StarFive
赛昉科技

Using VisionFive to Make An LED Blink at the PWM Frequency

with Python

Application Note

Version: 1.1

Date: 2022/07/29

Doc ID: VisionFive-ANEN-011-1.1

Legal Statements

Important legal notice before reading this documentation.

PROPRIETARY NOTICE

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.

Contact Us

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

Website: http://www.starfivetech.com

Email: sales@starfivetech.com(sales) , support@starfivetech.com(support)

Preface

About this guide and technical support information.

About this document

This application note provides steps to use VisionFive's GPIO pins to make an LED blink at the PWM frequency.

Revision History

Table 0-1 Revision History

Version	Released	Revision
V1.1	2022-07-29	Added "cd" in the codeblock `cd /usr/local/lib64/python3.9/site-packages` to make it a complete command.

Notes and notices

The following notes and notices might appear in this guide:

- **Tip:**
  Suggests how to apply the information in a topic or step.

- **Note:**
  Explains a special case or expands on an important point.

- **Important:**
  Points out critical information concerning a topic or step.

- **CAUTION:**
  Indicates that an action or step can cause loss of data, security problems, or performance issues.

- **Warning:**
  Indicates that an action or step can result in physical harm or cause damage to hardware.

Contents

List of Tables

List of Figures

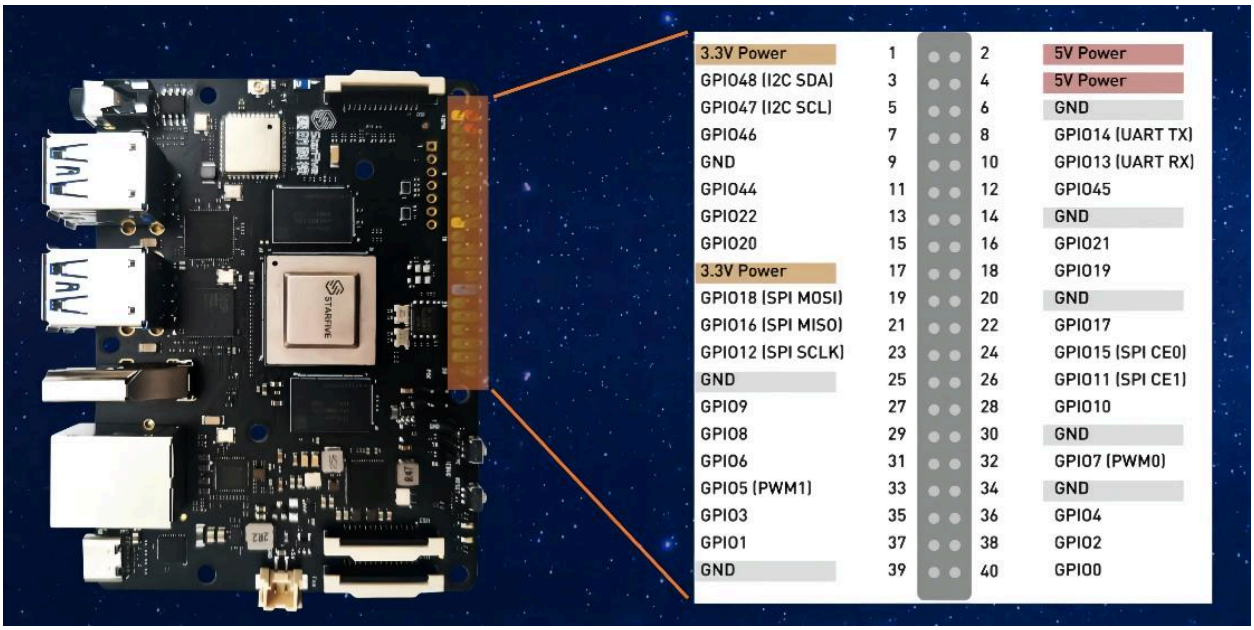
1. Introduction

This application note provides steps to use VisionFive's GPIO pins to make an LED blink at the PWM frequency.

