StarFive 赛昉科技

# VisionFive SBC
# Quick Start Guide

Version: V1.1

Date: 2022-03-01

**PROPRIETARY NOTICE**

Copyright © Shanghai StarFive Technology Co., Ltd., 2018-2022. All rights reserved.

Information in this document is provided "as is," with all faults. Contents may be periodically updated or revised due to the product development. Shanghai StarFive Technology Co., Ltd. (hereinafter “StarFive”) reserves the right to make changes without further notice to any products herein.

StarFive expressly disclaims all warranties, representations, and conditions of any kind, whether express or implied, including, but not limited to, the implied warranties or conditions of merchantability, fitness for a particular purpose and non-infringement.

StarFive does not assume any liability rising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation indirect, incidental, special, exemplary, or consequential damages.

All material appearing in this document is protected by copyright and is the property of StarFive. You may not reproduce the information contained herein, in whole or in part, without the written permission of StarFive.

**Shanghai StarFive Technology Co., Ltd.**

Address: Room 502, Building 2, No. 61 Shengxia Rd., China (Shanghai) Pilot Free Trade Zone, Shanghai, 201203, China

Website: www.starfivetech.com

Email: sales@starfivetech.com (sales)

support@starfivetech.com (support)

StarFive

# About This Manual

**Introduction**

This document mainly provides the users with the necessary information about the StarFive VisionFive development board, including features, specifications, board appearance and pinout, as well as the guidelines to get started with the Fedora operation system.

**Revision History**

| Version | Released | Revision |
|---|---|---|
| V1 | 2021-12-08 | The first official release. |
| V1.1 | 2022-03-01 | • Added new steps to use Ethernet to prepare software under Windows and Mac/Linux environment in the Software Setup section.<br>• Updated the Fedora image file name as `Fedora-riscv64-jh7100-developer-xfce-Rawhide-20211226-214100.n.0-sda.raw.zst`.<br>• Added steps to recover bootloader for Mac/Linux.<br>• Updated the description in the *Software Setup* section. |

StarFive

# Table of Contents



StarFive

# 1 Features and Specifications

VisionFive is the first generation of affordable RISC-V computers designed to run Linux. It is fully open-source with open-source software, open hardware design and RISC-V open architecture.

It is powered by RISC-V SiFive U74 Dual-Core 64-bit RV64GC ISA SoC with 8GB LPDDR4 RAM and has rich I/O peripherals such as USB 3.0 ports, 40-pin GPIO header, Gigabit Ethernet Connector, Micro-SD card slot and much more.

VisionFive also has rich AI features with Neural Network Engine and NVDLA Engine. It has onboard audio and video processing capabilities and has MIPI-CSI and MIPI-DSI connectors for video hardware. It has wireless capabilities with Wi-Fi and Bluetooth (BLE) and has a wide software compatibility including support for Fedora.

## 1.1 Features

- Truly open-source hardware, software and RISC-V open architecture
- Powerful and rich AI features with Neural Network Engine and NVDLA Engine
- Abundant I/O peripherals
- Wireless connectivity with Wi-Fi and BLE
- Onboard video and audio processing
- Wide software compatibility including support for Fedora

## 1.2 Specifications

**Table 1-1 Specifications**

| Specification | Details |
|---|---|
| Processor | • RISC-V SiFive U74 Dual-Core 64-bit RV64GC ISA SoC with 2MB L2 cache @ 1.0GHz<br>• Vision DSP Tensilica-VP6 for computing vision @ 600MHz<br>• NVDLA Engine (configuration 2048 MACs @ 800MHz)<br>• Neural Network Engine (1024MACs @ 500MHz) |
| Memory | 8GB LPDDR4 |
| Wireless Connectivity | • 2.4 GHz Wi-Fi (IEEE 802.11b/g/n)<br>• Bluetooth 4.2 (BLE) |

StarFive

| Specification | Details |
|---|---|
| Video Processing | • 2 x MIPI-CSI (up to 4K@30fps), 1 x MIPI-DSI (up to 4K@30fps)<br>• 1 x HDMI 1.4 (up to 1080p@60fps display)<br>• Video Decoder (H264/H265) up to 4K@60fps; Support Dual stream decoding for 2K@30fps each<br>• Dual channels of ISP, each channel support up to 4K@30FPS<br>• Support MIPI-CSI TX for video output after ISP and AI processing<br>• JPEG encoder/decoder |
| Dedicated Audio Processing DSP and Sub-system | • Ultra-low power Voice Activity Detector for audio bit-stream as a Voice Trigger<br>• On-chip Audio DAC<br>• Support DMIC and AMIC, up to 4 channels |
| Peripherals | • 4 x USB 3.0 ports<br>• 40 Pin GPIO Header (28 x GPIO, I2C, I2S, SPI, UART)<br>• Gigabit Ethernet Connector<br>• 3.5 mm Audio jack (4-pole stereo audio output)<br>• Micro-SD card slot for system boot and data storage<br>• Support TRNG and OTP<br>• Support DMAC, QSPI and other peripheral<br>• Reset button and Power Button |
| Power Supply | • Minimum: 5 V / 1.5 A<br>• Recommended: 5 V / 3 A |
| Power Connector | • USB Type-C port or 40-pin GPIO header |
| Dimensions | • 100 mm x 72 mm |

StarFive

# 2 Hardware Overview

## 2.1 Board Appearance

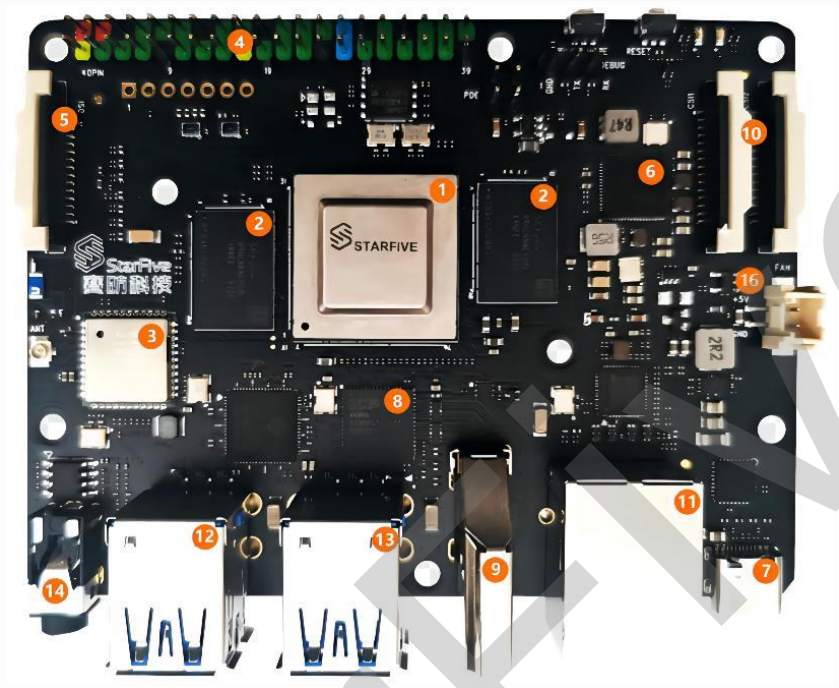



**Figure 2-1** Top View of VisionFive

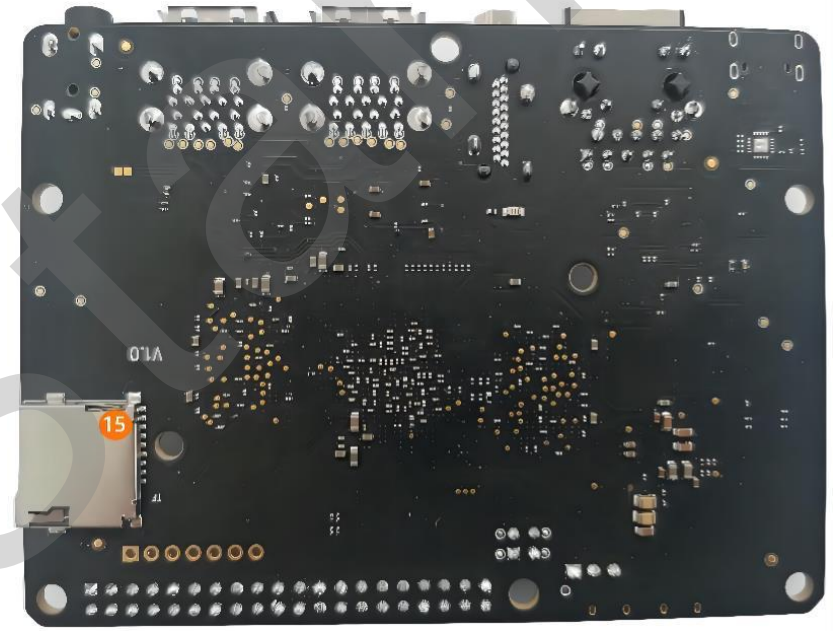



**Figure 2-2** Bottom View of VisionFive

StarFive

**Table 2-1 Interface Description**

| No. | Description | No. | Description |
|---|---|---|---|
| 1 | RISC-V SiFive U74 Dual-Core 64-bit RV64GC ISA SoC | 9 | HDMI 2.0 Connector |
| 2 | 4GB LPDDR4 RAM | 10 | 2 x MIPI-CSI Connector |
| 3 | 2.4 GHz Wi-Fi and Bluetooth 4.2 (BLE) | 11 | Gigabit Ethernet (RJ45 Connector) |
| 4 | 40 Pin GPIO Header | 12 | 2 x USB 3.0 Host Type-A |
| 5 | MIPI-DSI Connector | 13 | 2 x USB 3.0 Host Type-A |
| 6 | PMIC | 14 | 3.5 mm Audio Jack (4-pole stereo audio output) |
| 7 | USB Type-C Connector | 15 | Micro-SD SDXC Card Slot |
| 8 | LCD to HDMI IC | 16 | Fan Header for a 2-pin 5 V Fan |

**Notes:**

The recommended current for this board is 3 A. The reason is as the following: The onboard components need about 1 A, the 4 USB ports can draw a total of 1 A combined, one USB port alone can also draw 1 A and the remaining current is for expansion header and others.

