

HTT Series

For all variants of the HTT35A, HTT50A, HTT70A and HTT104A

Quick Start Guide

Revision 1.2



Contents

Revision History	2
Introduction	3
Features.....	3
Functional Diagram	3
Headers.....	4
HDMI Port.....	4
Mini USB.....	4
Barrel Jack	4
External Power	4
Getting Started	5
Windows	5
Raspberry Pi	5
TinkerBoard.....	5
BeagleBone	5
USB Drivers	6
Extended Display Information Data (EDID).....	7
Touchscreen	7
Troubleshooting	8
Power	8
Display.....	8
Touch	8
Recommended Accessories	9
HDMI Cable	9
USB Communication	9
External Power.....	9
Support	9
Contact	9

Revision History

Revision	Date	Description	Author
1.2	September 5, 2019	Clarified Raspberry Pi Getting Started section	Divino
1.1	December 1, 2017	Voltage Correction	Divino
1.0	November 6, 2017	First Release	Divino

Introduction

The HTT series offers full colour TFT displays, with a simple HDMI and USB plug and play interface. The HTT can be configured with a touch screen, providing control without the need of a mouse or keyboard. Designed with industrial applications in mind, the HTT is perfect for panel mounted applications and HMI interfaces.

Features

- Compatible with Windows and Linux hosts, including:
 - Raspberry Pi
 - BeagleBone
 - TinkerBoard
- Plug and Play
- USB for touch screen, HMI mouse interface
- Upgradable touch driver firmware
- DVI/HDMI interface
- Automatic resolution detected provided by EDID
- Low profile mounting holes
- Industrial 5-35V power input (-VPT) option
- Available in:
 - Non-touch,
 - Resistive touch
 - Capacitive touch

Functional Diagram

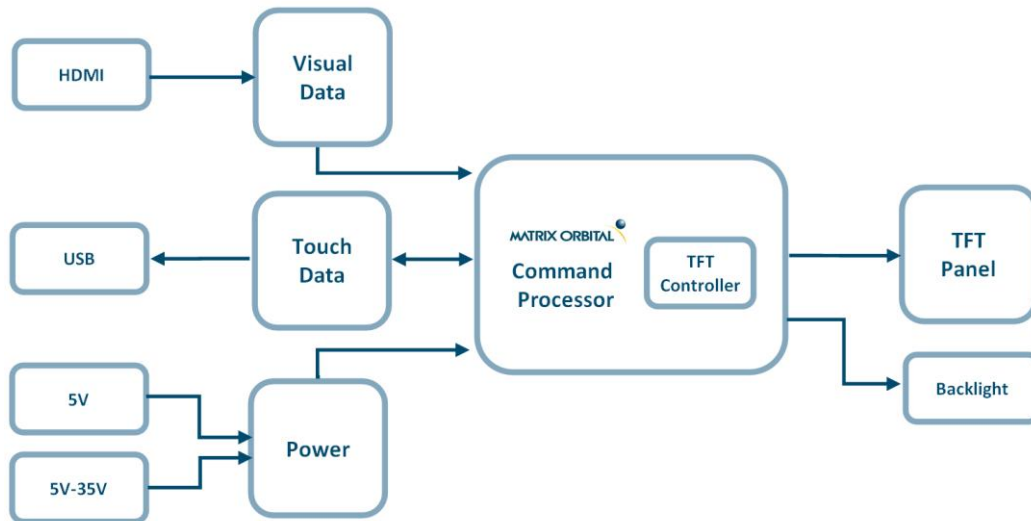


Figure 1: HTT Block Diagram

Headers

HDMI Port

Recommended Mate: HDMI3FT or HDMI6FT Cable

When properly connected, the HDMI cable will provide 5V at 55mA (minimum) for the purpose of reading the display's EDID settings.

The HTT uses HDMI/DVI 1.0 and is only compatible with unencrypted video signals. HDCP video signals are not supported.

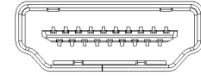


Figure 2: HDMI Header

Mini USB

Recommended Mate: EXTMUSB3FT Cable

Up to 500mA of power can be provided through the Mini-USB header to power the HTT. The USB header must be connected to the host in order for the touch screen to be functional, as touch communication occurs over USB.

The Mini USB header is also used to access the HTT's touch firmware.

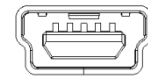


Figure 3: Mini-USB Header

Barrel Jack

Recommended Mate: PWR-ACDC-5V2A

A power adaptor may be applied to any HTT Series display, providing power to the unit. When choosing an adaptor, please ensure it is a center positive model that conforms to the voltage and current requirements of your display. Please consult the Power Consumption section for details.

For the HTT70A, it is suggested that additional power is supplied through the barrel jack or external power header.

