

M-A5D35

Linux-Ready Cortex-A5 SOM

&

Matrix-7XX

Linux-Ready Cortex-A5 IoT Gateway

Software Guide

Version: 1.3

2023 Nov.



Trademarks

The Artila logo is a registered trademark of Artila Inc. All other trademarks or registered marks in this manual belong to their respective manufacturers.

Disclaimer

Information in this document is subject to change without notice and does not represent a commitment on the part of Artila.

Artila provides this document as is, without warranty of any kind, either expressed or implied, including, but not limited to, its particular purpose. Artila reserves the right to make improvements and/or changes to this manual, or to the products and/or the programs described in this manual, at any time.

Information provided in this manual is intended to be accurate and reliable. However, Artila assumes no responsibility for its use, or for any infringements on the rights of third parties that may result from its use.

This product might include unintentional technical or typographical errors. Changes are periodically made to the information herein to correct such errors, and these changes are incorporated into new editions of the publication.

Document Amendment History

Revision	Date	Remark
V 0.1	2016 Feb.	Initial
V 0.3	2017 Jan.	User Guide
V 1.0	2017 Oct.	Software Guide
V 1.1	2018 Apr.	Add information: GNSS, MEMS, DIO, Audio ..
V 1.2	2020 Jul.	Content amended
V 1.3	2023 Nov.	Serial Port setting (Matrix-710)

Table of Contents

1. Overview	6
2. Access the USB Serial Console	8
2.1 USB Serial Console Introduction	8
2.2 USB Serial Console Log-in	9
3. Network Settings.....	10
3.1 Config the Network Interface.....	10
3.2 Configure the DNS Server	11
4. Access the SSH Console.....	12
5. Check Linux Kernel Version.....	13
6. File System Information.....	14
7. Serial Port Settings	15
7.1 Port Mapping	15
7.2 Configure the Serial Port.....	15
8. System Time and Real-Time Clock(RTC).....	17
8.1 Adjust System Time by data Command	17
8.2 Adjust RTC by hwclock Command	17
8.3 Synchronize System Time by NTP Server	17
8.3.1 Install the ntpdate utility.....	17
8.3.2 Using the ntpdate utility.....	18
9. Insert Kernel Modules.....	19
10. Insert Software Package.....	20
11. Mount/Unmount an SD Card.....	21
12. Mount/Unmount a USB Card	22
13. Web Server Settings	23
13.1 Nginx Web Server.....	23
13.2 Root Web Page Directory	23
13.3 PHP Support.....	24
14. Auto-execute User Applications/Shell Scripts	25
14.1 Modify the /etc/rc5.d directory	25
14.2 Modify the /etc/profile.....	25
15. Change the Welcome Message	26
16. Reboot the System.....	27
17. User Application Development	28
17.1 Install C/C++ Cross Compilation Toolchain	28
17.2 Using the C Cross Compiler.....	28
17.3 Using the C++ Cross Compiler.....	29
17.4 Using the Native C Compiler.....	29

17.5	Using the Native C++ Compiler.....	30
17.5.1	Install the Native C++ Toolchain	30
17.5.2	Using the Native C++ Compiler.....	30
17.6	Using the Python Interpreter	31
17.6.1	Python 2 Support	31
17.6.2	Python 3 Support	31
18.	GPIO Operation	32
19.	Install an USB Wi-Fi Dongle	34
19.1	Install Hardware Driver.....	34
19.2	Modify the network interface configuration	34
19.3	Modify the WPA configuration	35
19.4	Restart the wireless network interface.....	35
20.	Webmin Support	36
21.	Setup Eclipse IDE.....	37
21.1	Configure the Eclipse IDE.....	37
21.2	Install the Eclipse Yocto Plug-in	39
21.3	Configuring the Cross-Compiler Options.....	40
21.4	Create a Hello World Project.....	41
22.	Setup miniPCle (mPCle) Power.....	44
23.	Setup SIM card	45
24.	Setup “STATUS” LED indicator	48
25.	Setup Digital Input / Digital Output	49
26.	Setup GNSS & IMU.....	50
27.	Setup Audio Out.....	51
28.	Restore to Factory Default.....	52
29.	Backup the “File System” of Matrix-700/710/713	54
30.	Restore the “File System” of Matrix700/710/713	56

1. Overview

This software guide applies to Artila's Matrix-7xx series Industrial IoT gateway and M-A5D35 SoM (System on Module).

- **Operation System**

- Linux kernel 5.10.x / 4.19.x
- Supports bootup from eMMC or SD card
- Support Backup/Restore via SD card or USB device
- Boot Loader: Barebox
- File System: EXT4
 - M-A5D35 uses EXT4 file system for the built-in flash memory disk.
 - The files system is stored at NAND flash memory.

- **Software Development**

- Toolchain: gcc 9.3 / 6.2.x + glibc 2.31 / 2.24
- Supports in-place C/C++ code compilation

- **Package Management**

- Package repository: Artila self-maintained repository
- Command: Using standard apt-get command

- **Popular Packages**

- Web server: Apache/Nginx/Lighttpd
- Database: MySQL/SQLite3/PostgreSQL
- Script Language: PHP/Python/Perl/NodeJS
- Text editor: vim/nano/sed
- Administration: Webmin

- **Protocol Stacks**

- IPV4, ICMP, ARP, DHCP, NTP, TCP, UDP, FTP, HTTP, PPP, PPPoE, CHAP, PAP, SMTP, SNMP V1/V3, SSL, SSH 1/2

- **Utilities**

- Bash: Shell Command
- Telnet: Telnet client program
- Busybox: Linux utility collection
- FTP: FTP client program

- **Daemon**
 - pppd: Dial In/out over serial port and PPPoE
 - snmpd: SNMP agent program
 - inetd: TCP server program
 - ftpd: FTP server program
 - nginx: Web server program
 - sshd: secured shell server
 - iptables: Firewall service manager
- **Standard Device Drivers**
 - ttyS0: serial console port (M-A5D35 debug port)
 - ttyS1~ttyS4: serial ports (M-A5D35 UART0~UART3)
 - gpio: General Purpose I/O
 - mmc: SD/MMC:
 - rtc: Real Time Clock
 - sda: USB flash memory disk
 - ttyACM: USB Modem
 - ttyUSB: USB RS-232 adaptor
 - spi: spi bus
- **I/O devices Control**

M-A5D35 uses standard I/O device control to access following devices:




 - Ethernet: eth0, eth1
 - Serial Ports: ttyS1, ttyS2, ttyS3, ttyS4
 - Serial Console Port: ttyS0
 - Real time clock: rtc0
 - USB Flash Disk: sda, sda1, sdb, sdb1
 - SD memory Card: mmc0
 - USB WLAN dongle: wlan0
 - USB Serial Cable: ttyUSB0, ttyUSB1
 - SPI bus: spi0, spi1
- **Default Setting**
 - IP Default setting:
 - eth0: DHCP
 - eth1: 192.168.2.127 (Netmask: 255.255.255.0)
 - ssh Login: root
 - Password: root
 - Terminal type: VT100

2. Access the USB Serial Console

2.1 USB Serial Console Introduction

All the M-A5D35 based Matrix IoT gateways come with a USB client port (micro-USB connector), which is used as the serial console. Please prepare a USB-to-microUSB cable to connect the Matrix IoT gateway to a Desktop/Notebook PC. The Matrix IoT gateway can be directly driven by USB power.

When the Matrix IoT gateway finished its boot up process, it will automatically emulate an USB CDC/ACM compatible serial device.

M-A5D35 based Matrix IoT gateway comes with a USB client port	Use a standard on-the-shelf USB-to-MicroUSB cable to connect to the Matrix-7XX	Linux/Windows/OSX Desktop/Notebook PC
		

The identifier name of the CDC/ACM serial port varies depending on your computer's operation system and the numbers of the serial ports which are already installed on your computer.

On Linux system, the serial port name appears like ttyACM0, ttyACM1, etc.

On OSX system, the serial port name appears like tty.usbmodem1421, tty.usbmodem1422, etc.

On Windows system, the serial port name appears like COM3, COM4, etc.

The serial communication parameters are: **115200**, **N81**, **VT100**. Use your preferred serial terminal tools to access the Matrix IoT gateway's serial console.

For example:

On Windows system, use **putty** or **teraterm**.

On Linux/OSX system, use **minicom** utility.

Note

For Linux, Mac OSX and Windows 10 computers, the CDC/ACM serial driver is already built-in and will be activated automatically.

For Windows 7/XP computers, it may need to install the CDC/ACM serial driver manually. Users can download the CDC/ACM driver from Artila web site.

(<http://www.artila.com/download/A5D35/Linux/toolchain/linux-cdc-acm.inf>).