## 2.2 Pinout Diagram

The following is the pinout diagram:

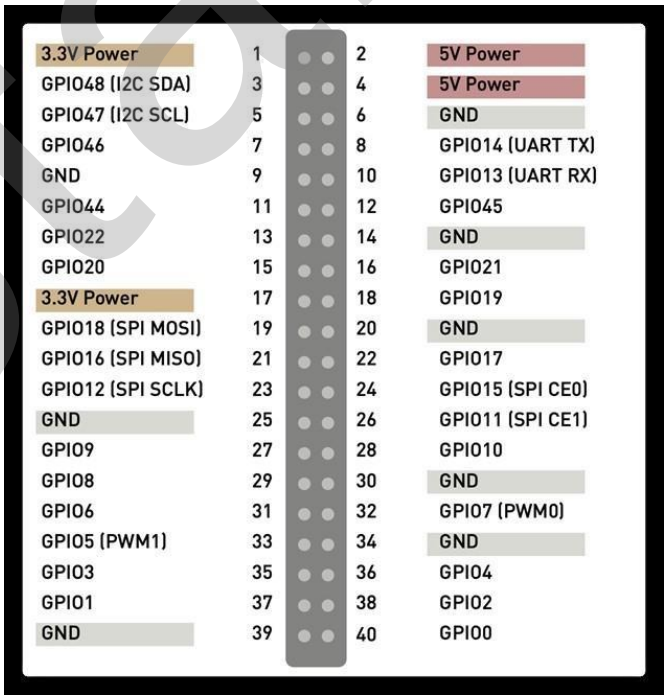



**Figure 2-3** Pinout Diagram

StarFive

**Notes:**

Each GPIO pin can safely draw a maximum current of 39 mA, whereas the maximum current draw when all GPIOs are combined should be less than 100 mA. Please take this into account or otherwise you will end up destroying the GPIO pins.

All GPIOs can be configured to support different functions including but not limited to SDIO, Audio, SPI, I2C, UART and PWM.

For the instructions, refer to StarFive_40-Pin_GPIO_Header_User_Guide.

StarFive

# 3 Getting Started

## 3.1 Required Hardware

You need to prepare the following hardware before getting started with VisionFive：

- VisionFive
- 16GB (or more) micro-SD card
- micro-SD card reader
- Computer (PC/Mac/Linux)
- USB to serial converter (3.3 V I/O)
- Ethernet cable
- Power adapter (5 V / 3 A)
- USB Type-C Cable

## 3.2 Connecting a Fan

It's recommended to use a fan with VisionFive. You can connect a 2-pin 5 V fan to the board as follows:

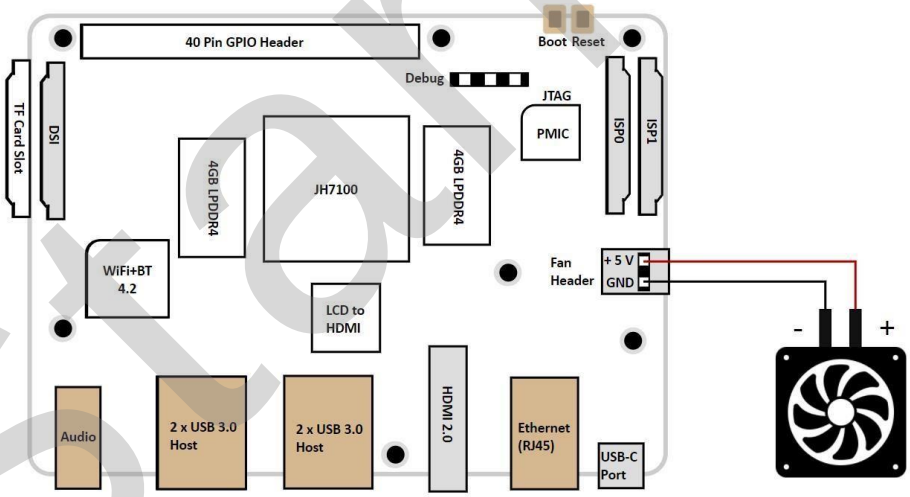



**Figure 3-1** Connecting a Fan

## 3.3 Flashing Fedora OS to Micro-SD Card

Now we need to burn Fedora (which a Linux distribution) to a micro-SD card, so that it can run on the VisionFive. Follow the steps below according to your operating system.

Start by downloading the Fedora image from here.

StarFive

### 3.3.1 For Windows

**Steps:**

**Step 1** Insert a micro-SD card to the computer through a micro-SD card reader, or by a built-in card reader on a laptop.

**Step 2** Download Zstandard-CLI software by visiting the links below:

- Zstandard-CLI for windows 32-bit
- Zstandard-CLI for windows 64-bit

**Step 3** Click here to visit the GitHub repo.

**Step 4** Extract the `.zip` file.

**Step 5** Copy the latest file (e.g.: `Fedora-riscv64-jh7100-developer-xfce-Rawhide-20211226-214100.n.0-sda.raw.zst`) and paste it into the `zstd` directory that you just extracted.

**Step 6** Open Windows Powershell and navigate to the zstd directory.

**Example:**

```
cd D:\Downloads\zstd
```

**Step 7** Type the following command to unzip the Fedora image:

```
./zstd.exe -d Fedora-riscv64-jh7100-developer-xfce-Rawhide-20211226-214100.n.0-sda.raw.zst -o Fedora-riscv64-jh7100-developer-xfce-Rawhide-20211226-214100.n.0-sda.raw
```

**Result:**

Now your image file is named as `Fedora-riscv64-jh7100-developer-xfce-Rawhide-20211226-214100.n.0-sda.raw`.

**Step 8** Visit this link to download BalenaEtcher. We will use BalenaEtcher software to flash the Fedora image to a micro-SD card.

**Step 9** Install BalenaEtcher and open it.

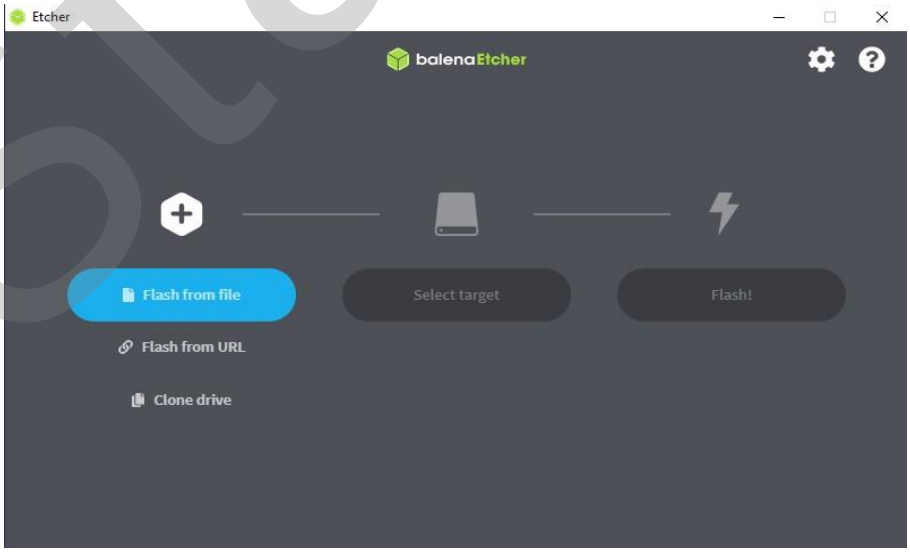



**Figure 3-2** Install BalenaEtcher

**Step 10** Click on **Flash from file** and select the location of the image that we just unzipped:

Starfive

```
Fedora-riscv64-jh7100-developer-xfce-Rawhide-20211226-
214100.n.0-sda.raw
```

**Step 11** Click **Select target** and select the connected micro-SD card.

**Step 12** Click **Flash!**

### 3.3.2 For Mac/Linux

**Steps:**

**Step 1** Insert a micro-SD card to the computer through a micro-SD card reader, or by a built-in card reader on a laptop.

**Step 2** Open a terminal window on Mac/Linux.

**Step 3** Type the following to update the packages list.

```
sudo apt-get update
```

> **Information:**
>
> If you are a Mac user, type `brew update`.

**Step 4** Type the following to install zstd package which we will use to unzip our Fedora image file:

```
sudo apt-get install zstd
```

> **Information:**
>
> If you are a Mac user, type `brew install zstd`.

**Step 5** Navigate to the location of the downloaded Fedora image directory before. Example Command:

```
cd Downloads/
```

**Step 6** Run the following command to unzip the Fedora image.

```
zstd -d Fedora-riscv64-jh7100-developer-xfce-Rawhide-
20211226-214100.n.0-sda.raw.zst
```

**Step 7** Burn the Fedora image to the micro-SD card by running the following command.

```
sudo dd if=Fedora-riscv64-jh7100-developer-xfce-Rawhide-
20211226-214100.n.0-sda.raw of=/dev/sdX bs=8M status=progress
&& sync
```

> **Information:**
>
> - If you are a Mac user, burn the Fedora image to the micro-SD card by running: `sudo dd if=Fedora-riscv64-jh7100-developer-xfce-Rawhide-20211226-214100.n.0-sda.raw of=/dev/sdX bs=8m && sync`
> - `of=/dev/sdX` corresponds to the location of the connected micro-SD card. You can find this by running lsblk command.
> - The whole burning process will take about 20 minutes.

Starfive

## 3.4 Logging in to Fedora

### 3.4.1 Using Xfce Desktop over HDMI

After installing Fedora, you can log in to Fedora OS on VisionFive using Xfce Desktop over HDMI.

**Steps:**

**Step 1** After the HDMI of the display screen is connected, insert the micro-SD card with the Fedora image into the VisionFive and power on.

**Step 2** After the desktop login system is displayed, you can use the keyboard and mouse on VisionFive.

**Step 3** Enter the credentials as follows:

- **Username**: riscv (default)
- **Password**: starfive

**Result:**

You will see the following interface:


**Figure 3-3** UI Example

### 3.4.2 Using SSH over Ethernet

After installing Fedora, you can log in to Fedora OS on VisionFive through an SSH connection over the local network.