1.1. 40-Pin Header Definition

The following figure shows the location of the 40-pin header. The VisionFive board is taken as an example:

Figure 1-1 40-Pin Definition



2. Preparation

Before executing the demo program, make sure you prepare the following:

2.1. Preparing Hardware

Prepare the following hardware items before running the demo code:

Table 2-1 Hardware Preparation

Type	M/O*	Item	Notes
General	M	StarFive single board computer	The following boards are applicable: • StarLight • VisionFive
General	M	• 16 GB (or more) micro-SD card • micro-SD card reader • Computer (Windows/MAC/Linux) • USB to serial converter (3.3 V I/O) • Ethernet cable • Power adapter (5 V / 3 A) • USB Type-C Cable	These items are used for flashing Fedora OS into a micro-SD card.
GPIO Demo (LED)	M	• An LED • A Breadboard • Two Male-Female jumper wires • 470 Ω color ring resistor	• LED stands for Light Emitting Diode, and glows when electricity is passed through it. The longer leg (known as the 'anode'), is always connected to the positive supply of the circuit. The shorter leg (known as the 'cathode') is connected to the negative side of the power supply, known as 'ground'. • Breadboard: See Breadboard Introduction *(on page 9)*. • Resistor: Resistors are a way of limiting the amount of elec-

Table 2-1 Hardware Preparation (continued)

Type	M/O*	Item	Notes
			tricity going through a circuit; specifically, they limit the amount of 'current' that is allowed to flow.

