

# FRC Vision on the Banana Pi M2 Berry

## Introduction

In many FRC challenges, vision targets are provided to help the robot identify key locations on the playing field. Many teams incorporate web-cams into their robot designs to capture images of the field and process them looking for the vision targets. Vision processing takes considerable CPU time and memory bandwidth, so it is now common for teams to offload this processing onto a Raspberry Pi. There is even an FRCVision Raspberry Pi Image which includes many of the libraries, and utilities needed to do effective vision processing and tie into the FRC Control system.

While the FRCVision image works very well with USB and web-cams, these types of cameras have limitations for robot control: poor synchronization, limited ability to set exposure control/shutter times in realtime, latency, lens focus, etc.

One alternative is to connect a video sensor directly to the “camera input” on the Raspberry Pi. In theory, this yields benefits as exposure control/shutter time can be controlled on a frame by frame basis, latency is minimized, and in some cases, lenses can be replaced. Unfortunately, the “camera input” on the Raspberry Pi uses a proprietary vision processing engine, and there are very few video sensors supported by the Raspberry Pi firmware: the Omnivision OV5647, and the Sony IMX219.

## Banana Pi M2 Berry

The success of the Raspberry Pi has spurred a number of similar single board processors with differing capabilities. One of these alternative boards is the Banana Pi M2 Berry which:

- uses a quad-core ARM processor (Allwinner V40 vs Broadcom)
- has the same form factor as the Raspberry Pi Model 3B and 3B+
- includes a 40 pin I/O header (many of the pins on the 40 pin header can be configured for various functions in software)
- has built in Ethernet, WiFi, Bluetooth
- also includes a SATA connector for connecting a hard-drive
- has a camera-input connector (although the connector isn't compatible with the Raspberry Pi cameras)
- can support a 2<sup>nd</sup> camera through pins on the 40 pin I/O header

Unlike the Broadcom processor used with the Raspberry Pi family, the Allwinner V40 documentation available online includes the specifications for the camera input processor. In theory, this means that it is much easier to support a variety of video sensors, but for some reason, there is only one camera module compatible with the Banana Pi family Camera connector (uses an OV5640 image sensor).

There are other Banana Pi boards with various capabilities (and prices), however, they have different Allwinner processors, and will take much more effort to configure and test the Linux kernel drivers. The information provided below can be used as a reference for using these other, but will take significantly more effort.

From here on, any reference to the Banana Pi will specifically mean the Banana Pi M2 Berry.

## Choosing an Image Sensor

There are many image sensors/camera modules to choose from. Things to consider are:

- Hardware Interfaces

- The Allwinner V40 CSI (Camera Sensor Interface) supports an 8-bit parallel interface with Horizontal, Vertical and Pixel Clock signals.
- A sensor with a 10 bit parallel interface can easily be connected by leaving the least significant parallel signals un-connected.
- Resolution
  - The number of rows and columns of pixels.
  - More is not necessarily better as the ARM processors have limited computing power, and you will likely have to scale down the image to something on the order of 640x480 or 320x240 anyways
- Colour or Black and White
  - Most teams use a “coloured illuminator” to help distinguish the retro-reflective targets from the rest of the field. Many teams use Green as there are often Red and Blue lights, surfaces (and robot bumpers) that could get mistaken for targets.
- Frame Rate
  - The number of frames/second that the sensor is capable of sampling.
  - Some sensors have the capability to “sub-sample”, or bin pixels to reduce the effective resolution and increase Frame Rate
  - The processing power of the ARM processors is potentially the limiting factor
  - For video presented on the driverstation, network bandwidth will also limit the effective Frame Rate
- Shutter
  - Most image sensors use a “rolling shutter”. Only part (usually one line) of the image is captured at one time. This is why pictures of spinning propellers appear twisted.
  - Although they are less common, sensors with “global shutters” capture the entire image at one time. Aside from some blur (depending on exposure time), an image captured with a global shutter will not be oddly distorted.

To work with the Banana Pi, the sensor must have a parallel interface. There are many image sensors/camera modules that use this interface. One very inexpensive sensor is the OV7670. These sensors are often connected to an Arduino and can be purchased for less than CA\$5.00 (<https://www.aliexpress.com/item/32787576233.html?spm=a2g0s.9042311.0.0.27424c4d2uDxGr>). Boards such as this can be easily wired to the appropriate CSI1 pins on the 40 pin header to achieve a decent colour image at 30Hz.

At the other end of the cost spectrum (~CA\$70), the MT9V034 colour sensor provides a raw Bayer Pattern colour image at 60Hz. This sensor has a 10 bit parallel interface, and a global shutter. The module available from <https://www.uctronics.com/index.php/arducam-cmos-mt9v034-1-3-inch-0-36mp-monochrome-camera-module.html> can also be connected to the CSI pins on the 40 pin GPIO header.

## Getting Started

Before attempting to use the Banana Pi, it is important to realize that this tutorial should be sufficient to get you up and running, but it does require a reasonable level of knowledge of the Linux operating system. If you have never used Linux, or don't know how to program using C++, you may find this tutorial challenging.

## Installing the FRCVision Image

Although the processors on the Banana Pi M2 Berry and the Raspberry Pi are both ARM based, they have many differences, so its not possible to simply put a micro-sd card for the Raspberry Pi into a Banana Pi and expect it to work. It is, however, fairly easy to convert a micro-sd card containing the FRCVision image to work with the Banana Pi M2 Berry.

