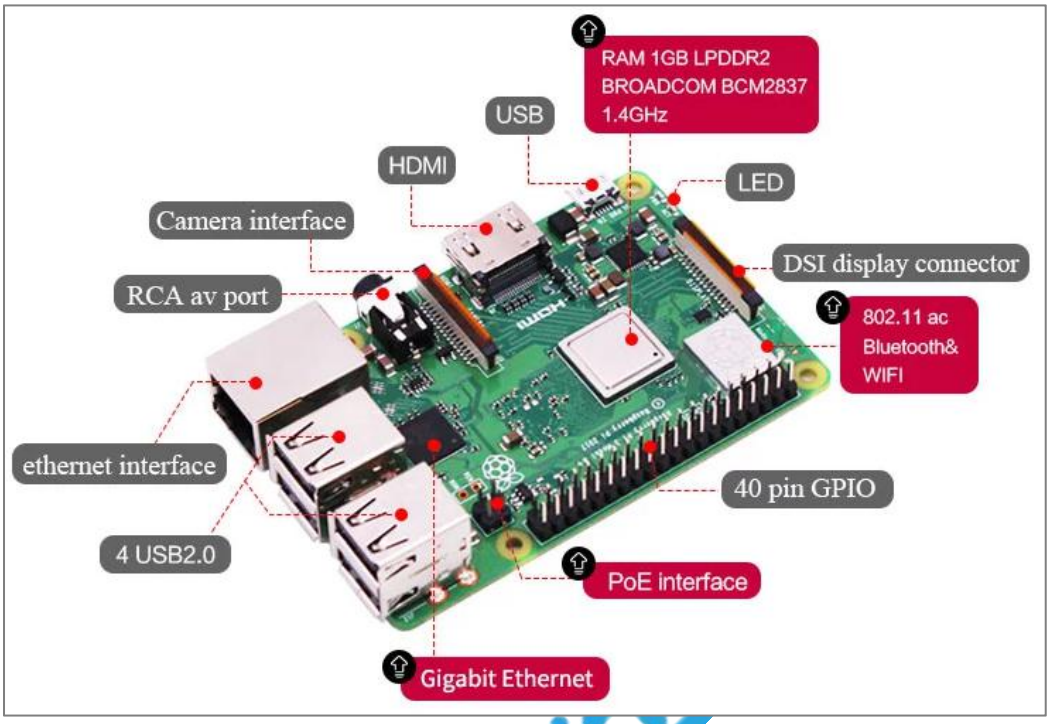
## 5.Raspberry Pi platform ------- KeyScanStart

### 1) Preparation




1-1 Raspberry Pi board

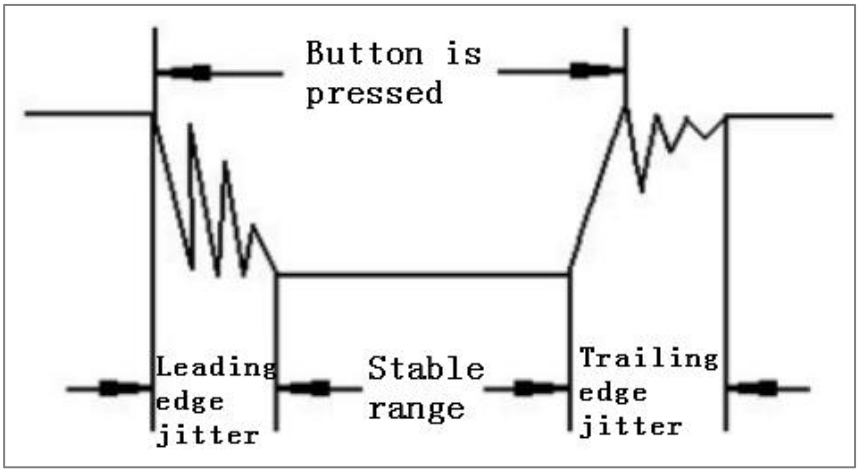
1-2 Key

### 2) Purpose of Experimental

After running the KeyScanStart executable in the Raspberry Pi system. You need to press the KEY to start the car, the car will advance 1 s ,back 1 s,turn left 2 s,turn right 2 s, turn left in place 3 s, turn right in place 3 s, stop 0.5s.

### 3)Principle of experimental

Generally, our button switches are mechanical elastic switches. When the mechanical contacts are opened and closed, due to the elastic action of the mechanical contactor, switchs will not be able to be connected immediately when closed and it will not be disconnected at once.



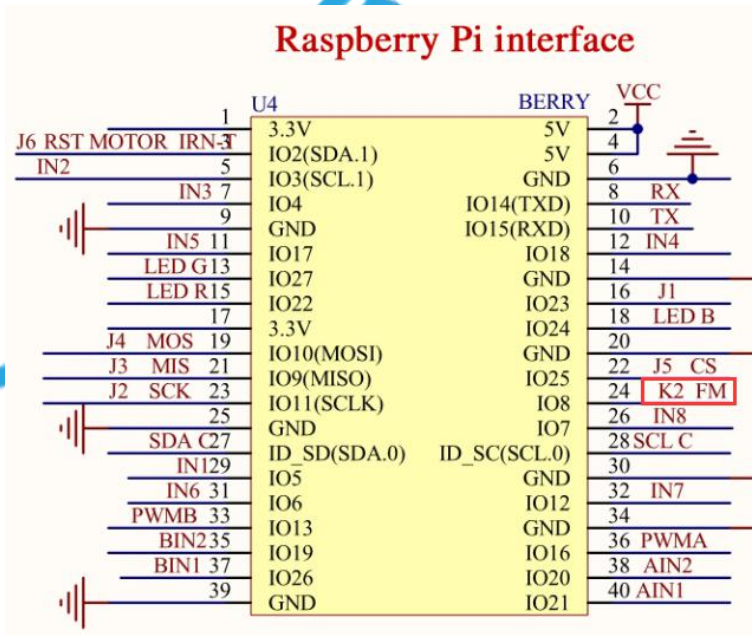

3-1 Button jitter state diagram

The jitter time is usually within 10ms. The button must be eliminated jitter to ensure that the program only responds once after the button is closed once.