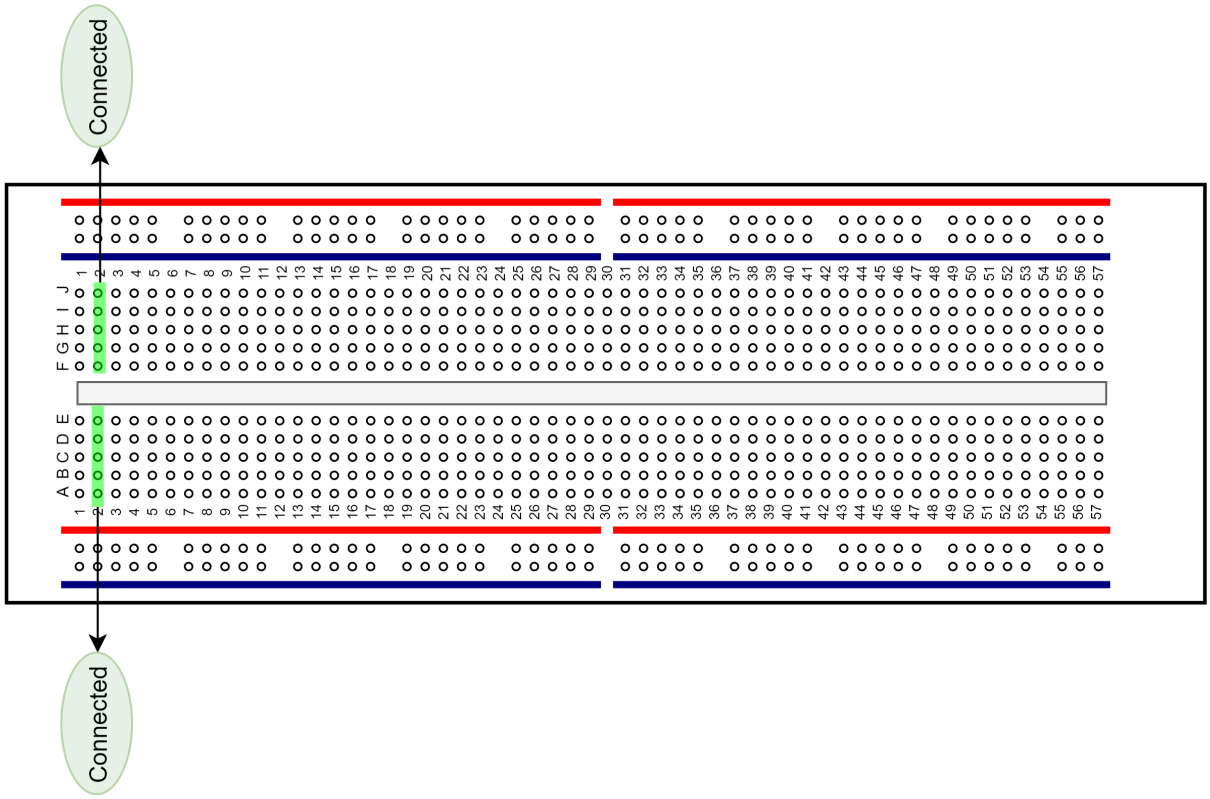
Note:
*: M: Mandatory, O: Optional

Breadboard Introduction

The breadboard is a way of connecting electronic components to each other without having to solder them together. They are often used to test a circuit design before creating a Printed Circuit Board (PCB). As shown in the following figure, there are two lines at the top and the bottom respectively of the breadboard. These two lines are used for power connection: the blue line is for negative and the red line is for positive. Besides, they are divided into two sections, and the holes in each section are connected.

In each column (from A to E, and F to J), holes are connected electrically. In each row (from 1 to 57), holes are not connected.

Figure 2-1 Breadboard Overview



2.1.1. Hardware Setup

The following table and figure describe how to connect LED to the 40-pin header:

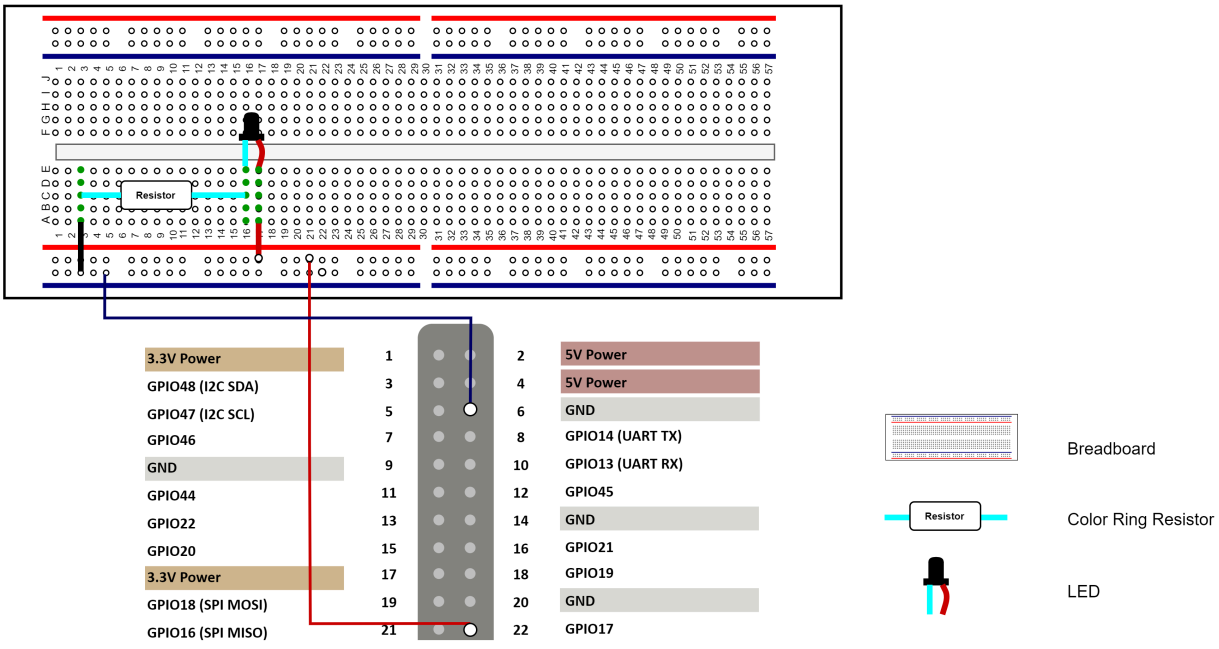
Table 2-2 Connect LED to the 40-Pin Header

LED	40-Pin GPIO Header	
	Pin Number	Pin Name
Positive	22	GPIO17
Negative	6	GND

Perform the following to connect the LED to the 40-pin GPIO Header:

1. Connect GPIO17 pin of VisionFive to the red line of the breadboard.
2. Set up the resistor as shown in the following figure.
3. Connect the longer leg of the LED to the red line of the breadboard.
4. Connect the shorter leg of the LED to the blue line of the breadboard.
5. Connect the GND pin of VisionFive to the blue line of the breadboard.

