7inch HDMI LCD (B) User Manual

Description

7 inch Capacitive Touch Screen LCD, HDMI interface, supports various systems.

Features

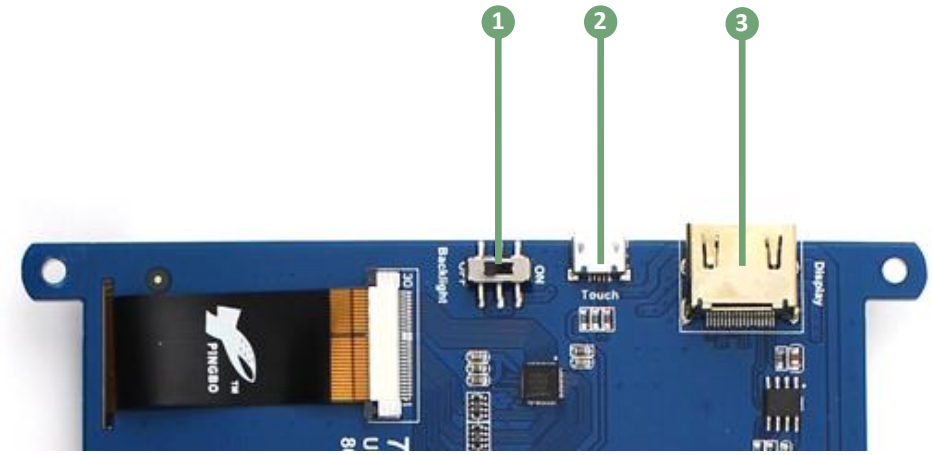
- 800×480 high resolution, touch control
- Supports Raspberry Pi, and driver is provided (works with custom Raspbian directly)
- Supports BB Black, comes with related images like: Angstrom
- Supports Banana Pi / Banana Pro, comes with related images like : Lubuntu, Raspbian
- Not only for mini-PCs, it can work as a computer monitor just like any other general HDMI screen (touch function is unavailable in this case)
- HDMI interface for displaying, USB interface for touch control

Rev 1.4, Date: September 25 2015

Contents

1. On-Board resource

Figure 1: Switches and interfaces



1) Backlight Switch: used to turn on/off the backlight.
2) USB Touch Interface: USB touch/power interface.
3) HDMI: it is used for connecting the main board to the LCD screen.

2. Working with Raspberry Pi

2.1. How to program Raspbian image file

In order to use with Raspberry Pi, you should configure the original system first. Of course, you can program a ready-to-use system image file to your Raspberry Pi board as well. In this section, we will illustrate how to program the image file by taking the ready-to-use system image file, 7inch HDMI LCD (B) Raspberry Pi 2 module B Raspbian image, as an example. This image file supports Raspberry Pi 2 Model B. Instead, for Raspberry Pi Model B/A+/B+, you can use 7inch HDMI LCD (B) Raspberry Pi B / B+ Raspbian image.

1) Download the zip file to your PC, unzip it and get an .img file.
2) Connect a TF card to your PC, and format your TF card with the SDFormatter.exe
   Notices: The capability of TF card in used here should be more than 4GB. In this operation, a TF card reader is also required, which has to be purchased separately.
3) Start the Win32DiskImager.exe, and select the system image file copied into your PC, then, click the button **Write** to program the system image file.

2.2. Hardware connection

1) Connect the LCD to the HDMI on the Raspberry Pi board with a HDMI cable;
2) Connect the USB Touch interface on the LCD to the USB interface on the Raspberry Pi board with a USB type-A male to micro-B cable.

