res.conf.ppp0
fi
echo "nameserver <b>114.114.114.114</b> " > /etc/dnsserver/res.conf.ppp0
echo "nameserver <b>8.8.8.8</b> " >> /etc/dnsserver/res.conf.ppp0
#
# have a default route to an ethernet gateway.
#
exec /usr/sbin/pppd debug lock modem crtscts <b>SERIAL_PORT BAUD_RATE</b> \
asyncmap 20A0000 escape FF kdebug 0 \$LOCAL_IP:\$REMOTE_IP \
noipdefault netmask \$NETMASK defaultroute connect \$DIALER_SCRIPT
2. The "ACCOUNT" and "PASSWORD" of China Mobile dial-up are NULL. So, there is no need to v

2. The "ACCOUNT" and "PASSWORD" of China Mobile dial-up are NULL. So, there is no need to verify account and password in /etc/ppp/ppp-on-dialer. China Mobile alse would not need to verify the "APN" in /etc/ppp/ppp-on-dialer. If the "ACCOUNT" and "PASSWORD" of Operator are not NULL and the Operator need to verify the "APN", you should add the "ACCOUNT", "PASSWORD" and "APN" of Operator into /usr/sbin/ppp-on file. And then, modify /etc/ppp/ppp-on-dialer and add the following red font parts into it.

```
#!/bin/sh
```

#

# This is part 2 of the ppp-on script. It will perform the connection

# protocol for the desired connection.

#





Enabling an Intelligent Planet

exe	c chat -v		Υ
	TIMEOUT	3	١
	ABORT	'\nBUSY\r'	١
	ABORT	'\nNO ANSWER\r'	١
	ABORT	'\nRINGING\r\n\r\nRINGING\r'	١
	н	\rAT	١
	'OK-+++\c-OI	K' ATHO	١
	TIMEOUT	30	١
	ОК	'AT+CGDCONT=1,"IP",'\"\$APN\	"" <b>\</b>
	ОК	ATDT\$TELEPHONE	١
	CONNECT	н	١
	ogin:ogin:	\$ACCOUNT	X
	assword:	\$PASSWORD	

3. Now, you can run the following command to start dial-up.

advantech# ppp-on

4. In a moment, Run "ifconfig" command, then you can see the device **"ppp0"**. That means this dial-up is succeed.

advantech#	ŧ ifconfig
ррр0	Link encap:Point-to-Point Protocol
	inet addr:10.191.195.18 P-t-P:10.191.195.18 Mask:255.255.255.255
	UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
	RX packets:5 errors:0 dropped:0 overruns:0 frame:0
	TX packets:7 errors:0 dropped:0 overruns:0 carrier:0
	collisions:0 txqueuelen:3
	RX bytes:44 (44.0 B) TX bytes:87 (87.0 B)
5. Run the	following command to turn off "eth0" and "eth1".

advantech# ifconfig eth0 down

advantech# ifconfig eth1 down

6. At this point, ppp dial-up should be configured on UNO-1251G.

Run the following commands to test it on UNO-1251G with "eth0" and "eth1" are down states.



Advantech# sudo ping www.baidu.com

PING www.baidu.com (111.13.100.92): 56 date bytes

64 bytes from 111.13.100.92: seq=0 ttl=53 time=397.893 ms

64 bytes from 111.13.100.92: seq=1 ttl=53 time=296.752 ms

64 bytes from 111.13.100.92: seq=2 ttl=53 time=296.363 ms

64 bytes from 111.13.100.92: seq=3 ttl=53 time=255.389 ms

7. If you want to turn off the ppp-dial, run the following commands on UNO-1251G.

advantech# ppp-off

### 7.6 Support 968EMC0061(MC7304) 4G Modules

UNO-1251G fully supported MC7304 4G Modules. When you insert the SIM card in MC7304 4G Modules, you will see /dev/ttyUSB0~ttyUSB2 after system boot up. Before use it, you need to do some settings on UNO-1251G.

I will give an example to show that how to use MC7304 4G Modules on UNO-1251G.