Figure 2-2 Connect LED to the 40-Pin Header



2.2. Preparing Software

Make sure the following procedures are performed:

1. Flash Fedora OS into a Micro-SD card as described in the *Flashing Fedora OS to a Micro-SD Card* section in the *VisionFive Single Board Computer Quick Start Guide*.
2. Log into the Fedora and make sure VisionFive is connected to the Internet. For detailed instructions, refer to the *Using SSH over Ethernet* or *Using a USB to Serial Converter* section in the *VisionFive Single Board Computer Quick Start Guide*.
3. Execute the `pip` command on VisionFive Fedora to install the `VisionFive.gpio` package:

   ```
   sudo pip install VisionFive.gpio
   ```

   Alternatively, you can execute the following command:

   ```
   sudo pip3 install VisionFive.gpio
   ```

4. (Optional) If you copy the source code to the local directory under VisionFive Fedora, execute the following commands under the source code directory:

   Tip:
   The source code can be downloaded by clicking the following link: VisionFive.gpio.

   ```
   sudo yum install python-devel python3-devel
   sudo python setup.py install
   ```

   Alternatively, you can execute the following command:

   ```
   sudo python3 setup.py install
   ```

3. Running Demo Code

To run the demo code, perform the following on VisionFive Fedora:

1. Locate to the directory where the test code, `pwm_led.py`, exists:

    a. Execute the following command to get the directory where `VisionFive.gpio` exists:

    ```
    pip show VisionFive.gpio
    ```

    Example Result:

    ```
    Location: /usr/local/lib64/python3.9/site-packages
    ```

    Note:

    The actual output depends on how the application is installed.

    b. Execute the following to enter the directory, for example, `/usr/local/lib64/python3.9/site-packages` as indicated in the previous step output:

    ```
    cd /usr/local/lib64/python3.9/site-packages
    ```

    c. Execute the following command to enter the `sample-code` directory:

    ```
    cd ./VisionFive/sample-code/
    ```

2. Under the `sample-code` directory, execute the following command to execute the demo code:

    ```
    sudo python pwm_led.py
    ```

    Alternatively, you can execute the following command:

    ```
    sudo python3 pwm_led.py
    ```

    Result:

    The LED blinks and the blink frequency will change according to the PWM frequency.

4. Demo Source Code

The Python source code of this demo is provided for reference purpose only.

`pwm_led.py`:

```
'''
Please make sure the LED is connected to the correct pins.
The following table describes how to connect LED to the 40-pin header.
-----------------------------------------
_______LED_________Pin Number_____Pin Name
    Positive          22          GPIO17
    Negative          6            GND
-----------------------------------------
'''

import time
import VisionFive.gpio as GPIO

led_pin = 17

#Configure the direction of led_pin as out.
GPIO.setup(led_pin, GPIO.OUT)
#Configure the voltage level of led_pin as high.
GPIO.output(led_pin, GPIO.HIGH)

#Configure the frequency as 10.
p = GPIO.PWM(led_pin, 10)
#Initialize the duty ratio as 0.
p.start(0)

try:
    #Change the LED blink frequency.
    while True:
        for dc in range(0, 101, 5):
            #Change the duty ratio from 0 to 100. Step size: 5
            p.ChangeDutyRatio(dc)
            time.sleep(1)
        for dc in range(100, -1, -5):
            #Change the duty ratio from 100 to 0. Step size: -5
            p.ChangeDutyRatio(dc)
            time.sleep(1)
except KeyboardInterrupt:
    pass

p.stop()
GPIO.cleanup(led_pin)
```