**Steps:**

**Step 1** Insert the micro-SD card with the Fedora image into the VisionFive and power on.

**Step 2** Connect one end of an Ethernet cable to the RJ45 connector on the VisionFive and the other end of the cable to a router.

**Step 3** After successful Ethernet connection, your router will assign an IP address to the VisionFive and it will be connected to the Internet.

StarFive

**For Windows**

**Steps:**

**Step 1** Log in to your router (usually you need to enter 192.168.1.1 on the web browser to enter the router).

**Step 2** Go to DHCP configuration and find the IP address of the VisionFive.

> **Information:**
>
> You can easily find the IP address of the VisionFive by referring to its host name, `fedora-starfive`.

**Step 3** Download and install Putty by visiting this link.

> **Information:**
>
> Putty is an SSH and telnet client through which you can connect to the Carrier Board. You can skip this step if you already have Putty installed.

**Step 4** Open Putty to log in to Fedora.

**Step 5** Select SSH under the **Connection Type**.

**Step 6** Configure the settings as follows:

- **Host Name**: IP address of your VisionFive
- **Port**: 22

**Step 7** Click Open.

**Step 8** Enter the credentials as follows:

- **Username**: riscv
- **Password**: starfive

**Result:**

Now you have connected with the VisionFive via SSH using windows!


**Figure 3-4** Example Output

**For Mac/Linux**

**Steps:**

**Step 1** Log in to your router (usually you need to enter 192.168.1.1 on the web browser to enter the router).

**Step 2** Go to DHCP configuration and find the IP address of the VisionFive.

> **Information:**
>
> You can easily find the IP address of the VisionFive by referring to its host name, **fedora-starfive**.

**Step 3** Open a terminal window and type the following:

```
ssh riscv@192.168.1.xxx
```

StarFive

> **Information:**
>
> 192.168.1.xxx is the IP address of VisionFive.

**Step 4** Type the password as starfive in the prompt.

**Result:**

Now you have connected with the VisionFive via SSH using Mac/Linux!

> **Information:**
>
> 192.168.1.xxx is the IP address of VisionFive.

```
ryan@ubuntu:~$ ssh riscv@192.168.100.22
The authenticity of host '192.168.100.22 (192.168.100.22)' can't be established.
ECDSA key fingerprint is SHA256:pta2CSJNc5o5VfUvI9xYomIRhLXqRX9OQJt7rCrrFy8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.100.22' (ECDSA) to the list of known hosts.
riscv@192.168.100.22's password:
Last login: Wed Jun 30 14:27:45 2021 from 192.168.100.23
[riscv@fedora-starfive ~]$
```

**Figure 3-5** Example Output

### 3.4.3 Using a USB to Serial Converter

You can log in to Fedora OS on VisionFive using a USB-to-Serial converter. Please follow the following steps:

**For Windows**

**Steps:**

**Step 1** Insert the micro-SD card with the Fedora image burnt into VisionFive.

**Step 2** Connect one end of the USB Type-C cable to the USB Type-C port on the VisionFive, and connect the other end of the cable to the power adapter.

**Step 3** Connect the jumper wires from the USB to Serial Converter to the 40-Pin GPIO header of the VisionFive as follows.

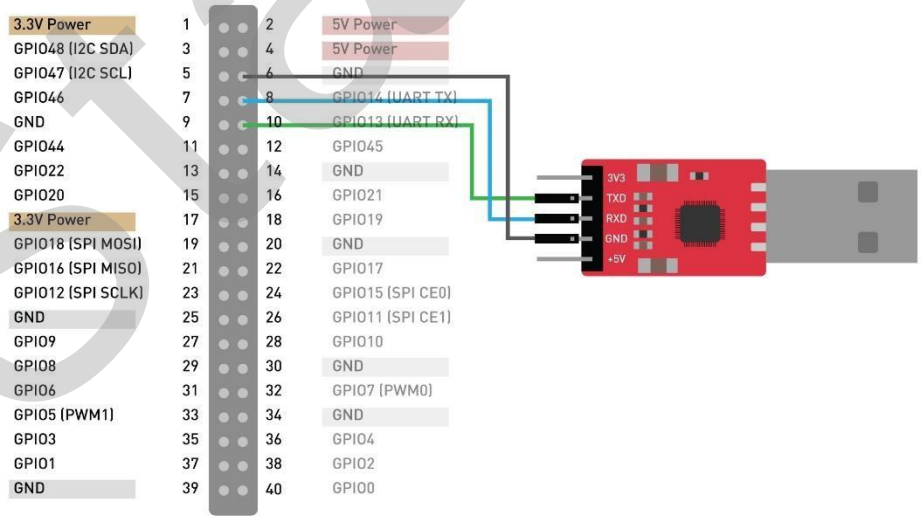



**Figure 3-6** Connecting Jumper Wire

**Step 4** Connect the USB-to-Serial converter to the PC.

**Step 5** Open **Device Manager** by typing `Device Manager` in the windows search box.

**Step 6** Click the drop-down arrow from Ports (COM & LPT) and find the name of the connected serial port (e.g.: COM4).


**Figure 3-7** Example

**Step 7** Download and install Putty by visiting this link.

> **Information:**
>
> Putty is an SSH and telnet client through which you can connect to the Carrier Board via SSH. You can skip this step if you already have Putty installed.

**Step 8** Open Putty to connect the PC to the Carrier Board.

**Step 9** Select Serial under the Connection Type.

**Step 10** Configure the settings as follows:

- Serial line: COM4 (choose your COM port)
- Speed: 115200

StarFive

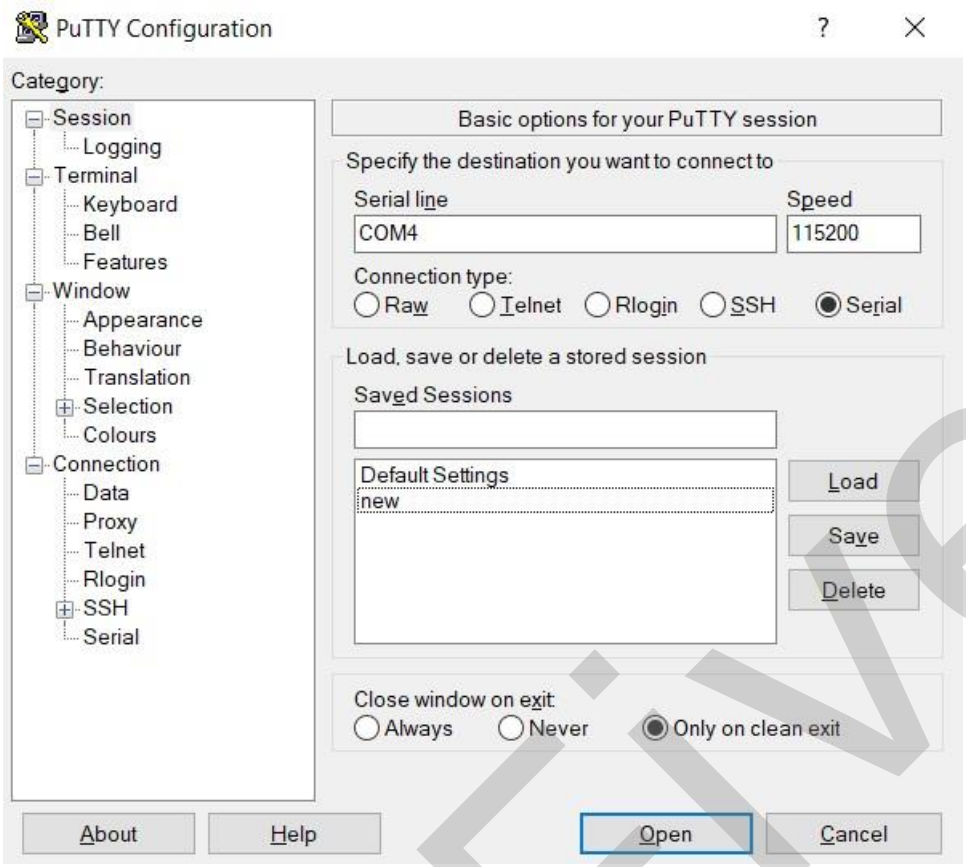



**Figure 3-8** Example Configuration

**Step 11** Click **Open**.

**Step 12** Power on the VisionFive.

**Step 13** Type username and password in the prompt as follows:

- Username: riscv
- Password: starfive

**Result:**

Now you have connected with the VisionFive via serial communication using windows!

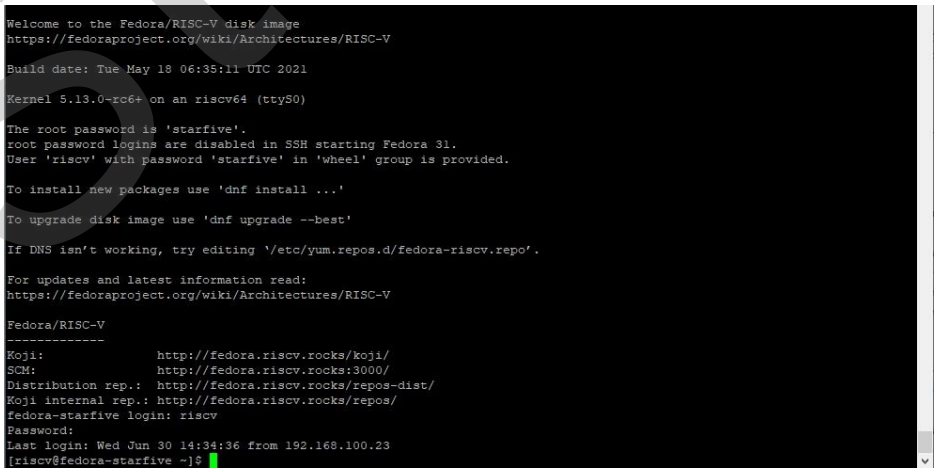



**Figure 3-9** Example Output

StarFive

**For Mac/Linux**

**Steps:**

**Step 1** Insert the micro-SD card with the Fedora image burnt into VisionFive.

**Step 2** Connect one end of the USB Type-C cable to the USB Type-C port on the VisionFive, and connect the other end of the cable to the power adapter.

**Step 3** Connect the jumper wires between the USB-to-Serial converter and the 40-Pin GPIO header of the VisionFive as follows.