- Wireless Modules: MC7304 4G Modules
- SIM Card: China Mobile CMNET

#### 7.6.1 Controlling MC7304 4G Modules

You can use a Terminal program (for example, microcom, minicom) to open /dev/ttyUSB2(MC7304). The serial port can be found in the dmesg output, the serial port settings are usually 115200 bits per second, 8 data bits, No parity and 1 stop bit (115200 8N1). Then, you must send AT command to the wireless module to open the full functionality of module, and check the signal strength of the device before Dial-up in 7.6.2 chapter .

root@advantech:/dev# microcom -t 5000 -s 115200 /dev/ttyUSB2

AT+CFUN=1

ОК

AT+CSQ

+CSQ: 21,99

If this doesn't work, you either have the wrong port, or the wrong settings.

Now, you can use following AT command to dial. *"\$TELEPHONE\_NUMBER"* is a telephone number. Of course, you can also use other AT commands.

Note: /dev/ttyUSB2(MC7304) serial port are have been tested.

ttyUSB3# ATD**\$TELEPHONE\_NUMBER;** 

#### 7.6.2 PPP Dial-up Configuration

Open and modify /usr/sbin/ppp-mc7304-on. You only need to modify the red font part. "APN" is the APN name for the connection. "TELEPHONE" is the telephone number for the connection. "ACCOUNT" is the account name for logon. "PASSWORD" is the password for this account. "SERIAL\_PORT" is serial port (for example /dev/ttyUSB2(MC7304)). "BAUD\_RATE" is the serial port of baud rate (115200). And you need add the DNS of the Operators of you used or the DNS of public service platform into /usr/sbin/ppp-on before Initiating the connection. Note: The DNS of public service platform is: 114.114.114(114 DNS) and 8.8.8.8(Google Public DNS). ppp-mc7304-on: #!/bin/sh # # Script to initiate a ppp connection. This is the first part of the *# pair of scripts. This is not a secure pair of scripts as the codes # are visible with the 'ps' command. However, it is simple.* # # These are the parameters. Change as needed. **APN**=CMNET # The APN for the connection # APN=3GNET # The APN for the connection **TELEPHONE**=\*99\*\*\*1# # The telephone number for the connection ACCOUNT=george # The account name for logon (as in 'George Burns') # The password for this account (and 'Gracie Allen') PASSWORD=gracie LOCAL IP=0.0.0.0 # Local IP address if known. Dynamic = 0.0.0.0 REMOTE IP=0.0.0.0 # Remote IP address if desired. Normally 0.0.0.0 NETMASK=255.255.255.0 # The proper netmask if needed # #Save the DNS Server IP of ppp0 if [ ! -f /etc/dnsserver/res.conf.ppp0 ]; then touch /etc/dnsserver/res.conf.ppp0 fi echo "nameserver 114.114.114.114" > /etc/dnsserver/res.conf.ppp0 echo "nameserver **8.8.8.8**" >> /etc/dnsserver/res.conf.ppp0 #



.....

# have a default route to an ethernet gateway.

#

exec /usr/sbin/pppd debug lock modem crtscts **SERIAL\_PORT BAUD\_RATE** \

asyncmap 20A0000 escape FF kdebug 0 \$LOCAL\_IP:\$REMOTE\_IP \

noipdefault netmask \$NETMASK defaultroute connect \$DIALER\_SCRIPT

3. The "ACCOUNT" and "PASSWORD" of China Mobile dial-up are NULL. So, there is no need to verify account and password in /etc/ppp/ppp-on-dialer. China Mobile alse would not need to verify the "APN" in /etc/ppp/ppp-on-dialer. If the "ACCOUNT" and "PASSWORD" of Operator are not NULL and the Operator need to verify the "APN", you should add the "ACCOUNT", "PASSWORD" and "APN" of Operator into /usr/sbin/ppp-mc7304-on file. And then, modify /etc/ppp/ppp-on-dialer and add the following red font parts into it.

```
#!/bin/sh
#
# This is part 2 of the ppp-on script. It will perform the connection
# protocol for the desired connection.
#
exec chat -v
    TIMEOUT
                 3
    ABORT
                 '\nBUSY\r'
    ABORT
                 '\nNO ANSWER\r'
                 '\nRINGING\r\n\r\nRINGING\r'
    ABORT
    ...
                 \rAT
    'OK-+++\c-OK' ATH0
    TIMEOUT
                 30
                  'AT+CGDCONT=1,"IP",'\"$APN\"" \
    ОК
    OK
                 ATDT$TELEPHONE
                                                 ١
                 н
    CONNECT
    ogin:--ogin: $ACCOUNT
                                                 ١
    assword:
                $PASSWORD
```