2.2 USB Serial Console Log-in

User name: root

Password: root

Following example by Matrix-7XX

```

Welcome to

      **           ** **
      **           ** **
     ** **       ** **
     ** **       **** **** ** *****
 **         ** ** ** ** ** ** ** ** ** ** ** **
 **         ** ** ** ** ** ** ** ** ** ** *****
*****       ** ** ** ** ** ** ** ** ** ** ** **
**          ** ** **^ ^ ^ ^ ^ ^ ^ ^
**          ** **   ** ** ** *****

For further information check:
http://www.artila.com/

Poky (Yocto Project Reference Distro) 2.2 matrix710 /dev/ttyGS0

matrix710 login: root
Password:
Welcome to

      **           ** **
      **           ** **
     ** **       ** **
     ** **       **** **** ** *****
 **         ** ** **^ ^ ^ ^ ^ ^ ^ ^
 **         ** ** **^ ^ ^ ^ ^ ^ ^ ^
*****       ** ** **^ ^ ^ ^ ^ ^ ^ ^
**          ** ** **^ ^ ^ ^ ^ ^ ^ ^
**          ** **   ** ** ** *****

For further information check:
http://www.artila.com/

[root@matrix710 ~]#

```

3. Network Settings

3.1 Config the Network Interface

The M-A5D35 based Matrix IoT gateways come two Ethernet ports, the default network settings are shown below:

Ethernet Type	Port Label	Device mapping	IP mode	IP address
Gigabit	GLAN	eth0	DHCP	auto
10/100Mbit	LAN	eth1	static	192.168.2.127

Users may need to modify the network settings to meet their LAN environment. The network interface configuration file path is **/etc/network/interfaces**. Edit and save the configuration file, then use **ifdown** and **ifup** command to ON/OFF the specific network interface to activate the network settings.

```
[root@Matrix700 ~]# cat /etc/network/interfaces
# /etc/network/interfaces -- configuration file for ifup(8), ifdown
(8)

# The loopback interface
auto lo
iface lo inet loopback

# Wired or wireless interfaces
# Gigabit
auto eth0
iface eth0 inet dhcp

# 10/100M
auto eth1
iface eth1 inet static
    address 192.168.2.127
    netmask 255.255.255.0
    network 192.168.2.0
    #gateway 192.168.2.1

[root@Matrix700 ~]#
```

The following screen capture shows the eth0 of the Matrix-7xx got a valid IP: 192.168.1.93.

```
[root@Matrix700 ~]# ifdown eth0
[root@Matrix700 ~]# ifup eth0
udhcpc: option -h NAME is deprecated, use -x hostname:NAME
udhcpc (v1.24.1) started
Sending discover...
Sending select for 192.168.1.93...
Lease of 192.168.1.93 obtained, lease time 86400
/etc/udhcpc.d/50default: Adding DNS 208.67.220.220
/etc/udhcpc.d/50default: Adding DNS 208.67.222.222

[root@Matrix700 ~]#
```

3.2 Configure the DNS Server

The DNS configuration file path is `/etc/resolv.conf`. Users may edit the file according to their specific network environment.

```
[root@matrix700 ~]#cat /etc/resolv.conf

[root@matrix700 ~]#ifconfig eth1
eth1      Link encap:Ethernet  HWaddr 00:13:48:03:08:4a
          inet addr:192.168.2.127  Bcast:192.168.2.255  Mask:255.255.255.0
          inet6 addr: fe80::213:48ff:fe03:84a/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:126 errors:0 dropped:0 overruns:0 frame:0
          TX packets:45 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:14966 (14.6 KiB)  TX bytes:3770 (3.6 KiB)
          Interrupt:41 Base address:0xc000

[root@matrix700 ~]#ping google.com
ping: unknown host google.com

[root@matrix700 ~]#echo 'nameserver 8.8.8.8' > /etc/resolv.conf

[root@matrix700 ~]#cat /etc/resolv.conf
nameserver 8.8.8.8

[root@matrix700 ~]#ping google.com
PING google.com (216.58.200.238) 56(84) bytes of data.
64 bytes from tsa03s01-in-f14.1e100.net (216.58.200.238): icmp_seq=
1 ttl=52 time
=13.9 ms
64 bytes from tsa03s01-in-f238.1e100.net (216.58.200.238): icmp_seq
=2 ttl=52 tim
e=15.3 ms
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 13.910/14.616/15.322/0.706 ms

[root@matrix700 ~]#
```

Warning: Please be noted that, the `/etc/resolv.conf` is physically located in the RAM disk, so the content of the file will disappear after system reboot.

4. Access the SSH Console

Most Linux/OSX computers come with built-in SSH client utility. For Windows users, it is highly recommended to use **putty** as an SSH client.

User name: **root**

Password: **root**

```
$ ssh root@192.168.1.64
The authenticity of host '192.168.1.64 (192.168.1.64)' can't be est
ablished. ECDSA key fingerprint is SHA256:gQQ9QzBGV0F0fZCmP5qLxioRk
bPlRqJDLnLuklLZVhQ. Are you sure you want to continue connecting (y
es/no)? yes Warning: Permanently added '192.168.1.64' (ECDSA) to th
e list of known hosts. root@192.168.1.64's
password:
Last login: Fri May 6 20:47:14 2016 from 192.168.1.54 Welcome to

      **
      **
    ** **
    ** **      ****      **** ** **      *****
**          ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** **
**          ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** **
*****      ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** **
**          ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** **
**          ** ** **          ** ** **          *****

For further information check:
http://www.artila.com/

[root@Matrix700 ~]#
```

5. Check Linux Kernel Version

```
[root@matrix700 ~]#uname -a
Linux matrix700 4.9.18-yocto-standard #1 Sun Mar 26 23:00:05 CST 20
17 armv7l armv7l armv7l GNU/Linux

[root@matrix700 ~]#uname -v
#1 Sun Mar 26 23:00:05 CST 2017

[root@matrix700 ~]#uname -r
4.9.18-yocto-standard

[root@matrix700 ~]#
```

6. File System Information

The M-A5D35 based Matrix IoT gateways come with 8GB on-board EMMC Flash memory, which contains boot loader, Linux kernel, root file system and user disk (/home).

```
[root@Matrix700 ~]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
mmcblk0             179:0    0   7.3G  0 disk
`--mmcblk0p1        179:1    0   7.3G  0 part /
mtdblock0           31:0    0   8.3M  0 disk
mtdblock1           31:1    0   8.2M  0 disk
mtdblock2           31:2    0   7.7M  0 disk
mtdblock3           31:3    0   7.7M  0 disk
mtdblock4           31:4    0   7.6M  0 disk
mtdblock5           31:5    0   3.9M  0 disk

[root@Matrix700 ~]#
```

```
[root@Matrix700 /]# ls -F
bin/   dev/   home/  lost+found/  mnt/   run/   sys/   usr/
boot/  etc/   lib/   media/       proc/  sbin/  tmp@   var/

[root@Matrix700 /]#
```

```
[root@Matrix700 ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       7.1G  252M  6.5G   4% /
devtmpfs        251M    0  251M   0% /dev
tmpfs           251M  72K  251M   1% /run
tmpfs           251M  104K  251M   1% /var/volatile

[root@Matrix700 ~]#
```

7. Serial Port Settings

7.1 Port Mapping

The M-A5D35 based Matrix IoT gateways come with four or eight serial communication ports. The first four serial ports are CPU native serial ports. Some M-A5D35 based Matrix IoT gateways provide more serial ports via USB-to-Serial chip. The serial port mapping information is listed below:

Port 1 ~ 4: for Matrix-7xx series, Port 5 ~ 8: only for Matrix-710

Port 1 → /dev/ttyS1

Port 2 → /dev/ttyS2

Port 3 → /dev/ttyS3

Port 4 → /dev/ttyS4

Port 5 → /dev/ttyUSB0

Port 6 → /dev/ttyUSB1

Port 7 → /dev/ttyUSB2

Port 8 → /dev/ttyUSB3

7.2 Configure the Serial Port

Please use the built-in **setuart** utility to display/modify the operation mode and communication parameters of the first four serial ports (ttyS1/2/3/4).

```
[root@Matrix700 ~]#setuart -h
Artila utility: setuart
Usage: setuart [OPTION]

-h          display this help and exit
-v          print version number and exit
-p          uart port number
-t          uart interface type [232,485]
-b          set baudrate, up to 921600bps

Examples:
  setuart -p 1                display port 1 type and baudrate
  setuart -p 1 -t 485 -b 115200  set port 1 type RS-485 and baud to 115200
  setuart -p 1 -t 232 -b 9600    set port 1 type to RS-232 and baud to 9600
```

⚠ Caution

The serial port's mode and associated communication parameters will go back to factory default after system reboot.

For Matrix-710

The serial port P8 default operation mode is RS-485. Please use the built-in **cfguart** utility to show/change serial port's operation mode to RS-232.

```
[root@matrix710 ~]#cfguart
opening port 0
ftHandle0 = 0x496b8
** Current interface is RS485. **

[root@matrix710 ~]#cfguart 232
opening port 0
ftHandle0 = 0x496b8
** Set interface to 232 **

[root@matrix710 ~]#cfguart
opening port 0
ftHandle0 = 0x496b8
** Current interface is RS232. **

[root@matrix710 ~]#cfguart 485
opening port 0
ftHandle0 = 0x496b8
** Set interface to 485 **
```

⚠ Caution

The serial port's mode will go back to RS-232 mode after system reboot.