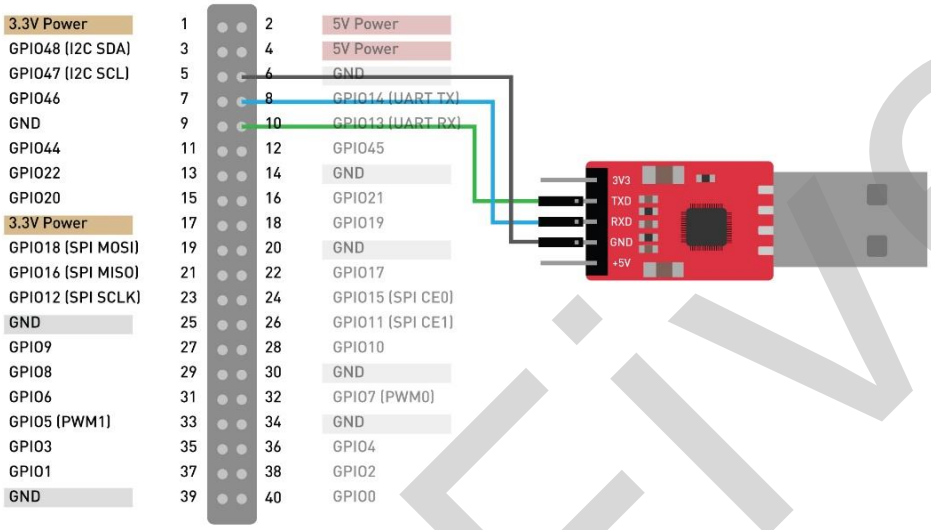



**Figure 3-10** Connecting Jumper Wires

**Step 4** Connect the USB-to-Serial converter to the PC.

**Step 5** Open a terminal window on Mac/Linux.

**Step 6** Update the packages list by typing the following command.

```
sudo apt-get update
```

**Step 7** Install minicom by typing the following command.

```
sudo apt-get install minicom
```

**Step 8** View the connected serial devices.

```
dmesg | grep tty
```

```
ryan@ubuntu:~$ dmesg |grep tty
[    1.066754] printk: console [tty0] enabled
[    2.587393] 00:05: ttyS0 at I/O 0x3f8 (irq = 4, base_baud = 115200) is a 16550A
[436488.744116] usb 3-2: pl2303 converter now attached to ttyUSB0
```

**Figure 3-11** Example Output

**Step 9** Connect to the serial device by typing the following command.

```
sudo minicom -D /dev/ttyUSB0 -b 115200
```

> **Information:**
>
> The baud rate is set to 115200.


**Figure 3-12** Example Output

**Step 10** Power on the VisionFive

**Step 11** Type username and password in the prompt as follows:

- **Username**: riscv
- **Password**: starfive

**Result:**

Now you have connected with the VisionFive via serial communication using Mac/Linux!

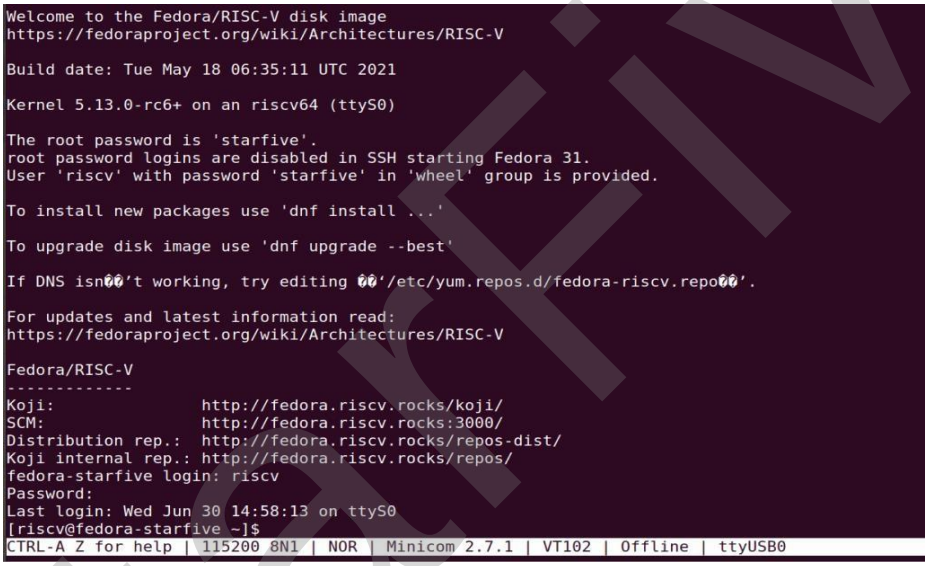



**Figure 3-13** Example Output

StarFive

# 4 Appendix A: VisionFive Boot Flow

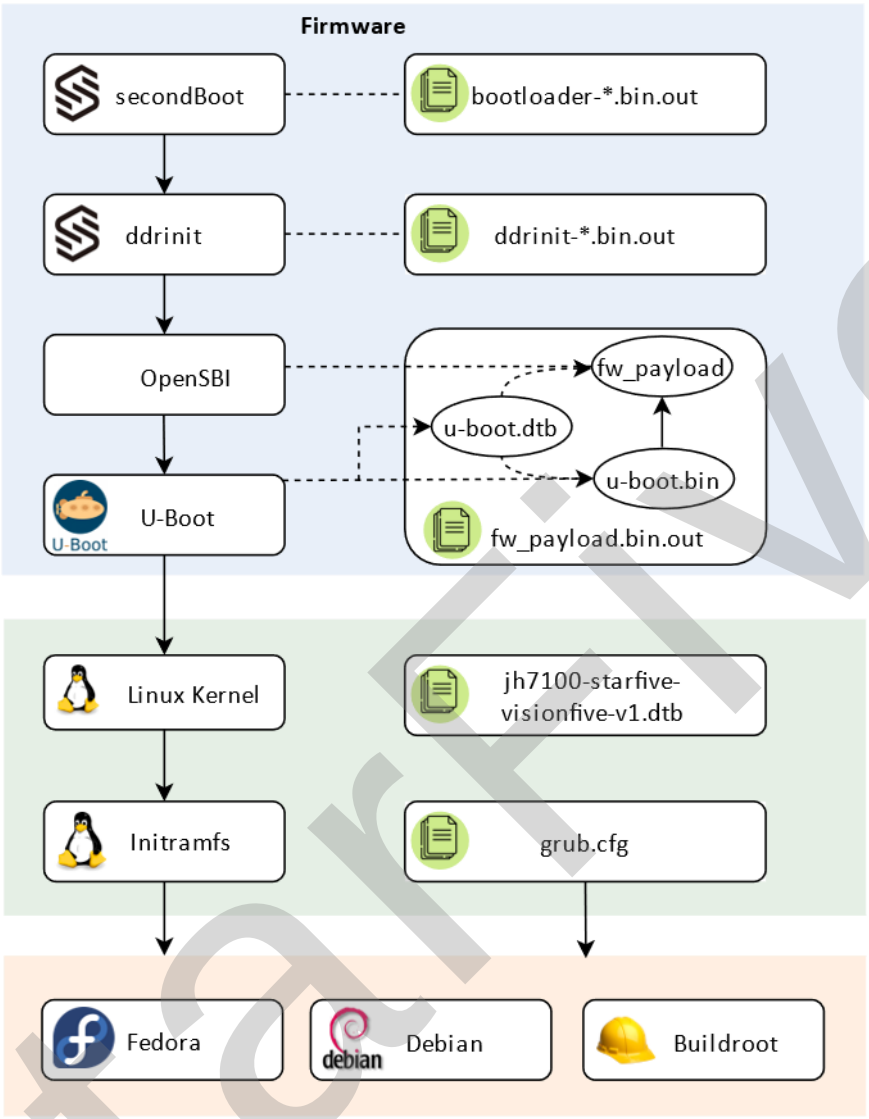



**Figure 4-1** VisionFive Boot Flow

StarFive

# 5 Appendix B: Updating Firmware and u-boot

## 5.1 Using Pre-Built Binary Packages

> **Information:**
>
> Follow the GitHub repo. Update the firmware and U-Boot if there is an update notice. Refer to How to access the GitHub repository? section in thie document for the link of the repo.

Please visit the following links to download the bootloader, ddr init and u-boot files.

- bootloader
- ddr init
- u-boot

### 5.1.1 Hardware Connection

To connect hardware, perform the following:

**Steps:**

**Step 1** Connect one end of the USB Type-C cable to the USB Type-C port on the VisionFive, and connect the other end of the cable to the power adapter.

**Step 2** Connect the jumper wires between the USB-to-Serial converter and the 40-Pin GPIO header of the VisionFive as follows.

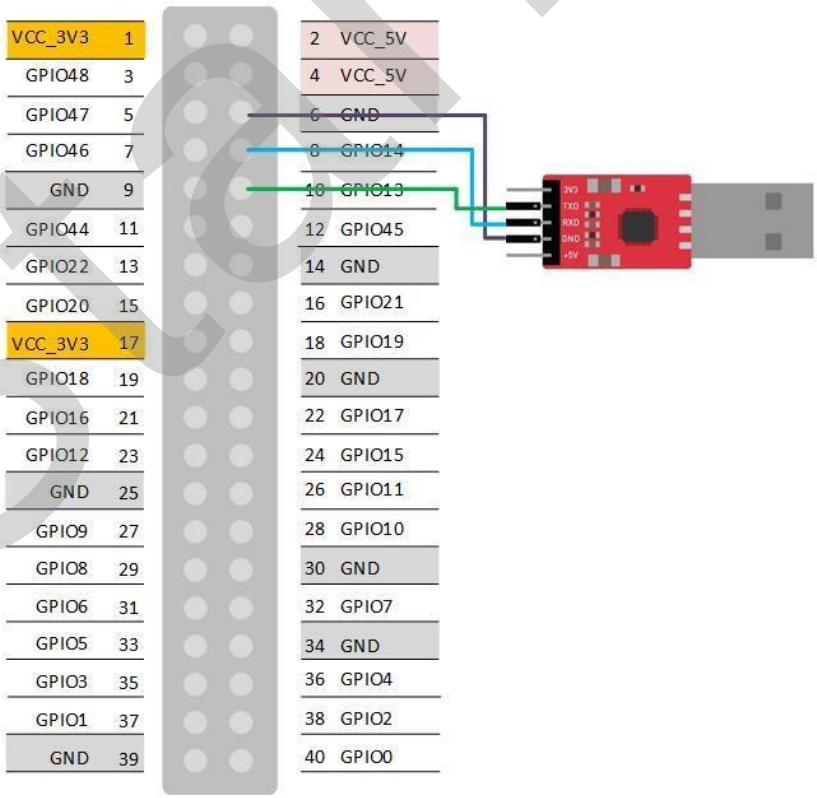



**Figure 5-1** Connecting the Jumper Wires

StarFive

**Step 3** Connect the power adapter to a power socket.

### 5.1.2 Software Setup

Before you update the Bootloader, ddr init and u-boot, you need to have a serial communication software on your computer in order to communicate with the VisionFive.