3. Now, you can run the following command to start dial-up.

#### advantech# ppp-mc7304-on

4. In a moment, Run "ifconfig" command, then you can see the device **"ppp0"**. That means this dial-up is succeed.

advantech# ifconfig

.....

ppp0

inet addr:10.191.195.18 P-t-P:10.191.195.18 Mask:255.255.255.255

UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1

RX packets:5 errors:0 dropped:0 overruns:0 frame:0

TX packets:7 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:3

*RX bytes:44 (44.0 B) TX bytes:87 (87.0 B)* 

5. Run the following command to turn off "eth0" and "eth1".

Link encap:Point-to-Point Protocol

advantech# ifconfig eth0 down

advantech# ifconfig eth1 down

6. At this point, ppp dial-up should be configured on UNO-1251G.

Run the following commands to test it on UNO-1251G with "eth0" and "eth1" are down states.

Advantech# sudo ping www.baidu.com

*PING www.baidu.com (111.13.100.92):* 56 date bytes

64 bytes from 111.13.100.92: seq=0 ttl=53 time=397.893 ms

64 bytes from 111.13.100.92: seq=1 ttl=53 time=296.752 ms

64 bytes from 111.13.100.92: seq=2 ttl=53 time=296.363 ms

64 bytes from 111.13.100.92: seq=3 ttl=53 time=255.389 ms

7. If you want to turn off the ppp-dial, run the following commands on UNO-1251G.

advantech# ppp-off

### 7.7 Support EWM-C109F601E/EWM-C109F6G1E

### (LISA-U200) 3G Modules

UNO-1251G fully supported LISA-U200 3G Modules. When you insert the SIM card in LISA-U200 3G Modules, you will see /dev/ttyACM0~ttyACM6 after system boot up. Before use it, you need to do some settings on UNO-1251G.

I will give an example to show that how to use LISA-U200 3G Modules on UNO-1251G.

➢ Wireless Modules: LISA-U200 3G Modules

SIM Card: China Mobile CMNET

#### 7.7.1 Controlling LISA-U200 3G Modules

You can use Terminal program (for example, microcom, minicom) to open /dev/ttyACM0(LISA-U200). The serial port can be found in the dmesg output, the serial port settings are usually 460800 bits per second, 8 data bits, No parity and 1 stop bit (460800 8N1). Then, you must send AT command to the wireless module to open the full functionality of module, and check the signal strength of the device before Dial-up in 7.7.2 chapter .

root@advantech:/dev# microcom -t 5000 -s 460800 /dev/ttyACM0

AT+CFUN=1

ОК

AT+CSQ

+CSQ: 23,99

If this doesn't work, you either have the wrong port, or the wrong settings.

Now, you can use following AT command to dial. *"\$TELEPHONE\_NUMBER"* is a telephone number. Of course, you can also use other AT commands.

Note: /dev/ttyACM0(LISA-U200) serial port are have been tested.

ttyUSB3# ATD\$TELEPHONE\_NUMBER;

#### 7.7.2 PPP Dial-up Configuration

Open and modify /usr/sbin/ppp-ublox-on. You only need to modify the red font part. "APN" is the APN name for the connection. "TELEPHONE" is the telephone number for the connection. "ACCOUNT" is the account name for logon. "PASSWORD" is the password for this account. "SERIAL\_PORT" is serial port (for example /dev/ttyACM0(LISA-U200)). "BAUD\_RATE" is the serial port of baud rate (460800). And you need add the DNS of the Operators of you used or the DNS of public service platform into /usr/sbin/ppp-ublox-on before Initiating the connection.

**Note:** The DNS of public service platform is: **114.114.114**(114 DNS) and **8.8.8.8**(Google Public DNS).

ppp-ublox-on:

#!/bin/sh
#
#
# Script to initiate a ppp connection. This is the first part of the
# pair of scripts. This is not a secure pair of scripts as the codes
# are visible with the 'ps' command. However, it is simple.
#
# These are the parameters. Change as needed.