8. System Time and Real-Time Clock(RTC)

8.1 Adjust System Time by date Command

The M-A5D35 based Matrix IoT gateways support standard **date** command to adjust the Linux system time manually. A typical usage is: **date MMDDhhmmYYYY**.

```
[root@Matrix700 ~]#date 050717132016
Sat May 7 17:13:00 UTC 2016

[root@Matrix700 ~]#
```

8.2 Adjust RTC by hwclock Command

To adjust the on-board Real-time clock (RTC), please follow the steps shown below: First, to adjust the system time by using the **date** command. Then use the **hwclock** command to synchronize the system time to the RTC.

A typical usage is: **hwclock -w**.

```
[root@Matrix700 ~]#hwclock
Thu May 26 15:31:49 2016  0.000000 seconds

[root@Matrix700 ~]#date
Thu May 26 15:32:00 UTC 2016

[root@Matrix700 ~]#hwclock -w

[root@Matrix700 ~]#
```

8.3 Synchronize System Time by NTP Server

8.3.1 Install the ntpdate utility

The M-A5D35 based Matrix IoT gateways support the **ntpdate** NTP client utility to synchronize the system date with specified NTP server. Users need to install the **ntpdate** utility first by executing the **apt-get install ntpdate** command.

```
[root@matrix700 ~]#apt-get install ntpdate
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
 ntpdate
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/41.0 kB of archives.
After this operation, 0 B of additional disk space will be used.
Selecting previously unselected package ntpdate.
(Reading database ... 17344 files and directories currently installed.)
Preparing to unpack .../ntpdate_4.2.8p9-r0_armhf.deb ...
Unpacking ntpdate (4.2.8p9-r0) ...
Setting up ntpdate (4.2.8p9-r0) ...

[root@matrix700 ~]#
```

8.3.2 Using the ntpdate utility

The following example shows how to use the *ntpdate* utility to synchronize the system with the NTP server **0.pool.ntp.org**.

```
[root@matrix700 ~]#date
Mon Apr 10 07:17:31 UTC 2017

[root@matrix700 ~]#date 050717132016
Sat May 7 17:13:00 UTC 2016

[root@matrix700 ~]#ntpdate 0.pool.ntp.org
10 Apr 07:18:36 ntpdate[1025]: step time server 61.216.153.106 offset
29167497.848661 sec

[root@matrix700 ~]#date
Mon Apr 10 07:18:45 UTC 2017

[root@matrix700 ~]#date;hwclock
Mon Apr 10 07:18:59 UTC 2017
Mon Apr 10 07:18:58 2017 0.000000 seconds

[root@matrix700 ~]#hwclock -w

[root@matrix700 ~]#date;hwclock
Mon Apr 10 07:19:15 UTC 2017
Mon Apr 10 07:19:15 2017 0.000000 seconds

[root@matrix700 ~]#
```

9. Insert Kernel Modules

Users can use command **lsmod** to list all installed kernel modules.

```
[root@matrix700 rc5.d]#lsmod
Module                Size Used by
usb_f_mass_storage    25809  2
usb_f_acm              4064   2
u_serial              7750   3 usb_f_acm
libcomposite          33643  12 usb_f_acm,usb_f_mass_storage
nfsd                  251055  11
auth_rpcgss           39359   1 nfsd
oid_registry          2441   1 auth_rpcgss
exportfs              3541   1 nfsd
nfs_acl               2510   1 nfsd
lockd                 53405   1 nfsd
grace                 1627   2 nfsd,lockd
sunrpc                175725  16 auth_rpcgss,nfsd,nfs_acl,lockd
atmel_usba_udc        15098   0
udc_core              10846   5 usb_f_acm,usb_f_mass_storage,atmel_usb
a_udc,u_serial,libcomposite
```

To load additional kernel modules during the system boot-up, you can modify the file:
/etc/modules.

```
[root@Matrix700 ~]#cat /etc/modules
atmel_usba_udc
#g_serial
#mt7601Usta

[root@Matrix700 ~]#
```

10. Insert Software Package

The M-A5D35 based Matrix IoT gateways support standard **apt** (Advanced Package Tool) package management utility. With this utility, users can easily install, upgrade, remove software packages. Artila provides a self-maintained software repository. The apt configuration file path is **/etc/apt/sources.list**.

```
[root@Matrix700 ~]#ls /etc/apt
apt.conf apt.conf.d preferences.d sources.list sources.list.d

[root@Matrix700 ~]#cat /etc/apt/sources.list
deb [trusted=yes] http://www.artila.com/download/A5D35/Linux/deb/cortexa5hf-vfp cortexa5hf-vfp main
deb [trusted=yes] http://www.artila.com/download/A5D35/Linux/deb/all all main
deb [trusted=yes] http://www.artila.com/download/A5D35/Linux/deb/matrix700 matrix700 main
```

** Please be noted the last line of the **/etc/apt/sources.list** varies according to specific model name.

Commonly used apt commands are listed below:

- **apt-get install** <package> to install package
- **apt-get remove** <package> to remove package
- **apt-cache search** <package> to search package
- **apt-get update** to update the package list
- **apt-get upgrade** to upgrade installed packages

11. Mount/Unmount an SD Card

The M-A5D35 based Matrix IoT gateways support SD card access. If an SD card is inserted, you can use **lsblk** command to find the device identifier name. And then use **mount** command to mount the SD card to a folder.

Before SD Insertion

```
[root@Matrix700 ~]#lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
mmcblk0              179:0    0   7.3G  0 disk
  `--mmcblk0p1 179:1    0   7.3G  0 part /
mtdblock0            31:0    0   8.3M  0 disk
mtdblock1            31:1    0   8.2M  0 disk
mtdblock2            31:2    0   7.7M  0 disk
mtdblock3            31:3    0   7.7M  0 disk
mtdblock4            31:4    0   7.6M  0 disk
mtdblock5            31:5    0   3.9M  0 disk
```

After SD Insertion

```
[root@Matrix700 ~]#lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
mmcblk0              179:0    0   7.3G  0 disk
  `--mmcblk0p1 179:1    0   7.3G  0 part /
  mmcblk1             179:24   0   1.9G  0 disk
mtdblock0            31:0    0   8.3M  0 disk
mtdblock1            31:1    0   8.2M  0 disk
mtdblock2            31:2    0   7.7M  0 disk
mtdblock3            31:3    0   7.7M  0 disk
mtdblock4            31:4    0   7.6M  0 disk
mtdblock5            31:5    0   3.9M  0 disk
```

Mount mmcblk1 to /media.

```
[root@Matrix700 ~]#mount /dev/mmcblk1 /media
[root@Matrix700 ~]#lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
mmcblk0              179:0    0   7.3G  0 disk
  `--mmcblk0p1 179:1    0   7.3G  0 part /
  mmcblk1             179:24   0   1.9G  0 disk /media
mtdblock0            31:0    0   8.3M  0 disk
mtdblock1            31:1    0   8.2M  0 disk
mtdblock2            31:2    0   7.7M  0 disk
mtdblock3            31:3    0   7.7M  0 disk
mtdblock4            31:4    0   7.6M  0 disk
mtdblock5            31:5    0   3.9M  0 disk
```

Unmount /media.

```
[root@Matrix700 ~]#umount /media
```

12. Mount/Unmount a USB Card

The M-A5D35 based Matrix IoT gateways support generic USB drives. If an USB drive is inserted, you can use **lsblk** command to find the device identifier name. And then use **mount** command to mount the USB drive to a folder.

Before USB drive Insertion

```
[root@Matrix700 ~]#lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
mmcblk0             179:0    0   7.3G 0 disk
├─mmcblk0p1         179:1    0   7.3G 0 part /
mtdblock0           31:0    0   8.3M 0 disk
mtdblock1           31:1    0   8.2M 0 disk
mtdblock2           31:2    0   7.7M 0 disk
mtdblock3           31:3    0   7.7M 0 disk
mtdblock4           31:4    0   7.6M 0 disk
mtdblock5           31:5    0   3.9M 0 disk
```

After USB drive Insertion

```
[root@Matrix700 ~]#lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                 8:0      1  14.5G 0 disk
├─sda1              8:1      1  14.5G 0 part
mmcblk0             179:0    0   7.3G 0 disk
├─mmcblk0p1         179:1    0   7.3G 0 part /
mtdblock0           31:0    0   8.3M 0 disk
mtdblock1           31:1    0   8.2M 0 disk
mtdblock2           31:2    0   7.7M 0 disk
mtdblock3           31:3    0   7.7M 0 disk
mtdblock4           31:4    0   7.6M 0 disk
mtdblock5           31:5    0   3.9M 0 disk
```

Mount sda1 to /media.

```
[root@Matrix700 ~]#mount /dev/sda1 /media
[root@Matrix700 ~]#lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda                 8:0      1  14.5G 0 disk
├─sda1              8:1      1  14.5G 0 part /media
mmcblk0             179:0    0   7.3G 0 disk
├─mmcblk0p1         179:1    0   7.3G 0 part /
mtdblock0           31:0    0   8.3M 0 disk
mtdblock1           31:1    0   8.2M 0 disk
mtdblock2           31:2    0   7.7M 0 disk
mtdblock3           31:3    0   7.7M 0 disk
mtdblock4           31:4    0   7.6M 0 disk
mtdblock5           31:5    0   3.9M 0 disk
```

Unmount /media.

```
[root@Matrix700 ~]#umount /media
```

13. Web Server Settings

13.1 Nginx Web Server

The M-A5D35 based Matrix IoT gateways come with pre-installed **nginx** web server. The configuration file is **/etc/nginx/nginx.conf**.