Follow the steps below according to your operating system.

**For Windows:**

1. Install Tera Term for windows as described in *Installing Tera Term*.
2. Update `bootloader`, `u-boot`, and `ddr init` by:
   - using Xmodem as described in the *Using Xmodem* section.
   - or using Ethernet as described in the *Using Ethernet* section.

**For Mac/Linux:**

1. Install minicom on Mac/Linux as described in *Installing Minicom*.
2. Update `bootloader`, `u-boot`, and `ddr init` by:
   - using Xmodem as described in the *Using Xmodem* section.
   - or using Ethernet as described in the *Using Ethernet* section.

**For Windows**

**Installing Tera Term**

**Steps:**

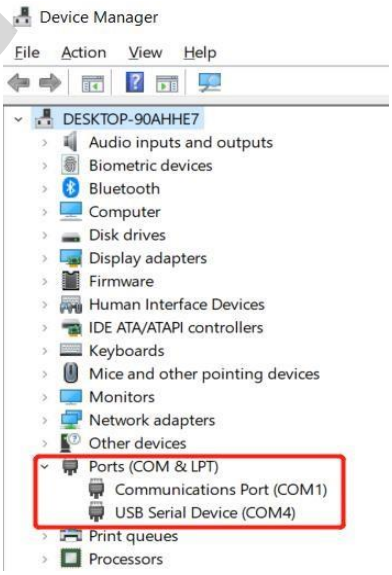
**Step 1** Download Tera Term by visiting this link.

**Step 2** Install Tera Term on the PC.

**Step 3** Connect the USB-to-Serial converter to the PC.

**Step 4** Open **Device Manager** by typing `Device Manager` in the windows search box.

**Step 5** Click the drop-down arrow from **Ports (COM & LPT)** and find the name of the connected serial port (e.g.: **COM4**).




StarFive

**Figure 5-2** Example

**Step 6** Open Tera Term software.

**Step 7** Navigate to **Setup** > **Serial port...**

**Step 8** Configure the settings as follows:

- **Port:** COM4 (choose your COM port)
- **Speed:** 115200

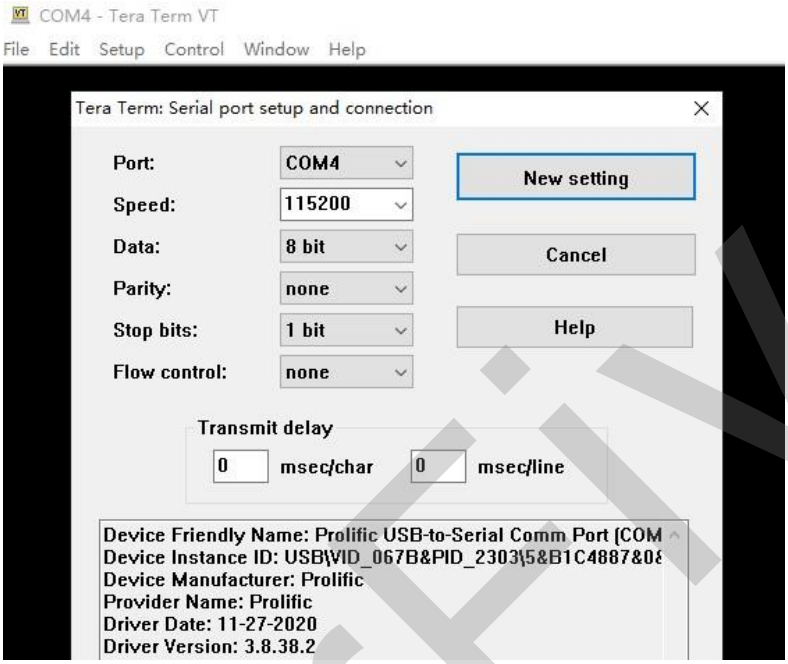



**Figure 5-3** Example

**Step 9** Click **New** > **open**.

**Using Xmodem**

**Step 1** After the hardware connections mentioned above, power on the VisionFive and you will see the startup information as follows.

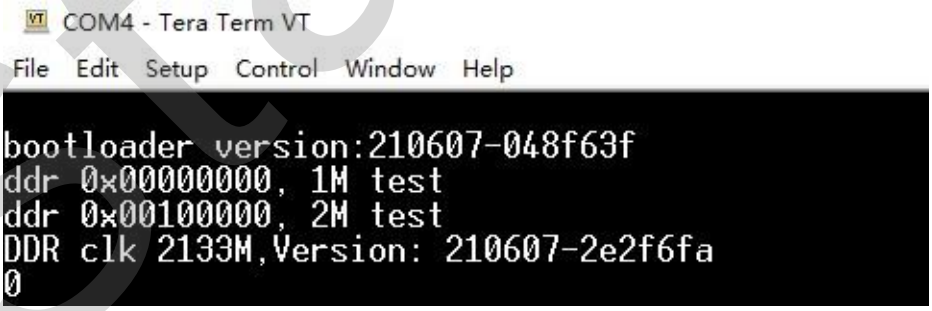



**Figure 5-4** Startup Information Example

**Step 2** Press any key as soon as it starts up to enter the upgrade menu. In this menu, you can only update u-boot.

starFive

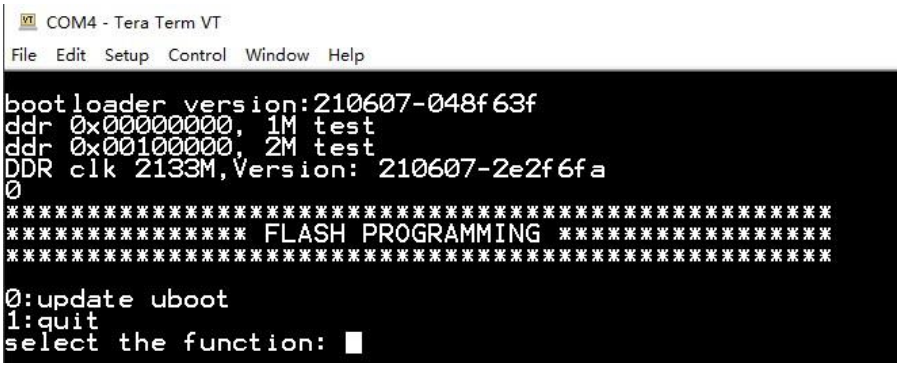



**Figure 5-5** Update Uboot

**Step 3** Type **root@s5**t and press **Enter** to enter the extended version of the upgrade menu. In this menu, you can update u-boot, bootloader and ddr init.

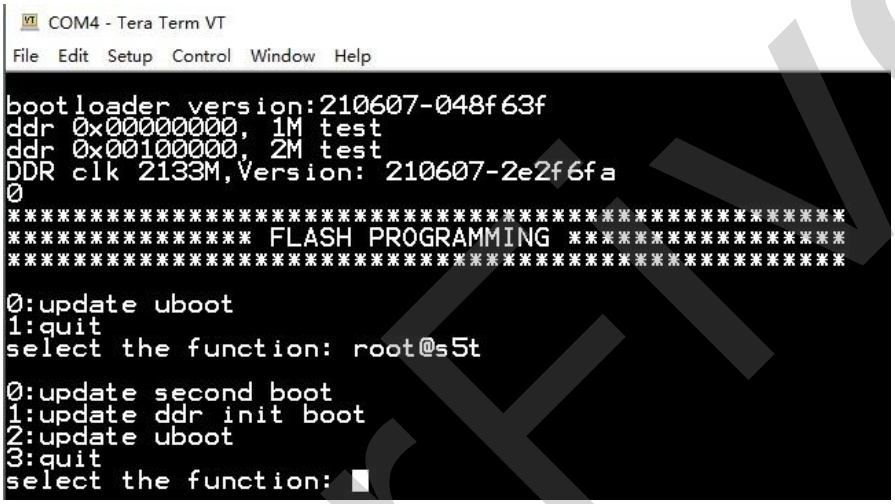



**Figure 5-6** Example Output

**Step 4** Type **0** and press Enter to update the bootloader.

**Step 5** Navigate to **File** > **Transfer** > **XMODEM** > **Send...** and choose the following file from the .zip file we downloaded before:

`bootloader-JH7100-211102.bin.out`

**Step 6** Repeat the steps 4 and 5 to update the ddr init as well according to the following:

Type **1** - update ddr init [Filename: `ddrinit-2133-211102.bin.out`]

**Step 7** Repeat the steps 4 and 5 to update the u-boot as well according to the following:

Type **2** - update u-boot [Filename: `fw_payload.bin.out`]

**Using Ethernet**

**Step 1** Connect one end of an Ethernet cable to the VisionFive RJ45 connector, and connect the other end of the cable to a router.