APN=CMNET	# The APN for the connection
# <mark>APN</mark> =3GNET	# The APN for the connection
<b>TELEPHONE</b> =*99***1#	# The telephone number for the connection
ACCOUNT=george	# The account name for logon (as in 'George Burns')
PASSWORD=gracie	# The password for this account (and 'Gracie Allen')
LOCAL_IP=0.0.0.0 # Lo	ocal IP address if known. Dynamic = 0.0.0.0
REMOTE_IP=0.0.0.0	# Remote IP address if desired. Normally 0.0.0.0
NETMASK=255.255.255	.0 <i># The proper netmask if needed</i>
#Save the DNS Server IF	P of ppp0
if [ ! -f /etc/dnsserver/re	es.conf.ppp0 ]; then
touch /etc/dn	sserver/res.conf.ppp0
fi	
echo "nameserver 114.	114.114.114" > /etc/dnsserver/res.conf.ppp0
echo "nameserver 8.8.8	3.8" >> /etc/dnsserver/res.conf.ppp0
#	
# have a default route t	o an ethernet gateway.
#	
exec /usr/sbin/pppd del	bug lock modem crtscts <b>SERIAL_PORT BAUD_RATE</b> \
asyncmap 20A000	0 escape FF kdebug 0 \$LOCAL_IP:\$REMOTE_IP \
noipdefault netma	sk \$NETMASK defaultroute connect \$DIALER_SCRIPT

4. The "ACCOUNT" and "PASSWORD" of China Mobile dial-up are NULL. So, there is no need to verify account and password in /etc/ppp/ppp-on-dialer. China Mobile alse would not need to verify the "APN" in /etc/ppp/ppp-on-dialer. If the "ACCOUNT" and "PASSWORD" of Operator are not NULL and the Operator need to verify the "APN", you should add the "ACCOUNT", "PASSWORD" and "APN" of Operator into /usr/sbin/ppp-ublox-on file. And then, modify /etc/ppp/ppp-on-dialer and add the following red font parts into it.

#!/bin/sh

#

# This is part 2 of the ppp-on script. It will perform the connection



# protocol for th	e desired connection.	
#		
exec chat -v		Ι
TIMEOUT	3	Ι
ABORT	'\nBUSY\r'	Ι
ABORT	'\nNO ANSWER\r'	/
ABORT	'\nRINGING\r\n\r\nRINGING\r'	/
н	\rAT	/
'OK-+++\c-(	ОК' АТНО	/
TIMEOUT	30	/
ОК	'AT+CGDCONT=1,"IP",'\"\$APN	Λ""
ОК	ATDT\$TELEPHONE	/
CONNECT	н	/
ogin:ogin	: \$ACCOUNT	A second s
assword:	\$PASSWORD	

3. Now, you can run the following command to start dial-up.

advantech# ppp-ublox-on

4. In a moment, Run "ifconfig" command, then you can see the device **"ppp0"**. That means this dial-up is succeed.

advantech#	ifconfig
ррр0	Link encap:Point-to-Point Protocol
	inet addr:10.191.195.18 P-t-P:10.191.195.18 Mask:255.255.255.255
	UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
	RX packets:5 errors:0 dropped:0 overruns:0 frame:0
	TX packets:7 errors:0 dropped:0 overruns:0 carrier:0
	collisions:0 txqueuelen:3
	RX bytes:44 (44.0 B) TX bytes:87 (87.0 B)
5. Run the f	ollowing command to turn off "eth0" and "eth1".
advantacht	tifeonfig athe down

advantech# ifconfig eth0 down



Enabling an Intelligent Planet

advantech# ifconfig eth1 down

6. At this point, ppp dial-up should be configured on UNO-1251G.

Run the following commands to test it on UNO-1251G with "eth0" and "eth1" are down states.

Advantech# sudo ping www.baidu.com

PING www.baidu.com (111.13.100.92): 56 date bytes

64 bytes from 111.13.100.92: seq=0 ttl=53 time=397.893 ms

64 bytes from 111.13.100.92: seq=1 ttl=53 time=296.752 ms

64 bytes from 111.13.100.92: seq=2 ttl=53 time=296.363 ms

64 bytes from 111.13.100.92: seq=3 ttl=53 time=255.389 ms

7. If you want to turn off the ppp-dial, run the following commands on UNO-1251G.

advantech# ppp-off

### 7.8 Support EWM-W150H02E(RT5390) Wireless Modules

UNO-1251G fully supported RT5390 Wireless Modules. When you insert the RT5390 Wireless Modules, you will see the message in "Isusb". Before use it, you need to do some settings on UNO-1251G.

I will give an example to show that how to use RT5390 Wireless Modules on UNO-1251G.