13.2 Root Web Page Directory

The default root web page directory is **/var/www/localhost/html**. This path can be changed by modifying the above configuration file.

```
[root@Matrix700 ~]#ls /var/www/localhost/html
50x.html  index.html

[root@Matrix700 ~]#
```

13.3 PHP Support

The M-A5D35 based Matrix IoT gateways support commonly used server-side script languages, including Perl, PHP and Python. Per and Python support are built-in, while PHP support needs to be installed manually using **apt-get** command.

- apt-get install **php-cli**
- apt-get install **php-cgi**
- apt-get install **php-fpm**

```
[root@matrix700 ~]#php-cgi -v
PHP 5.6.26 (cgi-fcgi) (built: Mar 19 2017 01:15:26)
Copyright (c) 1997-2016 The PHP Group
Zend Engine v2.6.0, Copyright (c) 1998-2016 Zend Technologies

[root@matrix700 ~]#php-fpm -v
PHP 5.6.26 (fpm-fcgi) (built: Mar 19 2017 01:15:19)
Copyright (c) 1997-2016 The PHP Group
Zend Engine v2.6.0, Copyright (c) 1998-2016 Zend Technologies

[root@matrix700 ~]#php -v
PHP 5.6.26 (cli) (built: Mar 19 2017 01:15:12)
Copyright (c) 1997-2016 The PHP Group
Zend Engine v2.6.0, Copyright (c) 1998-2016 Zend Technologies

[root@matrix700 ~]#
```


14. Auto-execute User Applications/Shell Scripts

14.1 Modify the */etc/rc5.d* directory

To automatically start user applications after system boot-up, please edit a shell script to execute the program, and put that script file to the folder: */etc/rc5.d*.

```
[root@matrix700 rc5.d]#ls
S01networking@   S19nfsccommon@   S60php-fpm@      S99stop-bootlogd@
S02dbus-1@       S20atd@           S90crond@        S99usb gadget@
S09sshd@         S20hwclock.sh@   S92nginx@        S99webmin@
S12rpcbind@      S20nfsserver@    S99readyled@
S15mountnfs.sh@ S20syslog@       S99rmnologin.sh@

[root@matrix700 rc5.d]#
```

14.2 Modify the */etc/profile*

To automatically start user shell scripts after system boot-up, please modify the */etc/profile* accordingly.

15. Change the Welcome Message

The welcome message file is `/etc/motd`, the default content is shown below, modify the content at your will.

```
[root@Matrix700 ~]#cat /etc/motd
Welcome to

      **                ** **
      **                ** **
     ** **            ** **
     ** **          ***** ** **          *****
**      **      **      **      **      **      **
**      **      **      **      **      **      *****
*****          **      **      **      **      **
**      **      **      **      **      **      **
**      **      **      **      **      **      *****

For further information check:
http://www.artila.com/

[root@Matrix700 ~]#
```

16. Reboot the System

To re-boot the system, use the ***reboot*** command.

```
[root@Matrix700 ~]#reboot  
Broadcast message from root@Matrix700 (ttyGS0) (Sun May 8 15:51:47 2  
016):  
The system is going down for reboot NOW!
```

17. User Application Development

17.1 Install C/C++ Cross Compilation Toolchain

The following instructions are based on **64-bit** Ubuntu Linux environment:

Step 1, Download the toolchain installation script from Artila's website, the URL is:

\$wget http://www.artila.com/download/sama5/linux/deb/sdk/poky-glibc-x86_64-meta-toolchain-cortexa5t2hf-vfp-m-a501-toolchain-3.1.1.sh

Step 2, Execute the toolchain installation script.

```
$ sh poky-glibc-x86_64-artila-full-cmdline-cortexa5hf-vfp-toolchain-3.1.1sh
```

Step 3, Activate the environment settings.

```
$ source /opt/poky/3.1.1/environment-setup-cortexa5hf-vfp-poky-linux-gnueabi
```

17.2 Using the C Cross Compiler

Step 1: Execute **\$CC** command to compile the C source file.

Step 2: Execute **scp** command to upload the compiled binary file to the Matrix IoT gateway.

```
$ cat hello.c
#include <stdio.h>

int main(){
printf("Hello World!\n");
return 0;
}

$ $CC -o hello_c hello.c

$ scp hello_c root@192.168.1.70:/home/root
root@192.168.1.70's password:
hello_c                               100% 9800      9.6KB/s   00:00

$
```

17.3 Using the C++ Cross Compiler

Step 1: Execute **\$CXX** command to compile the C++ source file.

Step 2: Execute **scp** command to upload the compiled binary to the Matrix IoT gateway.

```
$ cat hello.cpp
#include <iostream>
using namespace std;
int main() {
cout << "Hello! World!\n";
return 0;
}

$ $CXX -o hello_cpp hello.cpp

$ scp hello_cpp root@192.168.1.70:/home/root
root@192.168.1.70's password:
hello_cpp                                100%  11KB  10.9KB/s   00:0
0
$
```

17.4 Using the Native C Compiler

User application can also be directly developed on the M-A5D35 based Matrix IoT gateways. By default, **gcc** toolchain is pre-installed on the M-A5D35 based Matrix IoT gateways.

```
[root@Matrix700 ~]# cat hello.c
#include <stdio.h>
int main()
{
printf("Hello World!\n");
}

[root@Matrix700 ~]# gcc -o hello hello.c

[root@Matrix700 ~]# ./hello
Hello World!

[root@Matrix700 ~]#
```

17.5 Using the Native C++ Compiler

17.5.1 Install the Native C++ Toolchain

Users can install the native C++ toolchain via **apt-get** command. Two packages are needed to build a C++ source file, the **g++** package and the **g++-symlinks** package.

- apt-get install **g++ g++-symlinks**

17.5.2 Using the Native C++ Compiler

```
[root@Matrix700 ~]#cat hello.cpp
#include <iostream>
using namespace std;
int main() {
cout << "Hello World!\n";
return 0;
}

[root@Matrix700 ~]#g++ -o hello_cpp hello.cpp

[root@Matrix700 ~]#./hello_cpp
Hello World!

[root@Matrix700 ~]#
```

17.6 Using the Python Interpreter

17.6.1 Python 2 Support

The M-A5D35 based Matrix IoT gateways come with pre-built Python 2.7 interpreter.

```
[root@matrix700 ~]#python
Python 2.7.12 (default, Mar 19 2017, 00:10:10)
[GCC 6.2.0] on linux2
Type "help", "copyright", "credits" or "license" for more informatio
n.
>>>
```

The Python *pip* package manager is also included by default. To upgrade the pip itself, execute the following command:

- **pip install --upgrade pip**

17.6.2 Python 3 Support

Users can also install Python 3 via the **apt-get** command.

- apt-get install **python3-core**
- apt-get install **python3-pip**
- touch /user/lib/python3.5/site-packages/easy-install.pth
- **pip3** install --upgrade pip
-

```
[root@matrix700 ~]#python3
Python 3.5.2 (default, Mar 19 2017, 01:00:38)
[GCC 6.2.0] on linux
Type "help", "copyright", "credits" or "license" for more informatio
n.
>>>
```

-

```
[root@matrix700 site-packages]#pip list --format=columns
Package      Version
-----
pip          9.0.1
setuptools  22.0.5
```

18. GPIO Operation

The M-A5D35 SoM comes a bunch of GPIO (General Purpose IO) pins.

By implementation, all M-A5D35's GPIO pins are controlled in **user space**.

The M-A5D35 CPU provides five banks of GPIOs shown as below:

GPIO Bank	Bank A	Bank B	Bank C	Bank D	Bank E
GPIO label	PA0-31	PB0-31	PC0-31	PD0-31	PE0-31
GPIO chip	gpiochip0	gpiochip32	gpiochip64	gpiochip96	gpiochip128
GPIO number	0-31	31-63	64-95	96-127	128-159
GPIO mapping	pioA0-31	pioB0-31	pioC0-31	pioD0-31	pioE0-31

For example, the GPIO **PA31**, which is located on the pin43 of the CN1 connector, is mapped to number **31** ($31 = 0 + 31$); the GPIO **PD30**, which is located on the pin45 of the CN1 connector, is mapped to number **126** ($126 = 96 + 30$).