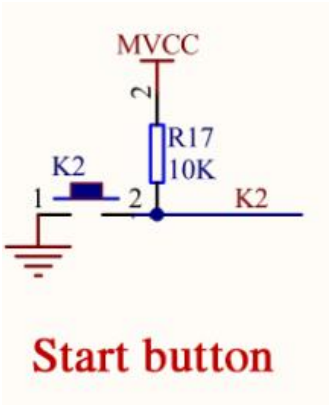
In this experiment, we took the software delay eliminated jitter. After detecting that the button is closed, delay codes is executed to generate a delay of 5ms to 10ms, and the state of the button is detected again after the leading edge jitter disappears. When it is detected that the button is released, it also needs to delay 5ms~10ms.

**4)Experimental Steps**

4-1 About the schematic




4-1 Raspberry Pi interface circuit diagram




4-2 key circuit diagram

| wiringPi | BCM | Funtion | Physical pin | | Funtion | BCM | wiringPi |
|---|---|---|---|---|---|---|---|
| | | 3.3V | 1 | 2 | 5V | | |
| 8 | 2 | SDA.1 | 3 | 4 | 5V | | |
| 9 | 3 | SCL.1 | 5 | 6 | GND | | |
| 7 | 4 | GPIO.7 | 7 | 8 | TXD | 14 | 15 |
| | | GND | 9 | 10 | RXD | 15 | 16 |
| 0 | 17 | GPIO.0 | 11 | 12 | GPIO.1 | 18 | 1 |
| 2 | 27 | GPIO.2 | 13 | 14 | GND | | |
| 3 | 22 | GPIO.3 | 15 | 16 | GPIO.4 | 23 | 4 |
| | | 3.3V | 17 | 18 | GPIO.5 | 24 | 5 |
| 12 | 10 | MOSI | 19 | 20 | GND | | |
| 13 | 9 | MISO | 21 | 22 | GPIO.6 | 25 | 6 |
| 14 | 11 | SCLK | 23 | 24 | CE0 | 8 | 10 |
| | | GND | 25 | 26 | CE1 | 7 | 11 |
| 30 | 0 | SDA.0 | 27 | 28 | SCL.0 | 1 | 31 |
| 21 | 5 | GPIO.21 | 29 | 30 | GND | | |
| 22 | 6 | GPIO.22 | 31 | 32 | GPIO.26 | 12 | 26 |
| 23 | 13 | GPIO.23 | 33 | 34 | GND | | |
| 24 | 19 | GPIO.24 | 35 | 36 | GPIO.27 | 16 | 27 |
| 25 | 26 | GPIO.25 | 37 | 38 | GPIO.28 | 20 | 28 |
| | | GND | 39 | 40 | GPIO.29 | 21 | 29 |

4-4 Raspberry Pi 40 pins comparison table

4-2 According to the circuit schematic:

Key-----24(Physical pin)----- 10(wiringPi)

(Note: We use the wiringPi library to write code.)

4-3 About the code

Please view .py and.c file

**A. For .c code**

(1) We need to compile this file in the Raspberry Pi system. (Note: we need to add -lwiringPi to the library file.)

We need to input:gcc KeyScanStart.c -o KeyScanStart -lwiringPi

(2) We need to run the compiled executable file in the Raspberry Pi system.We need to input: ./KeyScanStart

```
pi@yahboom4wd:~/SmartCar $ gcc KeyScanStart.c -o KeyScanStart -lwiringPi
pi@yahboom4wd:~/SmartCar $ ./KeyScanStart
```

(3)We can input: ctrl+c to stop this process, which mean is send a signal to the linux kernel to terminate the current process, but the state of the relevant pin is uncertain at this time, we also need to run a script to initialize all pins.
(Note:The initpin.sh script file is included in the SmartCar directory.)
You need to input: chmod 777 initpin.sh
./initpin.sh

```
pi@yahboom4wd:~/SmartCar $ sudo chmod 777 initpin.sh
pi@yahboom4wd:~/SmartCar $ ./initpin.sh
```

**B. For python code**

(1) We need to input following command to run python code.
python KeyScanStart.py

```
pi@yahboom4wd:~/python $ python KeyScanStart.py
```

(2) We can input: ctrl+c to stop this process, which mean is send a signal to the linux kernel to terminate the current process, but the state of the relevant pin is uncertain at this time, we also need to run a script to initialize all pins.
(3) You need to input: chmod 777 initpin.sh
./initpin.sh

```
pi@yahboom4wd:~/SmartCar $ sudo chmod 777 initpin.sh
pi@yahboom4wd:~/SmartCar $ ./initpin.sh
```

After completing the above steps, the experiment is over.

**5) Experimental phenomenon**

When we run the program, **you need to press the KEY to start the car,** the car will advance 1 s ,back 1 s,turn left 2 s,turn right 2 s, turn left in place 3 s, turn right in place 3 s, stop 0.5s.
KEY as shown below.