12/13/20 04:49:00 Z:_shares1\psystems\PBM4Y-RASPB\BM4Y-LR1-LinuxRPi.yaml

BM4Y-LR1
OpiBerry Embedded ARM Cortex-A72/A53
Software Guide

BM4Y-LR1 Software Guide

---

Trademarks
The Opiware logo is a registered trademark of Opiware Solutions. 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 Opiware.

Opiware provides this document as is, without warranty of any kind, either expressed or implied, including, but not limited to, its particular purpose. Opiware 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, Opiware 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

BM4Y-LR1 Software Guide

---

Document Amendment History

Revision	Date	Remark
V 0.1	2020 Nov.	Initial Doc

BM4Y-LR1 Software Guide

---

Table of Contents

BM4Y-LR1 Software Guide

----

1. Overview

This software guide applies to Opiware’s BM4Y/CM3Y series for Industrial Applications.

Operation System
- Windows IoT / Linux / OpenWRT / RT-Thread
- Supports bootup from SSD or SD card
- Support Backup/Restore from SD card/USB
- Boot Loader : Pi Bootloader
- File System : ETX4
    - RPi uses ETX4 file system for SSD drive.
    - files system is stored in SSD memory.

Software Development
- Toolchain: gcc + glibc/uclibc/musl
- Supports remote C/C++ compilation
- Package Management

Package Management
- Package repository: Opiware self-maintained repository
- Command: Standard apt, apt-get

Popular Packages
- Web server: Apache/Lighttpd
- Database: MySQL/SQLite3
- Scripting: PHP/Node.JS/Node-RED
- Text editor: vim/nano/sed
- Administration: OpiWebUI

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 (BM4Y/CM3Y debug port)
⍰ ttyS1~ttyS4: serial ports (BM4Y/CM3Y 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
BM4Y/CM3Y uses standard I/O device control to access following devices:
⍰ Ethernet: eth0
⍰ 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
• ssh Login: root
• Password: root23
• Terminal type: VT100

2. Access the USB Serial Console

2.1 USB Serial Console Introduction
All the BM4Y/CM3Y platforms 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 hardware platform to a Desktop/Notebook PC. The hardware platform can
be directly driven by USB power.
When the hardware platform finished its boot up process, it will automatically
emulate an USB CDC/ACM compatible serial device.
-------------------------------------------------------------------------------
BM4Y/CM3Y                    | Use a standard on-the   | Linux/Windows/OSX
hardware platform            | shelf USB-to-MicroUSB   | Desktop/Notebook PC
```

```
comes with a USB client port | cable to connect to the |
                             | hardware board          |
-------------------------------------------------------------------------------
image1                       | image1                  | image1
-------------------------------------------------------------------------------
```

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 hardware platform’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
Opiware web site. (http://www.opiware.com/download/BM4Y/Linux/toolchain/linux-cdc-acm.inf).

2.2 USB Serial Console Log-in

User name: root
Password: root123

Following example by PI4BX-BRD1 board

```
Welcome to Opiware

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

Poky (Yocto Project Reference Distro) 2.2 BM4Y /dev/ttyGS0
BM4Y login: root
Password:
```

3. Network Settings

3.1 Config the Network Interface

The BM4Y/CM3Y platforms come one Ethernet ports, the default network settings are shown below:

Ethernet Type	Port Label	Device mapping	IP mode	IP address
Gigabit	LAN	eth0	DHCP	auto

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@BM4Y:~]# 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
---------------------------------------
```

The following screen capture shows the eth0 of the PI4BX-BRD1 got a valid IP: 192.168.1.28.

```
---------------------------------------
[root@BM4Y:~]# ifdown eth0
[root@BM4Y:~]# ifup eth0
udhcpc: option -h NAME is deprecated, use -x hostname:NAME
udhcpc (v1.24.1) started
Sending discover...
Sending select for 192.168.1.28...
Lease of 192.168.1.28 obtained, lease time 86400
/etc/udhcpc.d/50default: Adding DNS 208.67.220.220
/etc/udhcpc.d/50default: Adding DNS 208.67.222.222
---------------------------------------
```