Example 1, to set PA31 as output:

```
[root@matrix700 ~]#cd /sys/class/gpio/

[root@matrix700 gpio]#ls
export  gpiochip0  gpiochip128  gpiochip32  gpiochip64  gpiochip96
unexport

[root@matrix700 gpio]#echo 31 > export

[root@matrix700 gpio]#ls
export      gpiochip128  gpiochip64  pioA31
gpiochip0  gpiochip32  gpiochip96  unexport

[root@matrix700 gpio]#cd pioA31

[root@matrix700 pioA31]#ls
active_low  device  direction  edge  power  subsystem  uevent  value

[root@matrix700 pioA31]#echo 'out' > direction

[root@matrix700 pioA31]#echo 1 > value

[root@matrix700 pioA31]#echo 0 > value

[root@matrix700 pioA31]#cd ..

[root@matrix700 gpio]#ls
export      gpiochip128  gpiochip64  pioA31
gpiochip0  gpiochip32  gpiochip96  unexport

[root@matrix700 gpio]#echo 31 > unexport

[root@matrix700 gpio]#ls
export  gpiochip0  gpiochip128  gpiochip32  gpiochip64  gpiochip96
unexport

[root@matrix700 gpio]#
```


Example 2, to set PD30 as output:

```
[root@matrix700 gpio]#pwd
/sys/class/gpio

[root@matrix700 gpio]#ls
export  gpiochip0  gpiochip128  gpiochip32  gpiochip64  gpiochip96
unexport

[root@matrix700 gpio]#echo 126 > export

[root@matrix700 gpio]#ls
export      gpiochip128  gpiochip64  pioD30
gpiochip0  gpiochip32  gpiochip96  unexport

[root@matrix700 gpio]#cd pioD30

[root@matrix700 pioD30]#ls
active_low  device  direction  edge  power  subsystem  uevent  value

[root@matrix700 pioD30]#echo 'out' > direction

[root@matrix700 pioD30]#echo 1 > value

[root@matrix700 pioD30]#echo 0 > value

[root@matrix700 pioD30]#cd ..

[root@matrix700 gpio]#ls
export      gpiochip128  gpiochip64  pioD30
gpiochip0  gpiochip32  gpiochip96  unexport

[root@matrix700 gpio]#echo 126 > unexport

[root@matrix700 gpio]#ls
export  gpiochip0  gpiochip128  gpiochip32  gpiochip64  gpiochip96
unexport

[root@matrix700 gpio]#
```

For more detailed information please refer to

<https://www.kernel.org/doc/Documentation/gpio/sysfs.txt>.

19. Install an USB Wi-Fi Dongle

The M-A5D35 based Matrix IoT gateways support USB Wi-Fi dongles. Current driver supports RT8192/RT5390 compatible hardware (e.g. [ASUS USB-N10 Nano Wireless-N](#) or WPER-172GN).

19.1 Install Hardware Driver

The USB Wi-Fi driver can be installed via apt-get utility.

```
[root@Matrix700 ~]#apt-get install kernel-module-rt18xxxu linux-firmware-rt18192cu
```

Install RT5390 driver via apt-get utility

```
[root@Matrix700 ~]#apt-get install kernel-module-rt2800usb linux-firmware-ralink
```

19.2 Modify the network interface configuration

The network interface configuration file path is `/etc/network/interfaces`. A typical configuration example is listed below:

```
# Wireless interfaces
auto wlan0
iface wlan0 inet dhcp
    wireless_mode managed
    wireless_essid any
    wpa-driver nl80211, wext
    wpa-conf /etc/wpa_supplicant.conf
```

Be noted the last line of the above example, which specifies an additional configuration file for WPA settings. In this example, the WPA configuration file path is `/etc/wpa_supplicant.conf`.

19.3 Modify the WPA configuration

Modify the `/etc/wpa_supplicant.conf` according to the Wi-Fi environment of your factory/office. A typical configuration example is listed below:

```
# WPA configuration

ctrl_interface=/var/run/wpa_supplicant
ctrl_interface_group=0
update_config=1
ap_scan=1

# WEP example
network={
    ssid="Artila"
    key_mgmt=NONE
    wep_key0=ABCABCABC
}

# WPA/WPA2 example
Network={
    ssid="Artila"
    key_mgmt=WPA-PSK
    auth_alg=OPEN
    psk="ABCABCABC"
}
```

19.4 Restart the wireless network interface

```
[root@Matrix700 ~]#ifdown wlan0

[root@Matrix700 ~]#ifup wlan0
```

20. Webmin Support

The M-A5D35 based Matrix IoT gateways support the Webmin, which is a browser_based system management tool.

To access the Webmin, please visit <https://192.168.2.127:10000>,

Username: admin

Password: admin



21. Setup Eclipse IDE

Users can integrate the M-A5D35 tool chain into the Eclipse IDE. It can be downloaded the Eclipse IDE for **C/C++ Developers (Luna)** from <https://www.eclipse.org/downloads/packages/release/Luna/SR2>

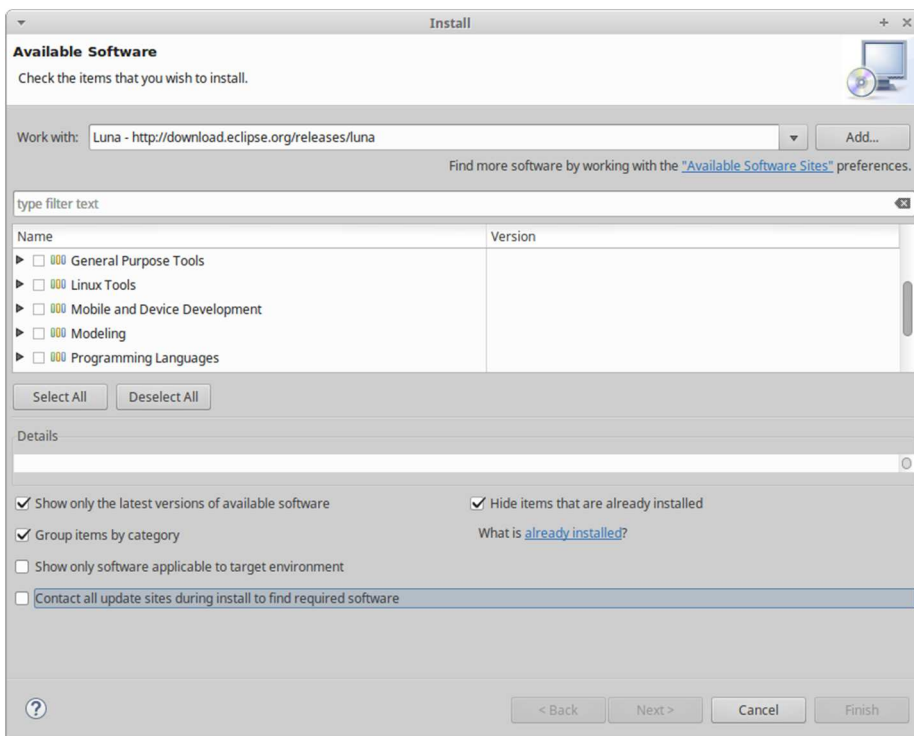
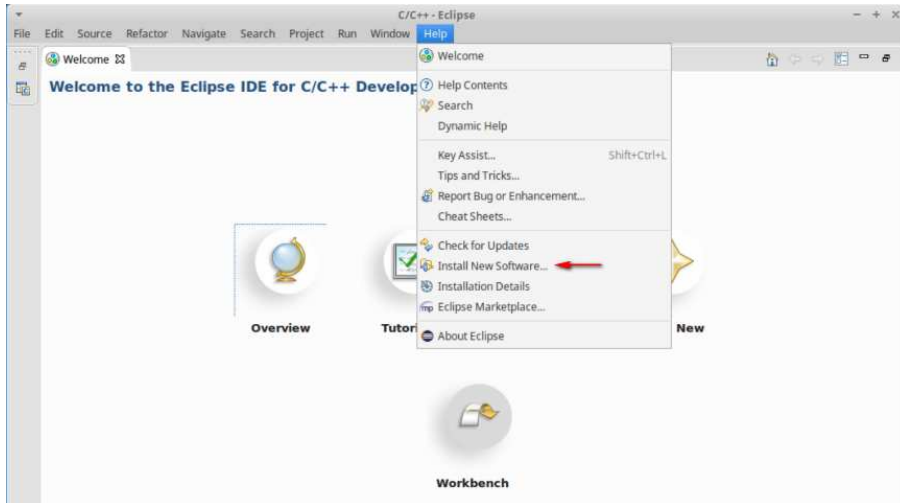
21.1 Configure the Eclipse IDE

Step 1, Start the Eclipse IDE.