**Step 2** Download TFTPD64 from tftpd64.

**Step 3** Install TFTPD64 on your PC.

**Step 4** Open TFTPD64 and configure the **Current Directory**, which specifies the file path to store `bootloader`, `ddr init`, and `u-boot` files. The following figure shows an example configuration:

StarFive

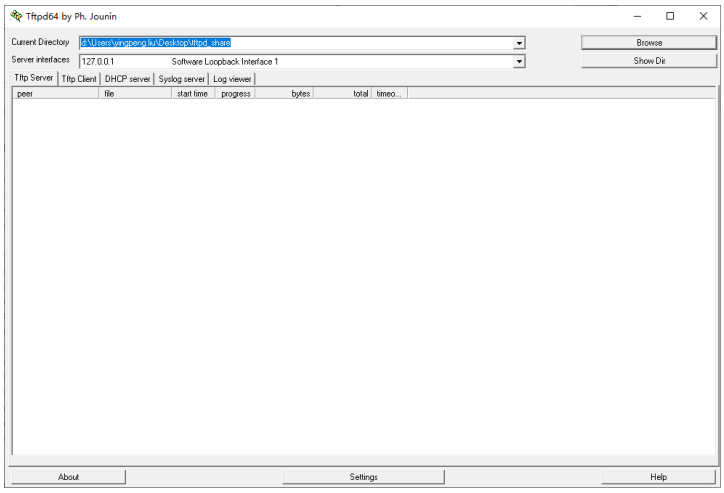



**Figure 5-7** Example Settings

**Step 5** Power on the VisionFive and wait until it enters the u-boot mode.

**Step 6** Configure the environment variables by executing:

```
setenv ipaddr 192.168.120.200;setenv serverip 192.168.120.12
```

> Generally, the default IP of a router is 192.168.120.1. In this case, use the server IP as the IP assigned by the DHCP server of the router and use the VisionFive IP as `192.168.120.xxx`. However, if your router IP is different (for example, `192.168.2.1`), make sure the server IP and VisionFive IP are in the same IP domain (for example, `192.168.2.xxx`).

**Step 7** Check the connectivity by pinging the host PC from VisionFive:

Example Command:

```
ping 192.168.120.12
```

**Result:**

The following output indicates that the host PC and VisionFive has established communication on the same network.

```
VisionFive #ping 192.168.120.12
Speed: 1000, full duplex
Using dwmac.10020000 device
host 192.168.120.12 is alive
VisionFive #
```

**Figure 5-8** Example Output

**Step 8** Connect to SPI Flash:

```
sf probe
```

**Example Output:**

```
VisionFive #sf probe
SF: Detected gd25lq128 with page size 256 Bytes, erase size 4 KiB, total 16 MiB
```

**Figure 5-9** Example Output

> **Information:**
>
> In the commands of the following step 9 to 11:

- `0x90000000` refers to the ddr address
- `192.168.120.12` refers to the tftp server IP
- `0x0` refers to the SPI flash offset for bootloader.
- `0x10000` refers to the SPI flash offset for ddrinit.
- `0x40000` refers to the SPI flash offset for u-boot.

If VisionFive fails to boot up after restart, refer to the *Appendix C: Recovering the Bootloader* chapter to recover.

**Step 9** Update bootloader:

```
tftpboot 0x90000000 192.168.120.12:bootloader-JH7100-211102.bin.out
sf update 0x90000000 0x0 ${filesize}
```

**Example Output:**

```
VisionFive #tftpboot 0x90000000 192.168.120.12:bootloader-JH7100-211102.bin.out
Speed: 1000, full duplex
Using dwmac.10020000 device
TFTP from server 192.168.120.12; our IP address is 192.168.120.200
Filename 'bootloader-JH7100-211102.bin.out'.
Load address: 0x90000000
Loading: ##################################################  9.2 KiB
         485.4 KiB/s
done
Bytes transferred = 9456 (24f0 hex)
VisionFive #sf update 0x90000000 0x0 ${filesize}
device 0 offset 0x0, size 0x24f0
0 bytes written, 9456 bytes skipped in 0.3s, speed 1613824 B/s
VisionFive #
```

**Figure 5-10** Example Output

**Step 10** Update ddr init:

```
tftpboot 0x90000000 192.168.120.12:ddrinit-2133-211102.bin.out
sf update 0x90000000 0x10000 ${filesize}
```

**Example Output:**

```
VisionFive #tftpboot 0x90000000 192.168.120.12:ddrinit-2133-211102.bin.out
Speed: 1000, full duplex
Using dwmac.10020000 device
TFTP from server 192.168.120.12; our IP address is 192.168.120.200
Filename 'ddrinit-2133-211102.bin.out'.
Load address: 0x90000000
Loading: ##################################################  85.5 KiB
         3.5 MiB/s
done
Bytes transferred = 87540 (155f4 hex)
VisionFive #sf update 0x90000000 0x10000 ${filesize}
device 0 offset 0x10000, size 0x155f4
0 bytes written, 87540 bytes skipped in 0.24s, speed 3320035 B/s
VisionFive #
```

**Figure 5-11** Example Output

**Step 11** Update u-boot:

```
tftpboot 0x90000000 192.168.120.12:fw_payload_vision-five.bin.out
sf update 0x90000000 0x40000 ${filesize}
```

**Example Output:**

```
VisionFive #tftpboot 0x90000000 192.168.120.12:fw_payload_visionfive.bin.out
Speed: 1000, full duplex
Using dwmac.10020000 device
TFTP from server 192.168.120.12; our IP address is 192.168.120.200
Filename 'fw_payload_visionfive.bin.out'.
Load address: 0x90000000
Loading: #################################################  2.9 MiB
         8.1 MiB/s
done
Bytes transferred = 3025228 (2e294c hex)
VisionFive #sf update 0x90000000 0x40000 ${filesize}
device 0 offset 0x40000, size 0x2e294c
1075532 bytes written, 1949696 bytes skipped in 8.442s, speed 366781 B/s
VisionFive #
```

**Figure 5-12** Example Output

**Step 12** Restart VisionFive to make the updates take effect.

**For Mac/Linux**

**Installing Minicom**

**Steps:**

**Step 1** Open a terminal window on Mac/Linux.

**Step 2** Type the following to update the packages list.

```
sudo apt-get update
```

**Step 3** Type the following to install minicom.

```
sudo apt-get install minicom
```

**Step 4** Connect the USB-to-Serial converter to the PC.

**Step 5** Type the following in the terminal to view the connected serial devices.

```
dmesg | grep tty
```

```
ryan@ubuntu:~$ dmesg|grep tty
[    1.030436] printk: console [tty0] enabled
[    2.572012] 00:05: ttyS0 at I/O 0x3f8 (irq = 4, base_baud = 115200) is a 16550A
[   95.234558] usb 3-2: pl2303 converter now attached to ttyUSB0
```

**Figure 5-13** Example Output

**Step 6** Connect to the serial device by typing the following.

```
sudo minicom -D /dev/ttyUSB0 -b 115200
```

**Information:**

The baud rate is set to **115200**.

```
ryan@ubuntu:~$ sudo minicom -D /dev/ttyUSB0 -b 115200
[sudo] password for ryan:

Welcome to minicom 2.7.1

OPTIONS: I18n
Compiled on Aug 13 2017, 15:25:34.
Port /dev/ttyUSB0, 23:23:12

Press CTRL-A Z for help on special keys
```

**Figure 5-14** Example

**Using Xmodem**

**Step 1** After the hardware connections mentioned above, power on the VisionFive and you will see the startup information as follows.

StarFive


**Figure 5-15** Example

**Step 2** Press any key as soon as it starts up to enter the upgrade menu. In this menu, you can only update u-boot.


**Figure 5-16** Example

**Step 3** Type root@s5t and press Enter to enter the extended version of the upgrade menu, and in this menu, you can update u-boot, bootloader and ddr init.