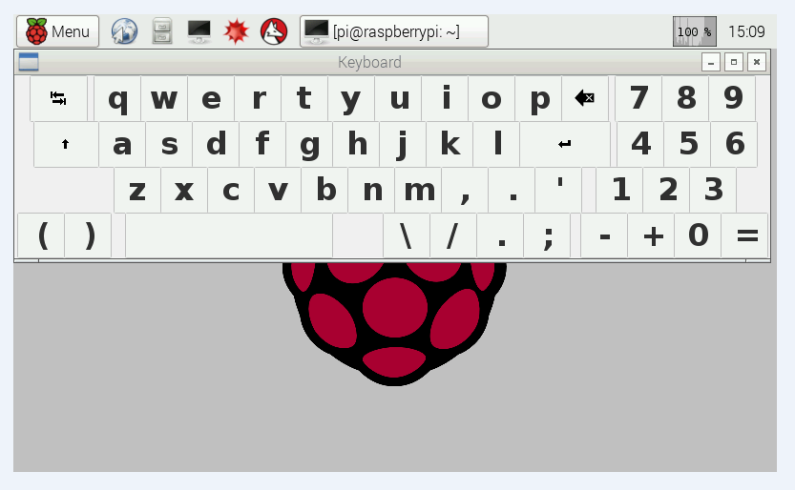
2.3. Virtual keyboard of Raspberry Pi

The Virtual keyboard of Raspbian system enables you to save the USB resource, providing easy system operations. After the LCD is working properly, this function can be invoked by the following command:

```
DISPLAY=:0.0 matchbox-keyboard -s 100 extended
```

Now, the virtual keyboard is ready to use, as Figure 2 shows.

Figure 2: Virtual keyboard of Raspberry Pi



2.4. Source code and protocol

1) Copy the source code to your Pi.
2) Execute the following command:

```
cd wavesahre-7inch-touchscreen-driver
chmod +x install.sh
sudo apt-get update
sudo ./install.sh
```

3) Shut down your Pi then power up again and usually you can use the display and touch functions.

Note: You may need for help about the source code but please consult the related website. We don't provide any supports of development environment building and source code modification.

See the Appendix: HID Proprietary Protocol.

2.5. How to configure original Raspbian image file

In the section above, we presented the steps of image programming by taking the ready-to-use system image file programming as an example (see Section 2.1), since the ready-to-use image is easier to use and understand. However, you can freely customize your system by configuring the original Raspbian image file to support this LCD module.

2.5.1. How to install driver script

1) Copy the driver *RPI2B_B_B+_USB_TOUCH_CAP_7.0_RASPBIAN.tar.gz For Raspberry Pi 2 Model B* or *RPIB_B+_USB_TOUCH_CAP_7.0_RASPBIAN_20150605.tar.gz For Raspberry Pi B+/A+/B* to your Raspbian system, and extract it. That is, enter the terminal and input the following command:
   `sudo tar zxvf file_name.tar.gz`
2) Change the current directory to the directory generated before and then run the script *USB_TOUCH_CAP_7.0_RASPBIAN* by the following command:
   `cd dir_file_name`
   `sudo ./ USB_TOUCH_CAP_7.0_RASPBIAN`
3) When finished, the system will reboot automatically. And the LCD module can work properly, including display and touch functions, after the system rebooted.
   Notice: If the max USB current is limited by the system, the LCD may not work properly. To unlock the current limitation, you can edit the /boot/config.txt and add:
   `max_usb_current=1`

2.5.2. How to enter graphical desktop

Notice: The Raspbian system boots to terminal interface by default. To make the operations simple when using with a LCD, it is recommended to set the system to boot to graphical desktop directly. Please follow the steps below to configure the system.

1) Enter the Raspbian system, and input the following command:
   `sudo raspi-config`
2) Select the option *Enable Boot to Desktop/Scratch* by using the Arrow keys, Space key or/and Enter key.
3) Select the option *Desktop Login as user 'Pi' at the graphical desktop*.
4) When you see the prompt "Would you like to reboot now?", select the option **Yes** to reboot the system.

2.6. How to use with the Ubuntu system

Program the image file *7inch HDMI LCD (B) ubuntu image* to the board. This image file supports Raspberry Pi 2B.

User Name: linaro

Password: linaro

3. How to use with BeagleBones

3.1. How to program Angstrom image file

If this LCD module is used for display only, you can program the latest Angstrom image file to the board directly without any change. The BeagleBone will read the display parameters of the 7 inch HDMI displayer and set the resolution to 800*480 automatically.

When using this LCD module as a touch screen, you should program the image file *7inch HDMI LCD (B) Angstrom image*. Please follow the steps below to program the image file.