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@BM4Y:~]# cat /etc/resolv.conf

[root@BM4Y:~]# ifconfig eth0
eth0 Link encap:Ethernet HWaddr 00:13:48:03:08:4a
     inet addr:192.168.1.28 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@BM4Y:~]# ping google.com
ping: unknown host google.com

[root@BM4Y:~]# echo 'nameserver 8.8.8.8' > /etc/resolv.conf
```

```
[root@BM4Y:~]# cat /etc/resolv.conf
nameserver 8.8.8.8

[root@BM4Y:~]# 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 time
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
---------------------------------------
```

Warning: Please be noted that, the /etc/resolv.conf is physically located in the RAM disk, so the content of the file will disappear 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: root123

```
---------------------------------------
[root@BM4Y:~]# ssh root@192.168.1.28
The authenticity of host '192.168.1.28 (192.168.1.28)' can't be established. ECDSA
key fingerprint is SHA256:gQQ9QzBGV0F0fZCmP5qLxioRkbPlRqJDLnLuklLZVhQ. Are you sure
you want to continue connecting (yes/no)? yes Warning: Permanently added
'192.168.1.28' (ECDSA) to the list of known hosts. root@192.168.1.28's
password:
Last login: Fri May 6 20:47:14 2020 from 192.168.1.28 Welcome to Opiware

For further information check:
http://www.opiware.com/
---------------------------------------
```

5. Check Linux Kernel Version

```
---------------------------------------
[root@BM4Y:~]# uname -a
Linux BM4Y 4.9.18-yocto-standard #1 Sun Mar 26 23:00:05 CST 20
17 armv7l armv7l armv7l GNU/Linux

[root@BM4Y:~]# uname -v
#1 Sun Mar 26 23:00:05 CST 2020

[root@BM4Y:~]# uname -r
4.9.18-yocto-standard
---------------------------------------
```

6. File System Information

The BM4Y/CM3Y platforms come with pre-installed Linux OpiWebUI OS, which contains boot loader, Linux kernel, root file system and user disk (/home).

```
---------------------------------------
[root@BM4Y:~]# 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@BM4Y:/]# ls -F
bin/ dev/ home/ lost+found/ mnt/ run/ sys/ usr/
boot/ etc/ lib/ media/ proc/ sbin/ tmp@ var/

[root@BM4Y:~]# 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
---------------------------------------
```

7. Serial Port Settings

7.1 Port Mapping

The BM4Y/CM3Y platforms come with four or eight serial communication ports. The first four serial ports are CPU native serial ports. Some MBM4Y based hardware platforms provide more serial ports via USB-to-Serial chip.

The serial port mapping information is listed below:

```
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 (RS-232/485) and communication parameters of the first four serial ports (ttyS1/2/3/4).

```
---------------------------------------
[root@BM4Y:~]# setuart -h

Opiware 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.

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

8.1 Adjust System Time by data Command

The BM4Y/CM3Y platforms support standard date command to adjust the Linux system time manually. A typical usage is: date MMDDhhmmYYYY.

```
---------------------------------------
[root@BM4Y:~]# date 050717132020
Sat May 7 17:13:00 UTC 2020
---------------------------------------
```

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@BM4Y:~]# hwclock
Thu May 26 15:31:49 2020 0.000000 seconds

[root@BM4Y:~]# date
Thu May 26 15:32:00 UTC 2020

[root@BM4Y:~]# hwclock -w
---------------------------------------
```