Attention: In order to avoid the Wifi Connection exception, do not set network when wifi course of connecting (Probably one minute).

#### 7.8.1 Scan the nearby available AP

advantech# insmod /usr/src/kernel/module/rt5370sta.ko

advantech# ifconfig ra0 up

advantech# iwlist ra0 scanning

Then you will see the nearby available AP list , just like the following context:

ra0	Scan completed :		
	Cell 01 - Address: 14:2D:27:EF:DD:C7		
	Protocol:802.11b/g/n		
	ESSID:"share_wifi"		
	Mode:Managed		
	Frequency:2.412 GHz (Channel 1)		
	Quality=0/100 Signal level=-95 dBm Noise level=-90 dBm		
	Encryption key:on		



#### Bit Rates:72 Mb/s

IE: IEEE 802.11i/WPA2 Version 1

Group Cipher : CCMP

Pairwise Ciphers (1) : CCMP

Authentication Suites (1) : PSK

#### 7.8.2 Using password-less wifi connection

.....

1. Use following command to connect wifi with password-less.

advantech# wifi-nopwdon ESSID

The "ESSID" is wifi connection name.

2. Use following command to disconnect wifi with password-less.

advantech# wifi-off

#### 7.8.3 Using WPA wifi connection

Open and modify /usr/sbin/wifi-wpaon. You only need to modify the red font part. "ESSID" is the wifi host name for the connection. "PASSWD" is the password for the connection. If you need to config static, enable "Set Static IP" "Set DNS Server" "Set Gateway" and disable "Do dhcp for ra0". The DNS server "*nameserver*" is 8.8.8.8 & 114.114.114.114. Static IP is *192.168.1.111*, Default gateway "*gw*" is 192.168.1.1 .

#!/bin/sh
ESSID=\$1
PASSWD=\$2
# Start
# Set Static IP
#ifconfig ra0 <b>192.168.1.111</b> up
# Set DNS Server
#echo "nameserver 114.114.114.114" > /etc/dnsserver/res.conf.ra0
#echo "nameserver 8.8.8.8" >> /etc/dnsserver/res.conf.ra0
# Set Gateway
#route del default



Enabling an Intelligent Planet

#route add default gw 192.168.1.1

ifconfig ra0 up

.....

# Do dhcp for ra0

/etc/init.d/dhcpc ra0 start

.....

1. Use following command to connect wifi with WPA encryption.

advantech# wifi-wpaon ESSID PASSWD

Parameter "ESSID" is the wifi connection name , "PASSWD" is the corresponding password .

2. Use following command to disconnect wifi with WPA encryption.

advantech# wifi-off

Enabling an Intelligent Plane

## 8. Software Management

### 8.1 OPKG Package Manager

Opkg is a lightweight package management system. It is written in C and resembles

apt/dpkg/yum in operation. It is intended for use on embedded Linux devices and is used in this capacity in the OpenEmbedded and OpenWrt projects.

Advantech Embedded Linux for UNO-1251G has built-in OPKG package manager, with this tool you can install most of the required software online, and manage them, such as uninstall, upgrades and so on.

## 8.2 Installation a New Software

If you want to install a software which is not exist in the current OS, you should follow the steps below

**Step 1:** Update the online software source:

advantech# opkg update

**Step 2**: Search whether the software source server has the software you need.

advantech# opkg list|grep SW\_KW

**Note.** SW\_KW is the keywords of the software name, for example, you want to search a ftp server, the SW\_NAME should be 'ftp'.

**Step 3:** Find the full name of the software you need in the search result list. And install it by following command:

advantech# opkg install SW\_FULL\_NAME

### 8.3 More about OPKG

More about use and development of OPKG, Please refer to the project website of OPKG:

https://code.google.com/p/opkg/

### 8.4 Example

I listed some softwares installation. If you need, you can refer to the following examples and install them.

1. If you want to install snmp server, run the following command.

advantech# opkg install net-snmp-server

2. If you want to install snmp client, run the following command.

advantech# opkg install net-snmp-client

3. If you want to install perl, run the following command.

advantech# opkg install perl

4. If you want to install cron server, run the following command.

advantech# opkg install cron



Enabling an Intelligent Planet

5. If you want to install python, run the following command.

advantech# opkg install python

## 9. Development

### 9.1 Advantech AM3517 Development Kit Overview

We have provided a SDK to facilitate your development of TI AM3517 based devices, such as UNO-1251G.