1) Download the zip file to your PC, unzip it and get an .img file.
2) Connect a TF card to your PC, and format your TF card with the SDFormatter.exe
   Notices: The capability of TF card in used here should be more than 4GB. In this operation, a TF card reader is also required, which has to be purchased separately.
3) Start the Win32DiskImager.exe, and select the system image file copied into your PC, then, click the button **Write** to program the system image file.
4) After programming the image file, please insert the TF card to your board, press the key **uBOOT** and hold it till power up. Then, you will enter the system located at the TF card. Wait a moment and you will enter the graphical desktop directly.

3.2. Hardware connection

1) Connect the LCD to the HDMI on the BeagleBone board with a HDMI to micro HDMI cable (sold separately);
2) Connect the USB Touch interface on the LCD to the USB interface on the BeagleBone board with USB type-A male to micro-B cable. (BeagleBone has two USB interfaces, one for host and the other for client. In here, you should connect the LCD module to the USB host interface).

4. How to use with Banana Pi

Before powering up the Banana Pi, you should connect it to a LCD displayer properly, since the Banana Pi may read the resolution parameters of the LCD displayer on startup. And the connection should be remained till the Banana Pi enters the desktop. In this case, even if you disconnect the LCD displayer and reconnect it again to the Banana Pi, the LCD can still work properly.

4.1. How to program Raspbian_For_BananaPi image file

Program the image file *7inch HDMI LCD (B) Raspbian image* to the board. This image file supports the modules BananaPi Pro and BananaPi.

1) Download the zip file to your PC, unzip it and get an .img file.
2) Connect a TF card to your PC, and format your TF card with the SDFormatter.exe
   Notices: The capability of TF card in used here should be more than 4GB. In this operation, a TF card reader is also required, which has to be purchased separately.
3) Start the Win32DiskImager.exe, and select the system image file copied into your PC, then, click the button **Write** to program the system image file.

4.2. Hardware connection

1) Connect the LCD to the HDMI on the Banana Pi board with a HDMI cable;
2) Connect the USB Touch interface on the LCD to the USB interface on the Banana Pi board with a dual micro USB cable.

4.3. How to load WiFi driver of BananaPi Pro

Comparing with the Banana Pi, the BananaPi Pro has added an on-board WiFi module. When using the BananaPi Pro, you can use SSH to connect to the Pi and execute the following command to load the WiFi driver:

```
sudo modprobe ap6210
```

4.4. How to use with the Lubuntu system

Program the image file *7inch HDMI LCD (B) Lubuntu image* to the board. This image file supports the modules BananaPi Pro and BananaPi.

User name: bananapi

Password: bananapi

5. Appendix

7inch capacitive touch screen HID protocol

The Frame lengths are fixed at 25 bytes.

5 touch coordinates are reported but the Raspberry Pi only uses the first one.

Byte	Description
Byte 1	The frame header is 0xAA.
Byte 2	Touch flag (1: valid touch data. 0: The flag of touch finished)
Byte 3	The x-coordinate high byte of the 1st touch point.
Byte 4	The x-coordinate low byte of the 1st touch point.
Byte 5	The y-coordinate high byte of the 1st touch point.
Byte 6	The y-coordinate low byte of the 1st touch point.
Byte 7	The 1st touch finished flag is 0xBB. Raspberry Pi only uses the first 7 Bytes.
Byte 8	The quantity of touch coordinates (1~5)
Byte 9	The x-coordinate high byte of the 2nd touch point.
Byte 10	The x-coordinate low byte of the 2nd touch point.
Byte 11	The y-coordinate high byte of the 2nd touch point.
Byte 12	The y-coordinate low byte of the 2nd touch point.
Byte 13	The x-coordinate high byte of the 3rd touch point.
Byte 14	The x-coordinate low byte of the 3rd touch point.
Byte 15	The y-coordinate high byte of the 3rd touch point.
Byte 16	The y-coordinate low byte of the 3rd touch point.
Byte 17	The x-coordinate high byte of the 4th touch point.
Byte 18	The x-coordinate low byte of the 4th touch point.
Byte 19	The y-coordinate high byte of the 4th touch point.
Byte 20	The y-coordinate low byte of the 4th touch point.
Byte 21	The x-coordinate high byte of the 5th touch point.
Byte 22	The x-coordinate low byte of the 5th touch point.
Byte 23	The y-coordinate high byte of the 5th touch point.
Byte 24	The y-coordinate low byte of the 5th touch point.
Byte 25	The Frame end is 0xCC.