8.3 Synchronize System Time by NTP Server

8.3.1 Install the ntpdate utility

The BM4Y/CM3Y platforms 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@BM4Y:~]# 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 install
ed.)
Preparing to unpack .../ntpdate_4.2.8p9-r0_armhf.deb ...
Unpacking ntpdate (4.2.8p9-r0) ...
Setting up ntpdate (4.2.8p9-r0) ...
---------------------------------------
```

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@BM4Y:~]# date
Mon Apr 10 07:17:31 UTC 2020

[root@BM4Y:~]# date 050717132020
Sat May 7 17:13:00 UTC 2020

[root@BM4Y:~]# ntpdate 0.pool.ntp.org
10 Apr 07:18:36 ntpdate[1025]: step time server 61.216.153.106 offs
et 29167497.848661 sec

[root@BM4Y:~]# date
Mon Apr 10 07:18:45 UTC 2020

[root@BM4Y:~]# date;hwclock
Mon Apr 10 07:18:59 UTC 2020
Mon Apr 10 07:18:58 2020 0.000000 seconds

[root@BM4Y:~]# hwclock -w

[root@BM4Y:~]# date;hwclock
Mon Apr 10 07:19:15 UTC 2020
Mon Apr 10 07:19:15 2020 0.000000 seconds
---------------------------------------
```

9. Insert Kernel Modules

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

```
---------------------------------------
[root@BM4Y:/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_
usba_udc,u_serial,libcomposite
----------------------------------------
```

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

```
----------------------------------------
[root@BM4Y:~]# cat /etc/modules
atmel_usba_udc
#g_serial
#mt7601Usta
----------------------------------------
```

10. Insert Software Package

The BM4Y/CM3Y platforms support standard apt (Advanced Package Tool) package management utility. With this utility, users can easily install, upgrade, remove software packages.Opiware provides a self-maintained software repository. The apt configuration file path is /etc/apt/sources.list.

```
----------------------------------------
[root@BM4Y:~]# ls /etc/apt
apt.conf apt.conf.d preferences.d sources.list sources.list.d

[root@BM4Y:~]# cat /etc/apt/sources.list
deb [trusted=yes] http://www.opiware.com/download/BM4Y/Linux/deb/co
rtexa5hf-vfp cortexa5hf-vfp main
deb [trusted=yes] http://www.opiware.com/download/BM4Y/Linux/deb/al
l all main
deb [trusted=yes] http://www.opiware.com/download/BM4Y/Linux/deb/ma
trix700 BM4Y 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 BM4Y/CM3Y platforms 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@BM4Y:~]# 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@BM4Y:~]# 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@BM4Y:~]# mount /dev/mmcblk1 /media
[root@BM4Y:~]# 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@BM4Y:~]# umount /media
---------------------------------------
```

12. Mount/Unmount a USB Card

The BM4Y/CM3Y platforms 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@BM4Y:~]# 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@BM4Y:~]# 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@BM4Y:~]# mount /dev/sda1 /media
[root@BM4Y:~]# 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@BM4Y:~]# umount /media
--------------------------------------
```

13. Web Server Settings 22

13. Web Server Settings

13.1 Nginx Web Server 22

13.1 Nginx Web Server

The BM4Y/CM3Y platforms come with pre-installed nginx web server.
The configuration file is /etc/nginx/nginx.conf.

13.2 Root Web Page Directory 22

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@BM4Y:~]# ls /var/www/localhost/html
50x.html index.html
[root@BM4Y:~]#
--------------------------------------
```

13.3 PHP Support 23

13.3 PHP Support

The BM4Y/CM3Y platforms 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@BM4Y:~]# php-cgi -v
PHP 5.6.26 (cgi-fcgi) (built: Mar 19 2020 01:15:26)
Copyright (c) 1997-2020 The PHP Group
Zend Engine v2.6.0, Copyright (c) 1998-2020 Zend Technologies

[root@BM4Y:~]# php-fpm -v
PHP 5.6.26 (fpm-fcgi) (built: Mar 19 2020 01:15:19)
Copyright (c) 1997-2020 The PHP Group
Zend Engine v2.6.0, Copyright (c) 1998-2020 Zend Technologies

[root@BM4Y:~]# php -v
PHP 5.6.26 (cli) (built: Mar 19 2020 01:15:12)
Copyright (c) 1997-2020 The PHP Group
Zend Engine v2.6.0, Copyright (c) 1998-2020 Zend Technologies