Step 2, From “Help” menu select “Install New Software”

- Add “Luna - <http://download.eclipse.org/releases/luna>”.
- Select the following items (If these selections do not appear in the list, that means the items are already installed.)
 - Linux Tools
 - Linux Tools LTTng Tracer Control
 - Linux Tools LTTng Userspace Analysis
 - LTTng Kernel Analysis
 - Mobile and Device Development
 - C/C++ Remote Launch (Requires RSE Remote System Explorer)
 - Remote System Explorer End-user Runtime
 - Remote System Explorer User Actions
 - Target Management Terminal (Core SDK)
 - TCF Remote System Explorer add-in
 - TCF Target Explorer
 - Programming Languages
 - C/C++ Autotools Support
 - C/C++ Development Tools

Step3, Complete the installation and restart the Eclipse IDE



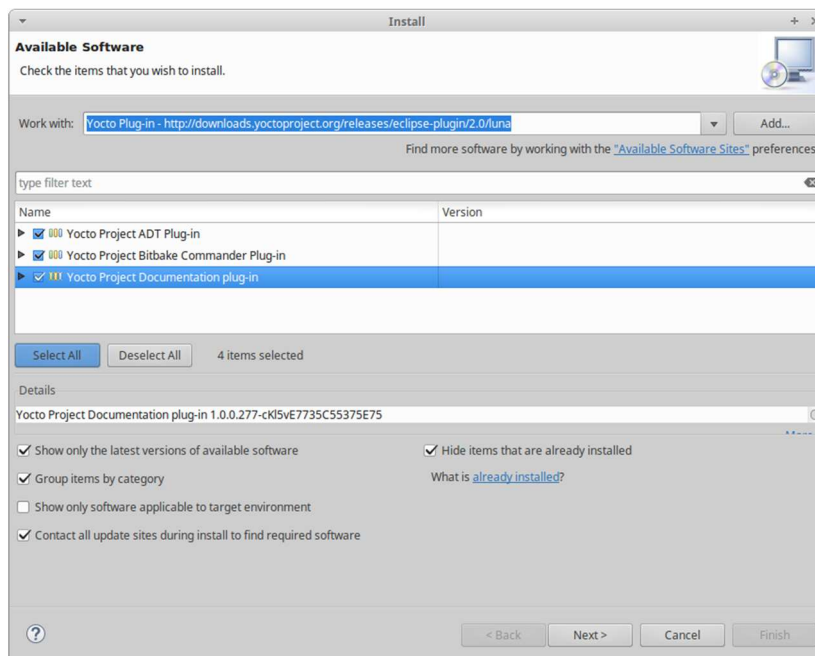
21.2 Install the Eclipse Yocto Plug-in

Please refer to following steps to install the Eclipse Yocto Plug-in:

Step1, From the “Help” menu select “Install New Software”

- Add URL “<http://downloads.yoctoproject.org/releases/eclipse-plugin/2.0/luna>” and provide a meaningful name.
- Select the following items
 - Yocto Project ADT Plug-in,
 - Yocto Project Bitbake Commander Plug-in
 - Yocto Project Documentation plug-in.

Step2, Complete the installation and restart the Eclipse IDE



21.3 Configuring the Cross-Compiler Options

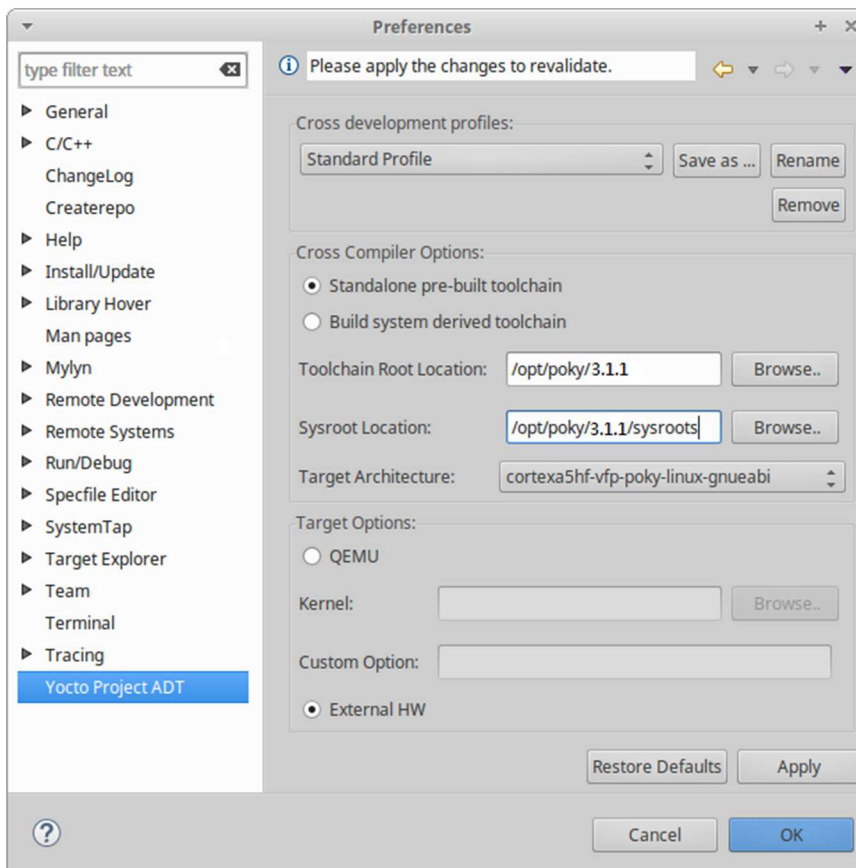
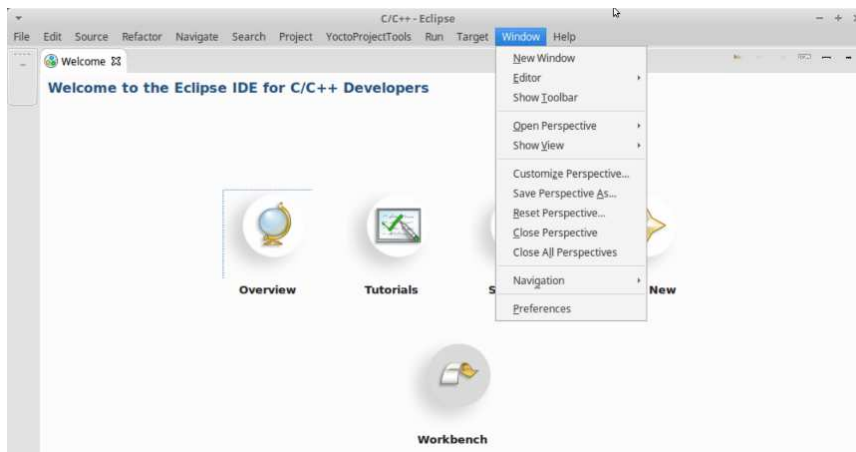
Please refer to following steps to config:

Step1, From the “Windows” menu select “Preferences”

Step2, Click "Yocto Project ADT" to display the configuration screen

Step3, Selecting the Toolchain Type: **Standalone pre-built toolchain**

- Point to the Toolchain: **/opt/poky/3.1.1**
- Specify the Sysroot Location: **/opt/poky/3.1.1/sysroots**
- Select the Target Architecture: **cortexa5hf-vfp-poky-linux-gnueabi**