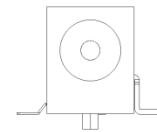


Figure 4: Barrel Jack Header

External Power

Additional power can be applied to the display through the external power header. Voltages between 5V-35V can be applied using this header.

For the HTT70A, it is suggested that additional power is supplied through the barrel jack or external power header

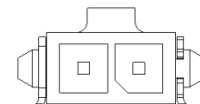


Figure 5: External Power Header

Getting Started

When connecting your HTT display, power must first be applied through the USB header, the optional barrel jack or the optional external power header. Once powered, connect the HTT to your source HDMI cable

Windows

When connected to Windows, the HTT's resolution will be automatically detected and set through EDID. The HTT display should immediately appear in display settings. No additional software setup should be required.

Raspberry Pi

Before the display can be used with a Raspberry Pi unit, some setup will be required. The HDMI resolution will have to be configured through the config.txt file (located in /boot/config.txt). We suggest that you edit this file remotely via SSH, or edit the file on a separate PC by powering the Pi down, ejecting the micro SD card, and inserting it into another computer's card reader.

To force the Raspberry Pi to use the HTT's resolution, the config.txt file must be modified to include the following:

```
hdmi_group=2  
hdmi_mode=87
```

The resolution will also have to be specified. The setting changed will be dependent on which display is used.

Table 1: HDMI Custom commands

Display	Command
HTT50A	hdmi_cvt= 800 480 60 6 0 0 0
HTT70A	hdmi_cvt= 800 480 60 6 0 0 0
HTT104A	hdmi_cvt= 1024 100 60 6 0 0 0

Finally, if the display is being powered solely through the USB header, the Raspberry Pi can be configured to increase how much current can be output through USB.

```
Max_usb_current=1
```

The following is an example of what modifications need to be made to configure the Raspberry Pi to work with a HTT50A/HTT70A

```
hdmi_group=2  
hdmi_mode=87  
hdmi_cvt= 800 480 60 6 0 0 0  
Max_usb_current=1
```

TinkerBoard

The HTT will be compatible with TinkerOS-Debian. Once connected, the Tinkerboard will autodetect the HTT's display settings. No further software configuration is required.

The HTT is not compatible with TinkerOS-Android Marshmallow.

BeagleBone

The BeagleBone board will autodetect the HTT's EDID display settings automatically. No further software configuration is required.

USB Drivers

Once the HTT is properly installed in windows, 3 new devices will be present in Device manager.

- HID-Compliant Mouse
- USB Composite Device
- WinUsb Device

Table 2: HTT Device VID/PID

VID	0x1B3D
PID	0x14C2

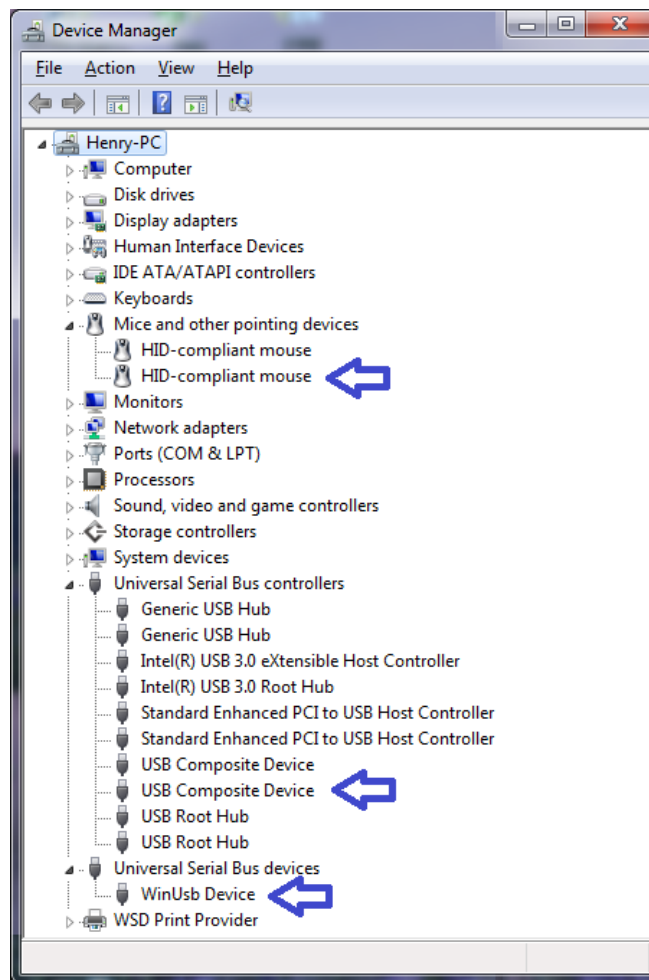


Figure 6: HTT USB Driver Listing

Extended Display Information Data (EDID)

The HTT's on-board EDID comes preconfigured, so users won't have to worry about setting up their screen resolution or display timings. Once plugged in, the EDID will automatically detect the HTT display resolution.