[root@BM4Y:~]#
```

14. Auto-execute User Applications/Shell Scripts 24

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@BM4Y:/rc5.d]# ls
S01networking@ S19nfscommon@ S60php-fpm@ S99stop-bootlogd
@
S02dbus-1@ S20atd@ S90crond@ S99usbgadget@
S09sshd@ S20hwclock.sh@ S92nginx@ S99webmin@
S12rpcbind@ S20nfsserver@ S99readyled@
S15mountnfs.sh@ S20syslog@ S99rmnologin.sh@

[root@BM4Y:/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@BM4Y:~]# cat /etc/motd
Welcome to
** ** **

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

[root@BM4Y:~]#
---------------------------------------
```

16. Reboot the System

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

```
---------------------------------------
[root@BM4Y:~]# reboot

Broadcast message from root@BM4Y (ttyGS0) (Sun May 8 15:51:47
2020):
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: Install and set up prerequisites for the toolchain:

```
[root@BM4Y:~]# sudo apt install build-essential git  ;Install native build tools and git
[root@BM4Y:~]# sudo mkdir -p /opt/raspi              ;Create 'raspi' folder in '/opt/'
```

Step 2: Download (clone) RPi's official cross compiling toolchain:

```
[root@BM4Y:~]# cd /opt/raspi
[root@BM4Y:~]# sudo git clone git://github.com/raspberrypi/tools.git

[root@BM4Y:~]# cd /opt/raspi/tools/arm-bcm2708      ;List the toolchain installed folders
```

There are four separate toolchain folders in here:

```
arm-bcm2708-linux-gnueabi
arm-bcm2708hardfp-linux-gnueabi
gcc-linaro-arm-linux-gnueabihf-raspbian      ;for 32-bit Linux host
gcc-linaro-arm-linux-gnueabihf-raspbian-x64  ;for 64-bit Linux host
```

17.2 Using the C Cross Compier

Step 1: Test the toolchain by print the gcc version

```
[root@BM4Y:~]# /opt/raspi/tools/arm-bcm2708/gcc-linaro-arm-linux-gnueabihf-raspbian/bin/arm-linux-gnueabihf-gcc -v
Output: gcc version 4.8.3 20140106 (prerelease) (crosstool-NG linaro-1.13.1-4.8-2014.01 - Linaro GCC 2013.11)
```

Step 2: Execute gcc command to compile the C source file.
Step 3: Execute scp command to upload the compiled binary file to the RPi.

```
-------------------------------------
[root@BM4Y:~]$ cat hello.c
#include <stdio.h>
int main() {
    printf("Hello World!\n");
    return 0;
}

[root@BM4Y:~]$ gcc -o hello_c hello.c
[root@BM4Y:~]$ scp hello_c user1@192.168.1.28:/home/user1
user1@192.168.1.28's password:
hello_c 100% 9800 9.6KB/s 00:00
-------------------------------------
```

17.3 Using the C++ Cross Compiler

Step 1: Execute g++ command to compile the C++ source file.
Step 2: Execute scp command to upload the compiled binary to the hardware platform.

```
-------------------------------------
[root@BM4Y:~]# cat hello.cpp
```

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

[root@BM4Y:~]# g++ -o hello_cpp hello.cpp
[root@BM4Y:~]# scp hello_cpp root@192.168.1.28:/home/root
root@192.168.1.28's password:
hello_cpp 100% 11KB 10.9KB/s 00:00
---------------------------------------
```

17.4 Using the Native C Compiler (ref:7gF9uK)

User application can also be directly developed on the BM4Y/CM3Y platforms. By default, gcc toolchain is pre-installed on the platforms.

```
---------------------------------------
[root@BM4Y:~]# cat hello.c
#include <stdio.h>
int main() {
    printf("Hello World!\n");
}

[root@BM4Y:~]# gcc -o hello hello.c

[root@BM4Y:~]# ./hello
Hello World!
---------------------------------------
```

17.5 Using the Native C++ Compiler (ref:7gF9uK)

17.5.1 Install the Native C++ Toolchain

By default, gcc toolchain is pre-installed on the platforms.