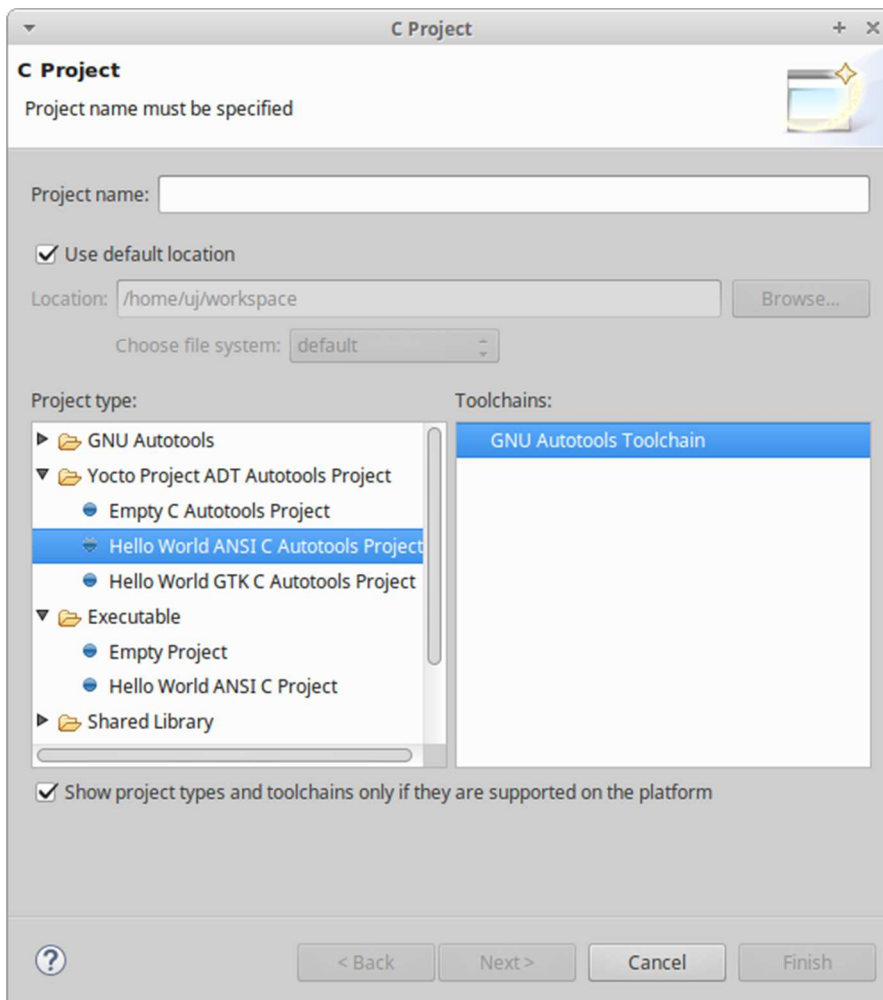
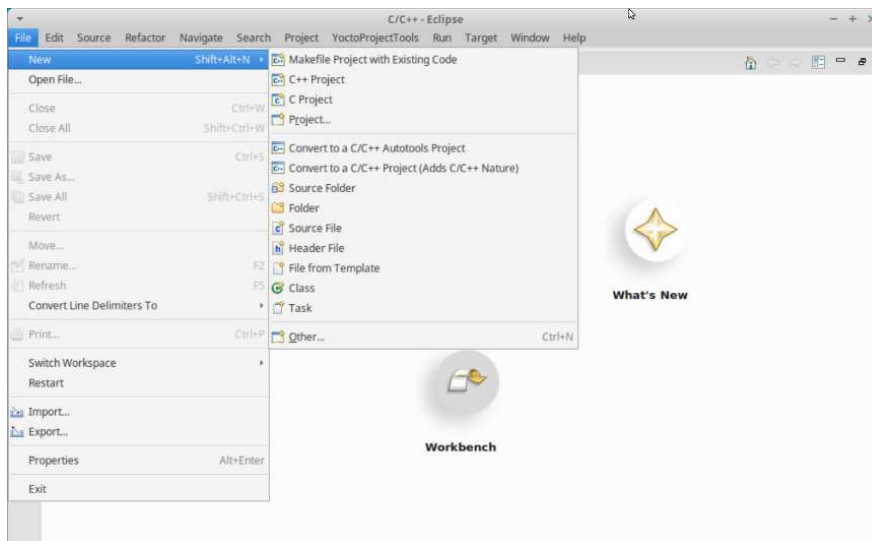
21.4 Create a Hello World Project

Step1, Select "Project" from the "File -> New" menu

- Double click C/C++
- Double click C Project to create the project
- Expand Yocto Project ADT Autotools Project
select Hello World ANSI C Autotools Project.
(This is an Autotools-based project based on a Yocto template)
- Put a name in the Project name field.
(Do not use hyphens as part of the name)
- Click "Next".
- Add information in the Author and Copyright notice fields.
- Click "Finish".

Step2, Right-click in the navigation pane and select "Reconfigure Project" from the pop-up menu. This selection reconfigures the project by running autogen.sh in the workspace for your project.

Step3, To build the project select "Build Project" from the "Project" menu.



22. Setup miniPCle (mPCle) Power

The M-A5D35 based Matrix IoT gateways, Model Matrix-710 & 713, support the miniPCle slot can be install LTE/4G/3G/WiFi module for wireless communication.

Following information shows how to turn on/off mPCle power (GPIO: PA2, PA3)

Example 1, Turn on of mPCle A (Matrix-713)

```
echo 2 > /sys/class/gpio/export
echo out > /sys/class/gpio/pcioA2/direction
echo 0 > /sys/class/gpio/pcioA2/value
```

Example 2, Turn off of mPCle A (Matrix-713)

```
echo 1 > /sys/class/gpio/pcioA2/value
```

23. Setup SIM card

The M-A5D35 based Matrix IoT gateways, Model Matrix-710 & 713, support the miniPCIe slot can be install LTE/4G/3G module for communication.

SIM Card setting is necessary before access by following:

NOTICE: Please unlock SIM PIN code first

Example: RYH2708

```
apt-get install kernel-module-cdc-acm kernel-module-pppoe kernel-module-ppp-mppe kernel-module-ppp-async ppp
```

configure /etc/ppp/peers/3g

```
/dev/ttyACM2 # modem port used
460800      # speed
defaultroute # use the cellular network for the default route
replacedefaultroute
noipdefault
usepeerdns  # use the DNS servers from the remote network
#nodetach   # keep pppd in the foreground
#nocrtscts  # hardware flow control
#lock       # lock the serial port
#noauth     # don't expect the modem to authenticate itself
#local      # don't use Carrier Detect or Data Terminal Ready
#persist
#demand
modem
#debug
# Use the next two lines if you receive the dreaded messages:
#
#   No response to n echo-requests
#   Serial link appears to be disconnected.
#   Connection terminated.
#
lcp-echo-failure 4
lcp-echo-interval 65535
connect      "chat -v -f /etc/ppp/chats/connect"
disconnect   "chat -v -f /etc/ppp/chats/disconnect"
```

/etc/ppp/chats/connect

```

TIMEOUT 10
ABORT 'BUSY'
ABORT 'NO ANSWER'
# ABORT 'ERROR'
SAY 'Starting 3G connect script\n'

#"" 'AT+CPIN="0000"'

# Get the modem's attention and reset it.

"" 'ATZ'

# E0=No echo, V1=English result codes
#OK 'ATQ0 V1 E1 S0=0 &C1 &C2 +FCLASS=0'
#OK 'ATQ0 V1 E1 &C1 &C2 '
OK 'ATQ0'

# Set Access Point Name (APN)
OK 'AT+CGDCONT=1,"IP","internet"'

# Dial the number
ABORT 'NO CARRIER'
SAY 'Dialing...\n'
OK 'ATDT*99#'

CONNECT ''

```

/etc/ppp/chats/disconnect

```

"" '\K'
"" '+++ATH0'
SAY "GPRS disconnected."

```

/etc/network/interfaces

```

# 3G PPP interface
#
# Example of a 3G ppp connection
#
auto ppp0
iface ppp0 inet ppp
provider 3g

```

Matrix-713 support Dual micro-SIM (GPIO: PA21) which Support cross-zone communication / seamless integration or dual-SIM for dual-LTE/4G/3G mPCle module.

Dual SIM Ctrl

Import notes:

If config is change which restart the mPCle power after.

Default: *

SIM_SW	mPCle A	mPCle B
*0	SIM A	SIM B
1	SIM A	NA

Example: Turn on and set 0

```
echo 21 > /sys/class/gpio/export  
echo out > /sys/class/gpio/pioA2/direction  
echo 0 > /sys/class/gpio/pioA2/value
```

24. Setup “STATUS” LED indicator

The M-A5D35 based Matrix IoT gateways, Model Matrix-713, support three “STATUS” LED indicators for user definition control by GPIO: A22, A23, A26

Example 1, Enable and turn on the LED1

```
echo 22 > /sys/class/gpio/export
echo out > /sys/class/gpio/gpioA22/direction
echo 0 > /sys/class/gpio/gpioA22/value
```

Example 2, Turn off and disable the LED1

```
echo 1 > /sys/class/gpio/gpioA22/value
echo 22 > /sys/class/gpio/unexport
```


25. Setup Digital Input / Digital Output

The M-A5D35 based Matrix IoT gateways, Model Matrix-713, support 2x digital input and 2x digital output port.

DI / Digital Input (GPIO: D7, D21)

DO / Digital Output (GPIO: D19, D20)

Example 1, Read value of DI1

```
cat /gpio/DI1/value
```

Example 2, Set High of DO1

```
echo 0 > /gpio/DO1/value
```

26. Setup GNSS & IMU

The M-A5D35 based Matrix IoT gateways, Model Matrix-713, support GNSS and IMU as following:

GNSS (Global Navigation Satellite System)

- Support Dual Satellite: GPS & GLONASS

Command: `gpsmon`

IMU (Inertial Measurement Unit)

- 1 x 3-Axis digital output Gyroscope

Command: `iio_info`

- 1 x 3-Axis Accelerometer (G-Sensor)

Command: `iio_info`

- 1 x 3-Axis Magnetometer (E-Compass)

Command: `apt-get install i2c-tools`

`i2cdump -y 3 0x0c`

27. Setup Audio Out

The M-A5D35 based Matrix IoT gateways, Model Matrix-713, support one Audio out as line-out R/L port, optional earphone R/L.