It contains the following content:

- Cross-compiler tool chain for am3517
- Embedded QT 4.7.2 and QT 4.8.3 development library
- Embedded QT 4.7.2 and QT 4.8.3 development tools
- OpenGL library for QT/E
- Example source code for UNO-1251G

#### 9.2 Installation

Please download the SDK package, and run the following command to extract it:

hostPC\$ sudo tar jxvf adv-am3505-devkit-3.0.tar.bz2

Then, go to the directory, and run the installation script:

hostPC\$ sudo cd adv-am3505-devkit-3.0

hostPC\$ sudo ./install.sh

#### 9.3 Development of C/C++ Programs

Development C/C++ program for UNO-1251G is consistent with X86, you just need to modify your makefile, and add a prefix arm-arago-linux-guneabi- to the gcc or g++.

### 9.4 Development of GUI Programs with QT 4.7.2 and QT

#### 4.8.3 Library

With the development kit, you can develop a qt-based GUI program. Follow these steps, you can quickly convert your QT Project to a GUI application for UNO-1251G:

1. Enter the directory of QT source project;

- 2. Remove old makefile;
- 3. Then, if you want to develop of GUI Programs with QT 4.7.2 library, run the following command:

hostPC\$ sudo adv-qt-make-4.7.2

if you want to develop of GUI Programs with QT 4.8.3 library, run the following command:

hostPC\$ sudo adv-qt-make-4.8.3

4.If everything goes well, an executable file will be generated in the directory. Copy the file to UNO-1251G;

5.Run it.

### 9.5 Deployment of the Application

#### 9.5.1 Starting Your Custom Application on System Boot up

You can add your application to the boot up list, and your application will start when Linux boots up. Follow these steps to add your application to the boot up list on UNO-1251G:

1. Copy your application to the path /usr/bin/;

2. If the application has some libraries, copy them to the path /usr/lib;

3. Edit the file /etc/adv.d/usrapp, and add your application to the application list.

For example, your application has been copied to the path /usr/bin/demoapp, you should add a line to the /etc/adv.d/usrapp like this:

/usr/bin/demoapp&

"&" means this application will run on background.

4. If your application has a GUI show, you should stop the Matrix GUI first, please refer to Section 6.5.4.

#### 9.5.2 Adding a New Application/Directory to Matrix

Below are step-by-step instructions for Adding a New Application/Directory to Matrix.

1. Create a new folder on your target file system at /usr/share/matrix-gui-2.0/apps/. The name should be a somewhat descriptive representation of the application or directory. The folder name must be different from any existing folders at that location.

2. Create a desktop file based on the parameters discussed below. It is recommended the name of the desktop file match the name of the newly created folder. No white spaces can be used for the desktop filename. The desktop file parameters should be set depending on if you want to add a new application or a new directory to Matrix. The Type field must be set according to your decision. The desktop file must have the desktop suffix.

3. Update the Icon field in the desktop to refer to any existing Icon in the

/usr/share/matrix-gui-2.0 directory or subdirectories. You can also add a new 96x96 png image and place it into your newly created folder.

4. Optionally for applications you can add a HTML file that contains the application description in your newly created directory. If you add a description page then update the X-Matrix-Description field in the .desktop file.

5. Refresh Matrix using the application "Refresh Matrix" located in the Settings submenu.

Blank template icons for Matrix can be found here:

http://gforge.ti.com/gf/download/frsrelease/712/5167/blank\_icons\_1.1.tar.gz

The .desktop file is based on standard specified at the URL:

http://standards.freedesktop.org/desktop-entry-spec/latest/

Additional fields were added that are unique for Matrix.

Format for each parameter:

<Field>=<Value>

The fields and values are case sensitive.

### 9.6 Development of Kernel Driver modules

With the development kit, you can compile driver modules for UNO-1251G. You must ensure that

you installed adv-am3505-devkit-2.0. Follow these steps, you can quickly compile driver module for UNO-1251G:

1. Enter the directory of adv-am3505-devkit-2.0/kernel-dev-UNO1251G;

2. Refer to example driver and ensure that the **"RISC\_KERNEL\_SRC"** of your drivers Makefile is current **"kernel-UNO1251G"** directory;

3. Then, run the "make" command;

4. If everything goes well, a driver module file will be generated in the directory. Copy the file to UNO-1251G and insmod it.