17.5.2 Using the Native C++ Compiler

```
---------------------------------------
[root@BM4Y:~]# cat hello.cpp
#include <iostream>
using namespace std;
int main() {
    cout << "Hello World!\n";
    return 0;
}

[root@BM4Y:~]# g++ -o hello_cpp hello.cpp

[root@BM4Y:~]# ./hello_cpp
Hello World!
---------------------------------------
```

18. GPIO Operation

The BM4Y/CM3Y series comes a bunch of GPIO (General Purpose IO) pins. By

implementation, all BM4Y/CM3Y’s GPIO pins are controlled in user space.
The BM4Y/CM3Y CPU provides five banks of GPIOs shown as below:

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@BM4Y:~]# cd /sys/class/gpio/

[root@BM4Y:/gpio]# ls
export gpiochip0 gpiochip128 gpiochip32 gpiochip64 gpiochip96
unexport

[root@BM4Y:/gpio]# echo 31 > export

[root@BM4Y:/gpio]# ls
export gpiochip128 gpiochip64 pioA31
gpiochip0 gpiochip32 gpiochip96 unexport

[root@BM4Y:/gpio]# cd pioA31

[root@BM4Y:/pioA31]# ls

active_low device direction edge power subsystem uevent value

[root@BM4Y:/pioA31]# echo 'out' > direction

[root@BM4Y:/pioA31]# echo 1 > value

[root@BM4Y:/pioA31]# echo 0 > value

[root@BM4Y:/pioA31]# cd ..

[root@BM4Y:/gpio]# ls
export gpiochip128 gpiochip64 pioA31
gpiochip0 gpiochip32 gpiochip96 unexport

[root@BM4Y:/gpio]# echo 31 > unexport

[root@BM4Y:/gpio]# ls
export gpiochip0 gpiochip128 gpiochip32 gpiochip64 gpiochip96
unexport

[root@BM4Y:/gpio]#
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
---------------------------------------
```

Example 2, to set PD30 as output:

```
--------------------------------------
[root@BM4Y:/gpio]# pwd
/sys/class/gpio

[root@BM4Y:/gpio]# ls
export gpiochip0 gpiochip128 gpiochip32 gpiochip64 gpiochip96
unexport

[root@BM4Y:/gpio]# echo 126 > export

[root@BM4Y:/gpio]# ls
export gpiochip128 gpiochip64 pioD30
gpiochip0 gpiochip32 gpiochip96 unexport

[root@BM4Y:/gpio]# cd pioD30

[root@BM4Y:/pioD30]# ls

active_low device direction edge power subsystem uevent value

[root@BM4Y:/pioD30]# echo 'out' > direction

[root@BM4Y:/pioD30]# echo 1 > value

[root@BM4Y:/pioD30]# echo 0 > value

[root@BM4Y:/pioD30]# cd ..

[root@BM4Y:/gpio]# ls
export gpiochip128 gpiochip64 pioD30
gpiochip0 gpiochip32 gpiochip96 unexport

[root@BM4Y:/gpio]# echo 126 > unexport

[root@BM4Y:/gpio]# ls
export gpiochip0 gpiochip128 gpiochip32 gpiochip64 gpiochip96
unexport
--------------------------------------
```

For more detailed information please refer to https://www.kernel.org/doc/Documentation/gpio/sysfs.txt.

19. Install an USB Wi-Fi Dongle

The BM4Y/CM3Y platforms 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@BM4Y:~]# apt-get install kernel-module-rtl8xxxu linux-fir
mware-rtl8192cu
---------------------------------------
```

Install RT5390 driver via apt-get utility

```
---------------------------------------
[root@BM4Y:~]# apt-get install kernel-module-rt2800usb linux-fi
rmware-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=”Opiware”
key_mgmt=NONE
wep_key0=ABCABCABC
}

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

```
---------------------------------------
```

19.4 Restart the wireless network interface