Download and install the FRCVSION\_image-v2020.1.1.zip file onto a micro SD card using the instructions at <https://github.com/wpilibsuite/FRCVision-pi-gen/releases>.

In addition to what is needed to create the Raspberry Pi FRCVision micro-sd card image, you will need to download the amd64 Minimal Installation CD from <https://gentoo.org/downloads>, burn the ISO to a dvd or Flash Drive, and boot a computer (or a virtual machine) using that image. Then follow the steps below:

- download the Raspberry Pi to Banana Pi converter package by typing  
`wget https://2702rebels.com/resources/2020/rpi2bpi.tgz`
- extract the package by typing  
`tar xvzf rpi2bpi.tgz`
- change to the newly created directory using the command  
`cd rpi2bpi`
- insert flashed micro SD the into the micro SD to USB dongle into the computer
- find the device file for the micro SD file. To do this:  
enter the command  
`dmesg`

A lot of text will scroll by, but the last few lines should look similar to:

```
[ ] scsi 3:0:0:0: Direct-Access Generic STORAGE DEVICE 1404 PQ: 0 ANSI: 6
[ ] sd 3:0:0:0: Attached scsi generic sg2 type 0
[ ] sd 3:0:0:0: [sdb] 30318592 512-byte logical blocks: (15.5 GB/14.5 GiB)
[ ] sd 3:0:0:0: [sdb] Write Protect is off
[ ] sd 3:0:0:0: [sdb] Mode Sense: 21 00 00 00
[ ] sd 3:0:0:0: [sdb] Write cache: disabled, read cache: enabled, doesn't support DPO or FUA
[ ] sdb: sdb1 sdb2
[ ] sd 3:0:0:0: [sdb] Attached SCSI removable disk
```

The device file in this case is `/dev/sdb`

- update the flash image using the command  
`./make-bpi-image.sh device_file`  
where `device_file` is the device file found in the step above.
- Once the script completes, you can remove the micro SD card from the dongle and plug it into the Banana Pi, and power it on. The 1<sup>st</sup> boot will take a few minutes as the filesystem is expanded to fill the micro SD card, but subsequent boots will be much faster.
- At this point, the Banana Pi can be used the same way as the Raspberry Pi. You will likely want to use PUTTY or another SSH client to log into the Banana Pi command line. The login is `pi` and the password is `raspberry`
- In addition to the files installed with the regular Raspberry Pi FRCVision Image, a folder `FRC2702-files` is placed in the home directory by the `make-bpi-image.sh` script. A `README.txt` file in the `FRC2702-files` folder explains what the scripts and sub-folders contain, and how to build a kernel that supports your camera.

## A better way to connect the sensors

While it is possible to connect one of these sensors to the Banana Pi M2 Berry using Dupont leads (<https://www.aliexpress.com/item/32829131026.html?spm=a2g0s.9042311.0.0.27424c4dl6CbOP>), it is

not a very robust solution. Individual wires can get tugged on and will pull off the pins. A better solution is to develop a custom PCB. A sample board design that will permit 2 of the same sensor to a single Banana Pi M2 Berry can be found at <https://2702rebels.com/resources/2020/bpi-cam.zip> (a current software limitation prevents using 2 different image sensors with the Banana Pi M2 Berry).

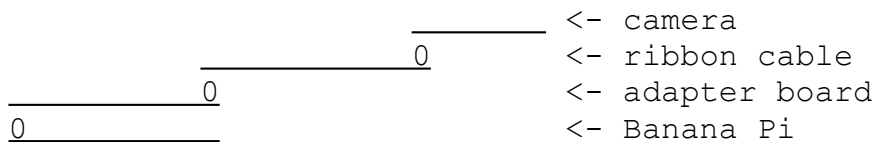
Within the zip file are two KiCad projects with different revisions of an adapter board. The Linux Kernel, Device Tree, and patches installed by the earlier scripts refer to the v1 board. The device tree files will need to be updated to reflect the updated GPIO pin numbering for the v2 board.

Notes on the V1 Board:

- This board has a few errors in signal names (PDWN vs PWDN) which resulted in incomplete signal networks.
- The footprint for the Arduino Pro Mini was to allow for debugging of the camera interface without needing an actual sensor
- The circuit board is a 4 layer board. Board manufacturers like PCBWay.com can make 10 of these boards for US\$49 + shipping.

Notes on the V2 Board:

- This board has the errors from the V1 board corrected.
- The camera connector headers have been “flipped” to allow for a ribbon cable with opposing connectors to be used to connect the sensors.



- A 2 layer version of this board is included in the project, but the gerber files required for manufacturing have not been generated. This 2 layer version of the board can be manufactured by PCBWay.com for US\$5.00 + shipping.
- The 30 pin flat-flex-cable connectors were removed as they were also removed from the MT9V034 camera board
- To make this design work on a 2 layer board, some of the signals (PWDN and RST) were re-assigned to different pins on the Banana Pi 40 pin GPIO header

## Disclaimer

The files and information presented are provided AS IS without warranty of any kind. While we may be able to answer some simple questions, we are not able to provide assistance with designing a new board, adapting these principles to other co-processors or image sensors...we have a robot to design and build :)