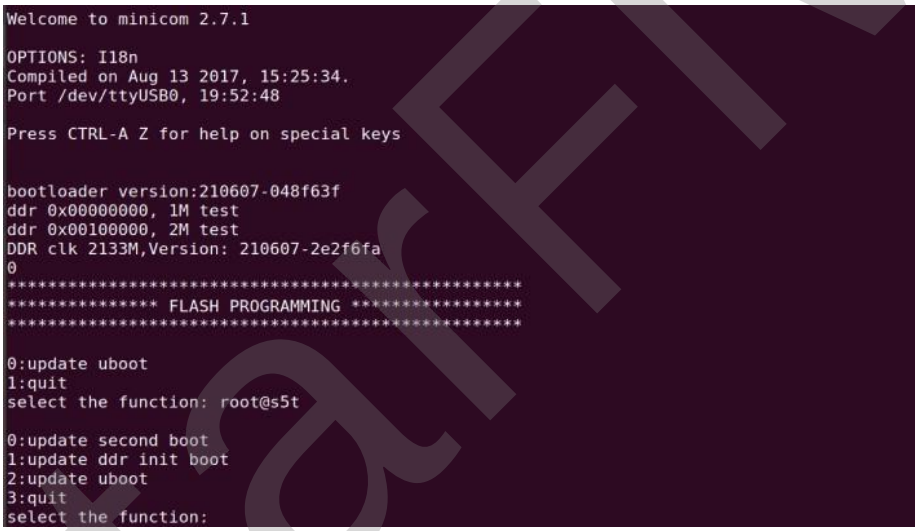



**Figure 5-17** Example

**Step 4** Type **0** and press **Enter** to update the bootloader.

**Step 5** Press **Ctrl+A** and then press **S** to enter upload mode.

**Step 6** Select **xmodem** and press **Enter**.

StarFive

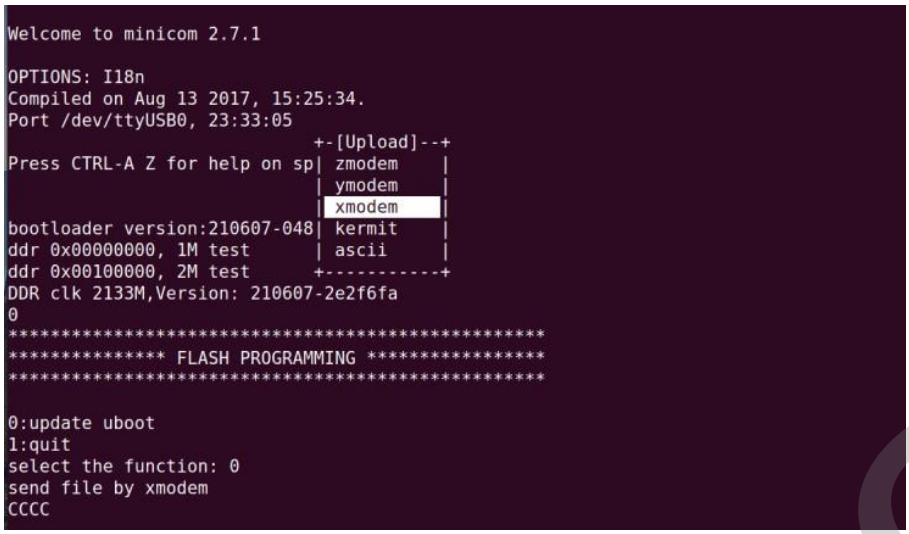



**Figure 5-18** Example

**Step 7** Select **Goto** from the bottom tab menu and press **Enter**.

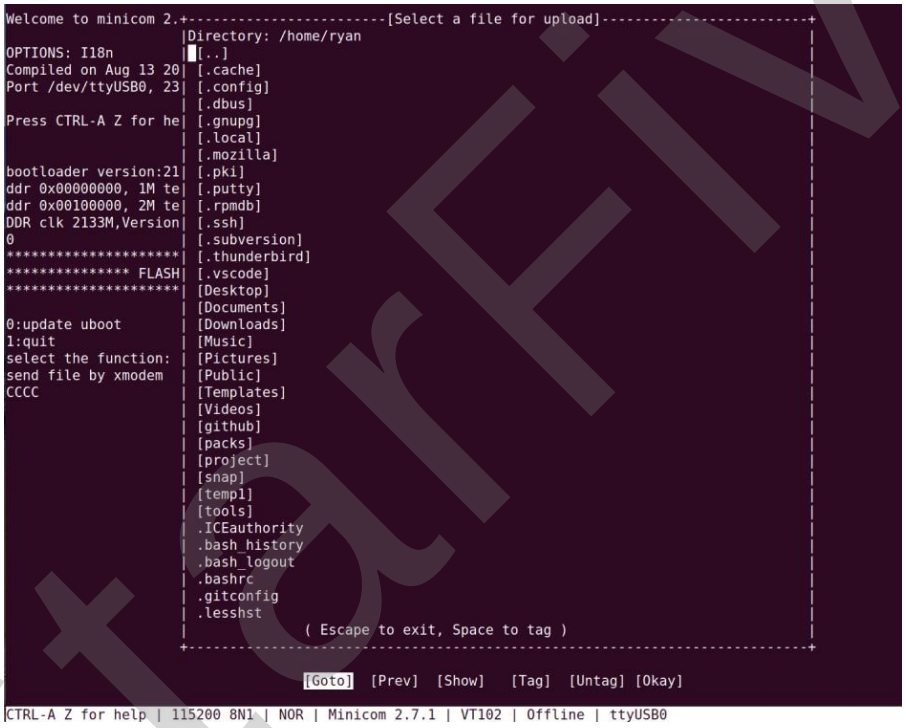



**Figure 5-19** Example

**Step 8** Enter the directory path and press Enter.

StarFive

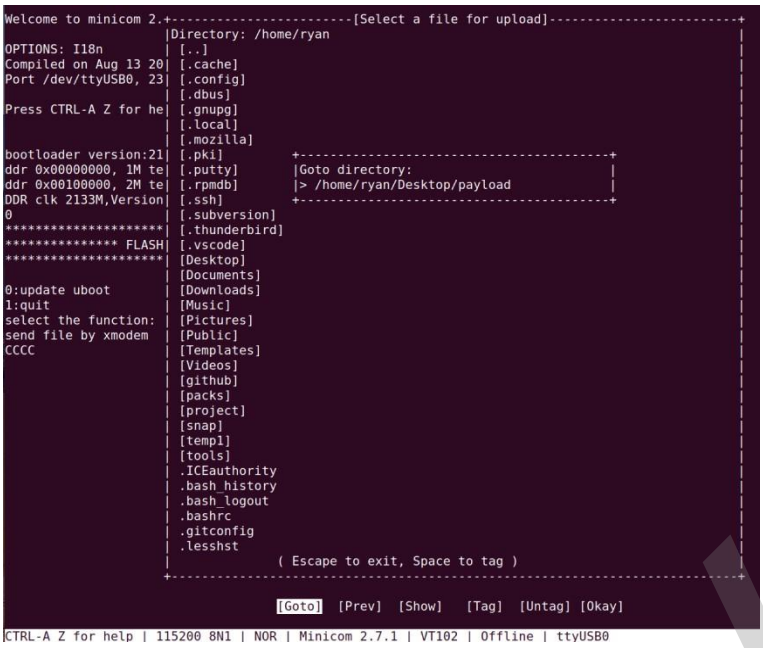



**Figure 5-20** Example

**Step 9** Select **bootloader-JH7100-211102.bin.out** by navigating using arrow keys, press **Space** and press **Enter**.

**Step 10** Repeat the steps from 4 to 9 to update the ddr init as well according to the following:

Type **1** - update ddr init [Filename: `ddrinit-2133-211102.bin.out`]

**Step 11** Repeat the steps from **4** to **9** to update the u-boot as well according to the following:

Type **2** - update u-boot [Filename: `fw_payload.bin.out`]

**Using Ethernet**

**Step 1** Connect one end of an Ethernet cable to the VisionFive RJ45 connector, and connect the other end of the cable to a router.

**Step 2** Install a tftp server on the host PC by executing:

```
sudo apt-get update
sudo apt install tftpd-hpa
```

**Step 3** Check the server status:

```
sudo systemctl status tftpd-hpa
```

**Step 4** Execute the following to enter the tftp server:

```
sudo nano /etc/default/tftpd-hpa
```

**Step 5** Configure the tftp server as follows:

```
TFTP_USERNAME="tftp"
TFTP_DIRECTORY="/home/user/Desktop/tftp_share"
TFTP_ADDRESS=":69"
TFTP_OPTIONS="--secure"
```

**Information:**

TFTP_DIRECTORY refers to the directory to store `bootloader`, `ddr init`, and `u-boot` files.

**Step 6** Restart the tftp server by executing:

```
sudo systemctl restart tftpd-hpa
```

**Step 7** Power on the VisionFive and wait until it enters the u-boot mode.

**Step 8** Configure the environment variables by executing:

```
setenv ipaddr 192.168.120.200;setenv serverip 192.168.120.12
```

Generally, the default IP of a router is `192.168.120.1`. In this case, use the server IP as the IP assigned by the DHCP server of the router and use the VisionFive IP as `192.168.120.xxx`. However, if your router IP is different (for example, `192.168.2.1`), make sure the server IP and VisionFive IP are in the same IP domain (for example, `192.168.2.xxx`).

**Step 9** Check the connectivity by pinging the host PC from VisionFive:

**Example Command:**

```
ping 192.168.120.12
```

**Result:**

The following output indicates that the host PC and VisionFive has established communication on the same network.

```
VisionFive #ping 192.168.120.12
Speed: 1000, full duplex
Using dwmac.10020000 device
host 192.168.120.12 is alive
VisionFive #
```

**Figure 5-21** Example Output

**Step 10** Connect to SPI Flash:

```
sf probe
```

**Example Output**:

```
VisionFive #sf probe
SF: Detected gd25lq128 with page size 256 Bytes, erase size 4 KiB, total 16 MiB
```

**Figure 5-22** Example Output

**Information:**

In the commands of the following step 9 to 11:

- `0x90000000` refers to the ddr address
- `192.168.120.12` refers to the tftp server IP
- `0x0` refers to the SPI flash offset for bootloader.
- `0x10000` refers to the SPI flash offset for ddrinit.
- `0x40000` refers to the SPI flash offset for u-boot.

If VisionFive fails to boot up after restart, refer to the *Appendix C: Recovering the*

StarFive

*Bootloader* chapter to recover.

**Step 11** Update bootloader:

```
tftpboot 0x90000000 192.168.120.12:bootloader-JH7100-211102.bin.out
sf update 0x90000000 0x0 ${filesize}
```

**Example Output:**

```
VisionFive #tftpboot 0x90000000 192.168.120.12:bootloader-JH7100-211102.bin.out
Speed: 1000, full duplex
Using dwmac.10020000 device
TFTP from server 192.168.120.12; our IP address is 192.168.120.200
Filename 'bootloader-JH7100-211102.bin.out'.
Load address: 0x90000000
Loading: #################################################  9.2 KiB
         1.3 MiB/s
done
Bytes transferred = 9456 (24f0 hex)
VisionFive #sf update 0x90000000 0x0 ${filesize}
device 0 offset 0x0, size 0x24f0
0 bytes written, 9456 bytes skipped in 0.3s, speed 1613824 B/s
VisionFive #
```

**Figure 5-23** Example Output

**Step 12** Update ddr init:

```
tftpboot 0x90000000 192.168.120.12:ddrinit-2133-211102.bin.out
```

**Example Output:**

```
VisionFive #tftpboot 0x90000000 192.168.120.12:ddrinit-2133-211102.bin.out
Speed: 1000, full duplex
Using dwmac.10020000 device
TFTP from server 192.168.120.12; our IP address is 192.168.120.200
Filename 'ddrinit-2133-211102.bin.out'.
Load address: 0x90000000
Loading: #################################################  85.5 KiB
         7 MiB/s
done
Bytes transferred = 87540 (155f4 hex)
VisionFive #sf update 0x90000000 0x10000 ${filesize}
device 0 offset 0x10000, size 0x155f4
0 bytes written, 87540 bytes skipped in 0.24s, speed 3320035 B/s
VisionFive #
```

**Figure 5-24** Example Output

**Step 13** Update u-boot:

```
tftpboot 0x90000000 192.168.120.12:fw_payload_vision-five.bin.out
sf update 0x90000000 0x40000 ${filesize}
```

**Example Output:**

```
VisionFive #tftpboot 0x90000000 192.168.120.12:fw_payload_visionfive.bin.out
Speed: 1000, full duplex
Using dwmac.10020000 device
TFTP from server 192.168.120.12; our IP address is 192.168.120.200
Filename 'fw_payload_visionfive.bin.out'.
Load address: 0x90000000
Loading: #################################################  2.9 MiB
         8.3 MiB/s
done
Bytes transferred = 3025228 (2e294c hex)
VisionFive #sf update 0x90000000 0x40000 ${filesize}
device 0 offset 0x40000, size 0x2e294c
0 bytes written, 3025228 bytes skipped in 0.821s, speed 3759506 B/s
VisionFive #
```

**Figure 5-25** Example Output

**Step 14** Restart VisionFive to make the updates take effect.

StarFive

**Information:**

The methods to compile and update u-boot are included in the following: VisionFive_Software_Technical_Reference_Manual.