```
---------------------------------------
[root@BM4Y:~]# ifdown wlan0

[root@BM4Y:~]# ifup wlan0
---------------------------------------
```

20. Restore to Factory Default

The following information shows how to restore to factory default:

Step 1, Download the restore image:
The factory default system image can be download from
http://www.opiware.com/download/BM4Y/Linux/image/.
Be sure to download the image fitting to your product model (e.g.,
restore_BM4Y_20201208.img).

Step 2, Make a bootable SD Card:
Prepare a micro SD card (8GB capacity is recommend).
In Windows environment, users can take advantage of the free Windows utility
win32diskimager, which can be download from
https://sourceforge.net/projects/win32diskimager/.
In Linux environment, please use the ‘dd’ utility, an example usage is shown below:

```
sudo dd bs=1m if=restore_BM4Y_20201208.img of=/dev/sdc
```

Step 3, Execute the restore command:
Be noted that it must insert the bootable SD card in advanced before poweron the hardware platform, then login as root. Then execute the ‘restore’ command to start the restore process, which will take around 10 minutes to compete. The hardware platform will reboot automatically when the restore process is finished.

21. Webmin Support

The BM4Y/CM3Y platforms support the Webmin, which is a browser_based system management tool.
To access the Webmin, please visit https://192.168.1.28/settings
Username: admin
Password: admin123

22. Setup Eclipse IDE

Users can integrate the BM4Y/CM3Y 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

22.1 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 estart the Eclipse IDE

22.2 Install the Eclipse Yocto Plug-in ..................................... 39
22.2 Install the Eclipse Yocto Plug-in
Step1, From the “Help” menu select “Install New Software”
-> Add URL “http://downloads.yoctoproject.org/releases/eclipseplugin/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

22.3 Configuring the Cross-Compiler Options ................................ 40
22.3 Configuring the Cross-Compiler Options
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/2.0.2
-> Specify the Sysroot Location: /opt/poky/2.0.2/sysroots
-> Select the Target Architecture: cortexa5hf-vfp-poky-linux-gnueabi

22.4 Create a Hello World Project .......................................... 41
22.4 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.

23. Setup miniPCIe (mPCIe) Power

The BM4Y/CM3Y platforms, Model PBM4Y & CM3Y, support the miniPCIe slot can be install LTE/4G/3G/WiFi …. module for wireless communication.

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

Example 1, Turn on of mPCIe A (PBM4Y)

```
  echo 2 > /sys/class/gpio/export
  echo out > /sys/class/gpio/pioA2/direction
  echo 0 > /sys/class/gpio/pioA2/value
```

Example 2, Turn off of mPCIe A (PBM4Y)

```
  echo 1 > /sys/class/gpio/pioA2/value
```

24. Setup SIM card

The BM4Y/CM3Y platforms, Model PBM4Y & CM3Y, 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-pppmppe 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
---------------------------------------
```

PBM4Y support Dual micro-SIM (GPIO: PA21) which Support cross-zone communication / seamless integration or dual-SIM for dual-LTE/4G/3G mPCIe module.

```
Dual SIM Ctrl
  Import notes:
    If config is change which restart the mPCIe power after.

Default: *

    SIM_SW | mPCIe A | mPCIe 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
```

25. Setup “STATUS” LED indicator

The BM4Y/CM3Y platforms, Model PBM4Y, 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/pioA22/direction
  echo 0 > /sys/class/gpio/pioA22/value
```

Example 2, Turn off and disable the LED1

```
  echo 1 > /sys/class/gpio/pioA22/value
  echo 22 > /sys/class/gpio/unexport
```

26. Setup Digital Input / Digital Output

The BM4Y/CM3Y platforms, Model PBM4Y, 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
```

27. Setup GNSS & IMU

The BM4Y/CM3Y platforms, Model PBM4Y, 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
. x 3-Axis Magnetometer (E-Compass)

```
Command: apt-get install i2c-tools
    i2cdump -y 3 0x0c
```

28. Setup Audio Out
The BM4Y/CM3Y platforms, PBM4Y, 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