Command: `aplay` (support format type: `voc`, `wav`, `raw` or `au`)

Example: `aplay sample.wav`

28. Restore to Factory Default

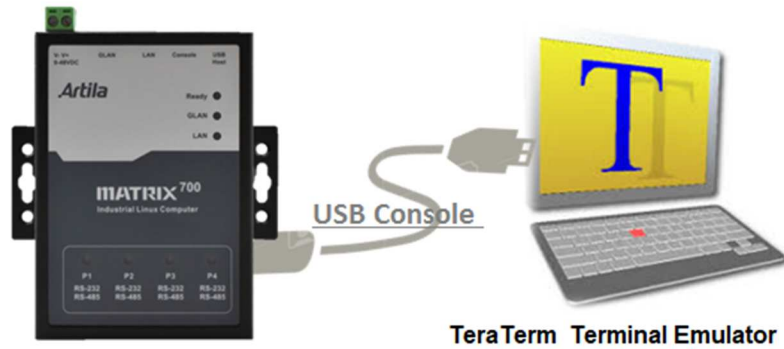
The following information shows how to restore to factory default:

Step 1,

Power-on Matrix-700 (“Ready” LED in green)

After system ready (“Ready” LED turns to yellow),

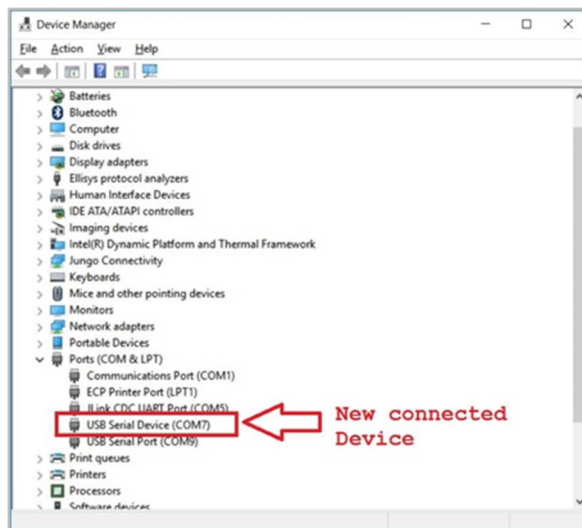
Connect the Matrix-700’s USB/serial console to a PC



Step 2,

It shows *USB Serial Device “COMx”* at Device Manager / PC Windows:

X : depends on your PC's environment



Communicate via this Serial device (EX. By TeraTerm @115200,N81).

Step 3,

Execute the command “**restore factory**” and “**y**”, this command will force Matrix-700/710/713 to boot again to start the restore process.

```
# restore factory
Restore from factory, Sure?(y/n)
y
.....
```

The **READY** LED will be blinking during the restore process

This process may take 20~30minutes.

USB/serial console is NOT accessible during the restore process.



After completed above steps, the Matrix-700/710/713 will automatically reboot again when the backup process is finished.

29. Backup the “File System” of Matrix-700/710/713

The following information shows how to restore to factory default:

Step 1,

Insert an empty 16GB(or above) USB drive (FAT32 format). Let’s say the USB drive maps to `/dev/sda1`.

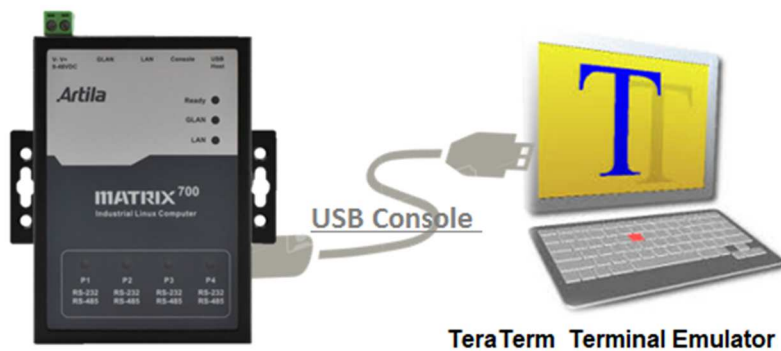


Step 2,

Power-on Matrix-700/710/713 (“Ready” LED in green)

After system ready (“Ready” LED turns to yellow),

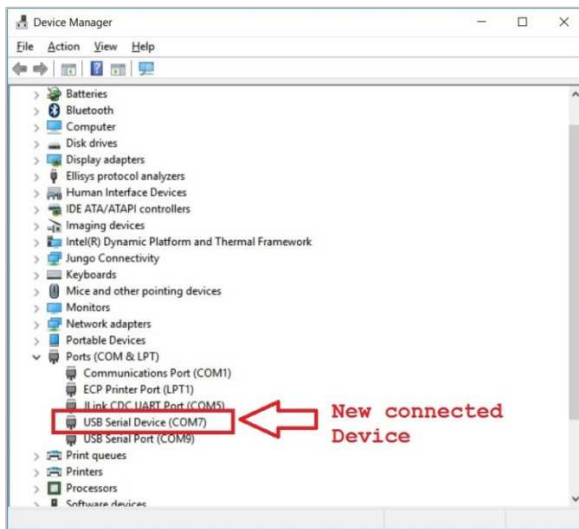
Connect the Matrix-700/710/713’s USB/serial console to a PC



Step 3,

It shows USB Serial Device "COMx" at Device Manager / PC Windows:

X : depends on your PC's environment



Communicate via this Serial device (EX. By TeraTerm @115200,N81).

Step 4,

Execute the command "**backup /dev/sda1**" and "**y**", this command will force Matrix-700/710/713 to boot again to start the backup process.

```
# backup /dev/sda1
Backup to /dev/sda1, Sure?(y/n)
y
.....
```

The **READY** LED will be blinking during the backup process

This process may take **20~30** minutes

USB/serial console is NOT accessible during the backup process



After completed above steps, the Matrix-700/710/713 will automatically reboot again when the backup process is finished.

Remove the USB drive and keep it properly as the golden copy.

30. Restore the “File System” of Matrix700/710/713

The following information shows how to restore to factory default:

Step 1,

Insert the golden copy USB drive. Let's say the USB drive maps to `/dev/sda1`.

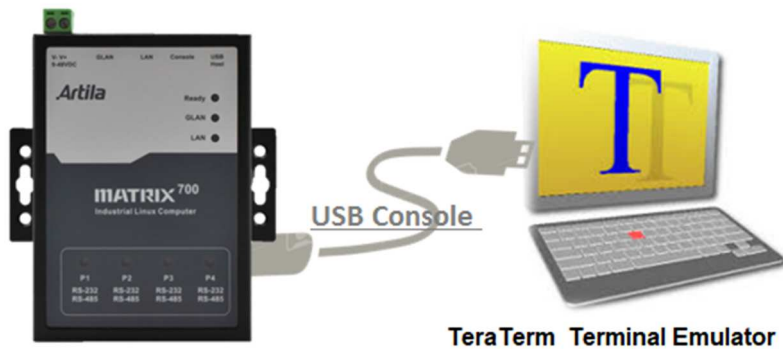


Step 2,

Power-on Matrix-700/710/713 (“Ready” LED in green)

After system ready (“Ready” LED turns to yellow)

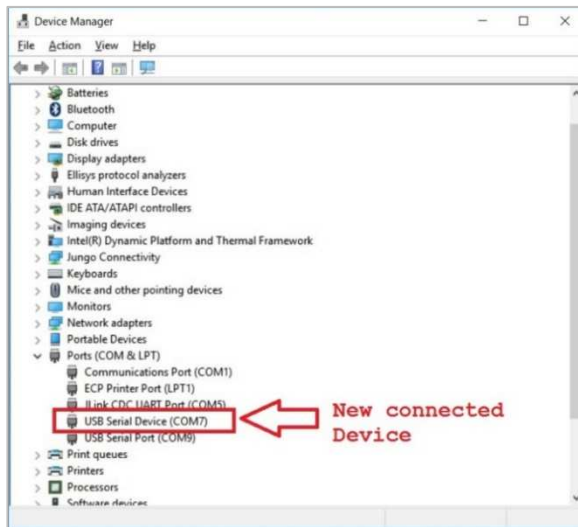
Connect the Matrix-700/710/713's USB/serial console to a PC



Step 3,

It shows USB Serial Device "COMx" at Device Manager / PC Windows:

X : depends on your PC's environment



Communicate via this Serial device (EX. By TeraTerm @115200,N81).

Step 4,

Execute the command "**restore /dev/sda1**" and "**y**", this command will force Matrix-700/710/713 to boot again to start the restore process.

```
# restore /dev/sda1
Restore from /dev/sda1, Sure?(y/n)
y
.....
```

The **READY** LED will be blinking during the restore process,

This process may take 20~30minutes

USB/serial console is NOT accessible during the restore process.



After completed above steps, the Matrix-700/710/713 will automatically reboot again when the restore process is finished.

Remove the USB drive.