StarFive

# 6 Appendix C: Recovering the Bootloader

The bootloader is stored inside the SPI flash storage. There may be situations where you accidentally emptied the flash or if the flash is damaged on your VisionFive. In these situations, it's better to reset the bootloader. Follow the steps below to load JH7100_recovery_boot.bin to on-chip SRAM, run it, and then flash bootloader, ddr init and u-boot.

## 6.1 Hardware Setup

Connect the jumper wires between the USB-to-Serial converter and the DEBUG header of the VisionFive as follows.

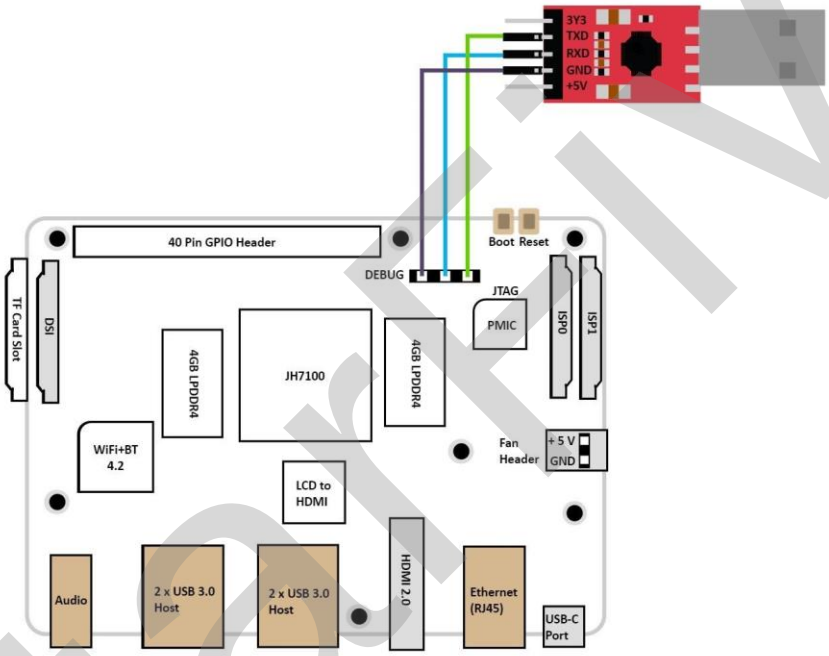



**Figure 6-1** Connecting the Debug Header

## 6.2 Software Setup

Before you recover the bootloader, you need to have a serial communication software on your computer in order to communicate with the VisionFive. We will use:

- a software called Tera Term for windows as described in *For Windows*.
- a Bootloader recovery and updater tool for Mac/Linux as described in *For Mac/Linux*.

Follow the steps below according to your operating system.

### 6.2.1 For Windows

**Step 1** Open Tera Term software.

**Step 2** Navigate to **Setup** > **Serial port...**

**Step 3** Configure the settings as follows.

StarFive

- **Port**: COM4 (choose your COM port)
- **Speed**: 9600

**Step 4** Click on **New open**.

**Step 5** Press the BOOT button while turning on VisionFive.

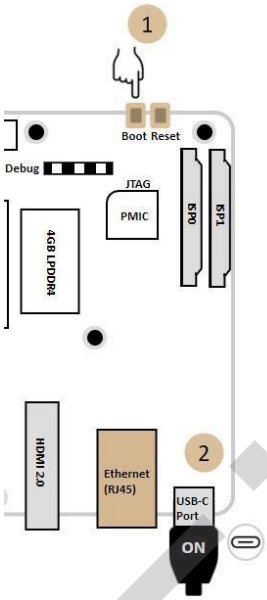



**Figure 6-2** Pressing the Boot Button

**Step 6** You will see the following output on Tera Term.

```
(C) SiFive
```

**Step 7** Type the following.

```
load 0x18000000
```

**Result:**

You will see an output like this:

```
CCCCCCCCCCCCCCC
```

**Step 8** Navigate to **File** > **Transfer** > **XMODEM** > **Send...** and choose the following file from the .zip file we downloaded before: `JH7100_recovery_boot.bin`.

**Result:**

You will see an output like this after the transfer is complete.

```
Load file ok.
```

**Step 9** Type the following.

```
do 0x18000000
```

**Result:**

You will see an output like this:

StarFive

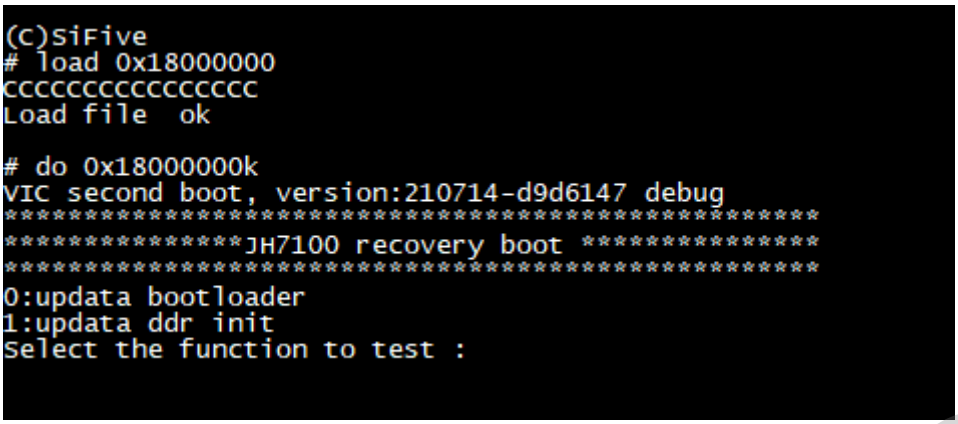



**Figure 6-3** Example Output

**Step 10** Type 0 and press Enter to update the bootloader.

**Step 11** Navigate to **File** > **Transfer** > **XMODEM** > **Send...** and choose the following file from the `.zip` file we downloaded before:

```
bootloader-JH7100-211102.bin.out
```

**Step 12** Repeat the steps **10** and **11** to update the ddr init according to the following.

Type 1 - update ddr init [Filename: `ddrinit-2133-211102.bin.out`]

### 6.2.2 For Mac/Linux

**Step 1** Clone the following GitHub repo which includes a bootloader recovery and updater tool.

```
git clone https://github.com/xypron/JH71xx-tools/
```

**Step 2** Navigate to the cloned repo and build the tool.

```
cd JH71xx-tools
gcc -o jh7100-recover jh7100-recover.c
```

**Step 3** Copy `JH7100_recovery_boot.bin`, `bootloader` and `ddr init` to the `JH71xx-tools` directory.

**Information:**

This step is not a must, but it makes it more convenient in the following steps when we point to the file locations.

**Step 4** Type the following in the terminal to view the connected serial devices.

```
dmesg | grep tty
```

**Example Output:**

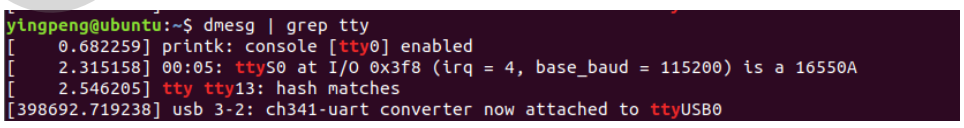



**Figure 6-4** Example Output

**Step 5** Type the following and it will wait for bootloader mode.

```
sudo ./jh7100-recover -D /dev/ttyUSB0 -r jh7100_recov-
ery_boot.bin -b bootloader-JH7100-211102.bin.out -d ddrinit-
```

```
2133-211102.bin.out
```

**Example Output:**

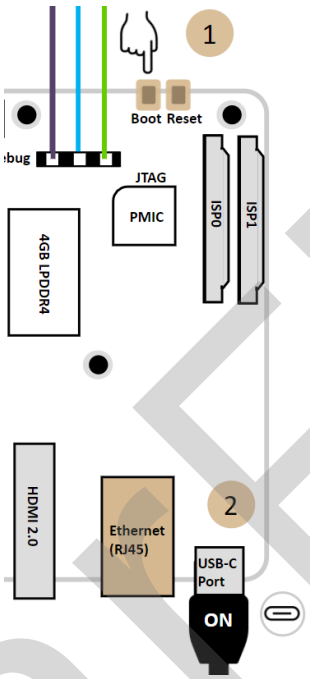
```
yingpeng@ubuntu:~/Desktop/github/JH71xx-tools$ sudo ./jh7100-recover -D /dev/ttyUSB0 -r jh7100_recovery_boot.bin -b bootloader-JH7100-21
1102.bin.out -d ddrinit-2133-211102.bin.out
Waiting for bootloader mode on /dev/ttyUSB0...
```

**Figure 6-5** Example Output

**Information:**

You may change the serial port according to yours and also the file locations if you haven't moved them into the JH71xx-tools directory.

**Step 6** Press on the **BOOT** button while turning on VisionFive to enter bootloader mode.




**Example Result:**

If you see the following output, you have successfully updated bootloader and ddr init.

StarFive


**Figure 6-6** Example Output

StarFive

# 7 Appendix D: GitHub Repository

The following table describes the links to GitHub repository:

**Table 7-1 GitHub Repository**

| Type | Item | Description | Repositories |
|---|---|---|---|
| StarFive firmware | Boot_recovery | Binary for recovering SPI flash | starfive-tech/bootloader_recovery |
| | secondboot | First stage bootloader | starfive-tech/JH7100_secondBoot |
| | ddrinit | First stage bootloader | starfive-tech/JH7100_ddrinit |
| RISC-V SBI | openSBI | RISC-V specific | riscv/opensbi |
| Bootloader | u-boot | Universal boot loader | starfive-tech/u-boot |
| Kernel | Linux | Linux Kernel | starfive-tech/linux |
| Distro | Fedora Image | Fedora 33 image | starfive-tech/Fedora_on_StarFive |

StarFive