EDID settings can be accessed through the HTT's I2C headers. Please find EDID examples below:

```
static byte[] EDID_HTT50A = {
    0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x35, 0xE3, 0x01, 0x00, 0x01, 0x00, 0x00, 0x00,
    0xFF, 0x1B, 0x01, 0x03, 0x80, 0x0B, 0x06, 0x00, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,
    0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xE4, 0x0C, 0x20, 0x80, 0x30, 0xE0, 0x2D, 0x10, 0x28, 0x30,
    0xD3, 0x00, 0x6C, 0x41, 0x00, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x2E,
};
```

```
static byte[] EDID_HTT70A = {
    0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x35, 0xE3, 0x02, 0x00, 0x01, 0x00, 0x00, 0x00,
    0xFF, 0x1B, 0x01, 0x03, 0x80, 0x0F, 0x09, 0x00, 0x0A, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,
    0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xE4, 0x0C, 0x20, 0x7D, 0x30, 0xE0, 0x2D, 0x10, 0x25, 0x30,
    0xD3, 0x00, 0x9A, 0x56, 0x00, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE1,
};
```

```
static byte[] EDID_HTT104A = {
    0x00, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0x00, 0x35, 0xE3, 0x03, 0x00, 0x01, 0x00, 0x00, 0x00,
    0xFF, 0x1B, 0x01, 0x03, 0x80, 0x1A, 0x03, 0x00, 0x0A, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,
    0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0xC4, 0x09, 0x00, 0x40, 0x41, 0x82, 0xB4, 0x00, 0x28, 0x30,
    0xD3, 0x00, 0x08, 0x1A, 0x10, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x10,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xEC,
};
```

Touchscreen

Both the Resistive and Capacitive touch variants of the HTT display comes with a touch driver. In order for the touch driver to operate properly, the HTT needs to be connected through USB. In addition to supplying power, the USB header will also allow transfer touch data to the controller board.

Troubleshooting

Power

To function correctly, the HTT70A must be supplied with the appropriate power. If the power 3V3 LED near the top right corner of the board is not illuminated, power is not applied correctly. Try the tips below.

- HTT devices have specific power requirements. Ensure the correct voltage and sufficient current are available to your device by consulting your HTT's Power consumption table.
- Check the power cable which you are using for continuity. If you don't have an ohm meter, try using a different power cable, if this does not help try a different power supply.
- Check the power connector in use on your display. If the connector has become loose or you are unable to resolve the issue, please use the Contact section to reach a friendly Matrix Orbital support representative.

If your display is showing picture but continuously flickering, your display is not receiving enough power. You'll have to apply more power through a powered USB hub, or the optional Barrel jack/external power header. If you are running a Raspberry Pi, you can configure your USB port to supply more current by editing the config.txt file.

Display

When connected and powered correctly, the display should show picture from edge to edge. There should be no white lines on the edges of the display. If this is not the case, check out these tips.

- If your display shows picture but there are white lines along the edge(s) of the display, your display may be receiving the incorrect number of pixels. The HTT does not have a resolution scaler, and therefore the appropriate 800 x 480 resolution must be specified for all the pixels to be used. If you are using a Raspberry pi, you can configure the config.txt file to force an 800x480 resolution.

Touch

If your display is showing picture but not responding to touch inputs, please check to ensure your USB cable is properly connected. If the USB LED near the top right corner of the board is not illuminated, the USB connection is not recognized.

Recommended Accessories

HDMI Cable



Figure 7: HDMI Cable

Part Number:
HDMI3FT or HDMI6FT

Description:
3ft (1m) or 6ft (2m) HDMI Cable

USB Communication



Figure 8: EXTMUSB3FT

Part Number:
EXTMUSB3FT

Description:
3ft (1m) Mini-B USB Cable

External Power



Figure 9: PWR-ACDC-5V2A

Part Number:
PWR-ACDC-5V2A

Description:
9ft (2.7m) AC to DC power adapter with 2.1mm center positive barrel jack.

Input Voltage: 100-240V AC
Output Voltage: 5V DC @ 2A

AC Plugs Included: North American (NEMA), Europlug (Type C), UK (Type G),
Australia/China/New Zealand/Argentina (Type I)

Support

Phone: 403.229.2737

Email: support@matrixorbital.ca

Manuals and Drivers: <http://www.matrixorbital.ca>

Forums: <http://www.lcdforums.com/forums>

Contact

Sales

Phone: 403.229.2737

Email: sales@matrixorbital.ca

Support

Phone: 403.229.2737

Email: support@matrixorbital.ca

Online

Purchasing: www.matrixorbital.com

Support: www.matrixorbital.ca