### 9.7 Source of Example

#### 9.7.1 Serial Port Programming

Please refer to <DEVKIT\_PATH>/Example/serial/

st.c and stl.c is an example of sending/receiving data via the serial port.

Send usage:

advantech# ./st /dev/ttyS0 -s -b 9600 -n81 -h 60

receive.c is an example of receiving data via the serial port.

Receiver usage:

```
advantech# ./st /dev/ttyS0 -r -b 9600 -n81 -h 60
```

Before using your program of serial port, please ensure that your serial port is in 232/422/485 mode.

#### 9.7.2 OLED Programming

Please refer to <DEVKIT\_PATH>/Example/oled/

Testoled.c is an example of control oled screen.

Usage:

Ę	ge:		
	-ic:	Disable all	icons.
	-io:	Enable sho	own iconswc: Wifi icon off.
	-wo:	Wifi icon s	how.
	-mc:	Mobile icc	on off.
	-mo:	o: Mobile icon show.	
-ct: Clean tex		Clean text	area.
	-ca:	Clean all s	creen.
	-dt: Display test.		st.
-l: print curre		print curre	ent message numbers for all level queue to kernel dmesg.
-h: Get more		Get more	help.
-rt [PERIOD]: PERIOD:		DD]: I	Repeat period for round-robin message show.
		IOD:	1~60;message show period (Sec).

-so ONCE;[PERIOD];[SIZE];[MSG]: Add a message which only show once.

PERIOD :	1~65535; message show period (Sec).
SIZE:	m/l or M/L; font size.

AD\ANTECH Enabling an Intelligent Planet

Advantech UNO-1251G Manual

MSG: message; must be smaller then 32 characters.

-st [LEVEL];[LABEL];[SIZE];[MSG]: Add message to round-robin queue (Repeat show).

LEVEL:	HIGH/LOW/DEL.
LABEL:	is used to track the message on message queue, must be smaller then 8.
SIZE:	m/M or I/L; font size.
MSG:	message; must be smaller then 32 characters

Attention:

Maximum message length is 32 by default, beyond the part of string will not be displayed!

Example:

Close the wifi icon:

advantech# ./testoled -wc

Set the round-robin message repeat (period = 35s):

advantech# ./testoled -rt 35

The once message "Hello Advantech" would show 10s with "m/M" size font:

(Method 1)

advantech# ./testoled -so "ONCE;10;m;Hello Advantech"

(Method 2)

advantech# echo "ONCE;<period>;<size>;<msg>" > /dev/oled

The round-robin message "eth0 172.21.73.21" with LOW priority, label is "lan" and size is set to M would show:

(Method 1)

advantech# ./testoled -st "LOW;lan;M;eth0 172.21.73.21"

(Method 2)

advantech# echo "<level>;<label>;<size>;<msg>" > /dev/oled

Delete the message with "lan" label (The highest priority message of all message queue will be delete):

(Method 1)

advantech# ./testoled -st "DEL;lan"

(Method 2)

advantech# echo "ONCE;<label>" > /dev/oled

# 10. Appendix

### 10.1 U-boot Configuration

#### 10.1.1 Enter U-boot mode

You can enter U-BOOT mode by press the "Enter Key" several times just after the device powered on.

#### 10.1.2 Loading Kernel from micro SD

As a general rule, you need to load Linux kernel form the internal micro SD card. If you have followed Section 4.2, and installed the Linux kernel on the micro SD card, then run the following command on the console of UNO-1251G:

U-boot> loaduimage=fatload mmc \${mmcdev} \${kloadaddr} ulmage

If you need to load kernel from tftp server, please refer to section 8.1.2

#### 10.1.3 Loading Kernel from Tftpd

Sometimes, you will need to load the Linux kernel over the network. Follow these steps:

- 1. Ensure that the UNO-1251G has been connected to the LAN.
- 2. Start the tftp Service on your PC.
- 3. Set the IP address of tftp server:

U-boot> set serverip YOUR\_SERVER\_IP

4. Set IP for UNO-1251G:

U-boot> set ipaddr LOCAL\_IP

5. Set net mask:

U-boot> set netmask 255.255.255.0

6. Set the load command of ulmage:

U-boot> set loaduimage 'tftp \${kloadaddr} uImage'

#### 10.1.4 Mounting File system

You can mount root file system form either internal boot SD or external SD.

If you want to mount root file system from the internal micro SD, please run the following command on UNO-1251G:

U-boot> set mmcargs 'run bootargs\_defaults; setenv bootargs \${bootargs} root=/dev/mmcblk0p2 ro

rootfstype=ext3 rootwait'

If you want to mount root file system from the external SD, please run the following command on UNO-1251G:

U-boot> set mmcargs 'run bootargs\_defaults; setenv bootargs \${bootargs} root=/dev/mmcblk1p1 ro

rootfstype=ext3 rootwait'



Enabling an Intelligent Plan

#### 10.2 APIs of ADVOLED Driver

int open( constchar \*pathname, intflags )

Description: Open device node file for reading or writing.

Header file: <sys/types.h>, <sys/stat.h>, <fcntl.h>

Return Value: File descriptor ( on success ) , -1 ( on error ) .

int close ( int fd );

Description: Close device node file.

Header file: <sys/types.h>, <sys/stat.h>, <fcntl.h>

Parameters:

fd: File descriptor.

Return Value: 0 ( on success ) , -1 ( on error ) .

ssize\_t read ( int fd, void \*buf, size\_t count );

Description: Read from a file descriptor.

Header file: <unistd.h>

Parameters:

fd: File descriptor. buf: Read context storage path. count: Read size.

Return Value: The number of bytes read ( on success ) , -1 ( on error ) .

ssize\_t write ( int fd, void \*buf, size\_t count );

Description: Write to a file descriptor.

Header file: <unistd.h>

Parameters:

fd: File descriptor.

buf: Write context storage path.

count: Write size.

Return Value: The number of bytes written ( on success ) , -1 ( on error ) .

int ioctl ( int d, int request, . . . );

Description: Control device.

Header file: <sys/ioctl.h>

Parameters:

d: File descriptor.

Request: Device dependent request code.

...: Untyped pointer to memory. It's traditionally char \*argp ( from the days before void \* was valid C ) , and will be so named for this discussion.

request: 0 ( on success ) , -1 ( on error ) .

### 10.3 RTC Battery Automatic detection

The advbattery\_check driver will be detection the RTC battery status and show the low battery alarm on OLED in real-time. Current system has automatically load the advoled & advbattery\_check driver



by default . The advbattery\_check will respond the gpio\_10 low or falling interrupt and call the advoled to show alarm.

### 10.4 Switch of Serial Port

The com1 is 485 mode, com2 is 422/485 mode, com3 is 232 mode on UNO-1251G.

You can set sw7 switch to shift the 485/422 slaver mode or 422 Master mode for com2:

1 1 1 1 -> 422 Master mode

1 1 1 0 -> 485/422 slaver mode

### 10.5 VNC remote login

The x11vnc server has been self-start on UNO1251G runlevel 5 by default .User can connect the vnc server by following these steps. In this case, the use of software is RealVNC Viewer. And currently supported vnc client as bellow:

RealVNC Viewer TightVNC Viewer UltraVNC Viewer .NET VNC Viewer

Step1. Launch the VNC client software ( e.g. RealVNC Viewer ) .



Step2. Use command "ifconfig" to get the current IP address from the host.



Step3. Set VNC server IP.



	VQ VNC Viewer			
	VNC® Viewer	V2		
	VNC Serve r: 172.21.73.184:1	<b>_</b>		
	Encryption: Let VNC Server choose	•		
	About Options	Connect		
Step4. Do Connect.				
	V2 VNC Viewer			
	VNC® Viewer	V2		
	VNC Server: 172.21.73.184:1	<b>_</b>		
	Encryption: Let VNC Server choose	•		
	About Options	Connect		
Step5. Do Continue.				
	V2 VNC Viewer - Encryption	×		
Unencrypted connection				
	The connection to this VNC Server will not be encrypted.			
	VNC Server: 172.21.73.184::5901			
	Your authentication credentials will be transmitted securely, but all subsequent data exchanged while the connection is in progress may be susceptible to interception by third parties.			
	🔲 Don't warn me about this again.			
	Con	ntinue Cancel		

After successfully connecting, you will see the xfce4 Disktop.





### 10.6 Get current IP address

Use command "ipdetect" to get the current IP address and print them in OLED screen.

advantech# ipdetect

Use command "ipdetect dev" to get the network status of testing equipment "dev".

dev: eth0/eth1/ra0/ppp0

advantech# ipdetect eth0