As Raspbian system is updated to Stretch, the python-smbus module is also updated a little with the system. But its efficiency seems to become lower. That leads to some problems: the communication between RPi and Shield become abnormal at some time, which lead to ultrasonic data returned by Shield and received by RPi is wrong. Other cases is normal, such as RPi write data to shield. We can repair the problem by two ways below.

1. Because only part of the ultrasonic data received by RPi is abnormal, it is feasible to filter out the erroneous data. We read twice data continuously, compare them, and judge rationality of the data. And then decide whether to use this data. If the data is relatively reasonable, return the data, otherwise the above operation will be circulated until the reasonable data is obtained or the maximum number of cycles reaches.
2. By using an oscilloscope to view the communication waveform between RPi and Shield, we found that when RPi issue a data request, the Shield immediately return to the requested data, but RPi seemed to be unable to respond quickly, and miss the time of data reception, then resulted in a data reception error. This situation does not exist in the previous Jessie system, which is the reason why we say the efficiency seems to be lower after the update. In this case, we modify the firmware, when RPi sends out a data request, make Shield wait a few microseconds to return the data to RPi. Through this solution, the quality of the communication has been greatly guaranteed, and the data receiving error has not occurred after a long time run.

If you prefer ultrasonic rather than camera, and you have the following conditions to upgrade firmware, you can make firmware upgrading to Shield. But if you prefer camera, these changes have nothing to do with you. You can ignore following contents, because the above solution way 1 is fully able to meet your needs.

Firmware upgrading methods:

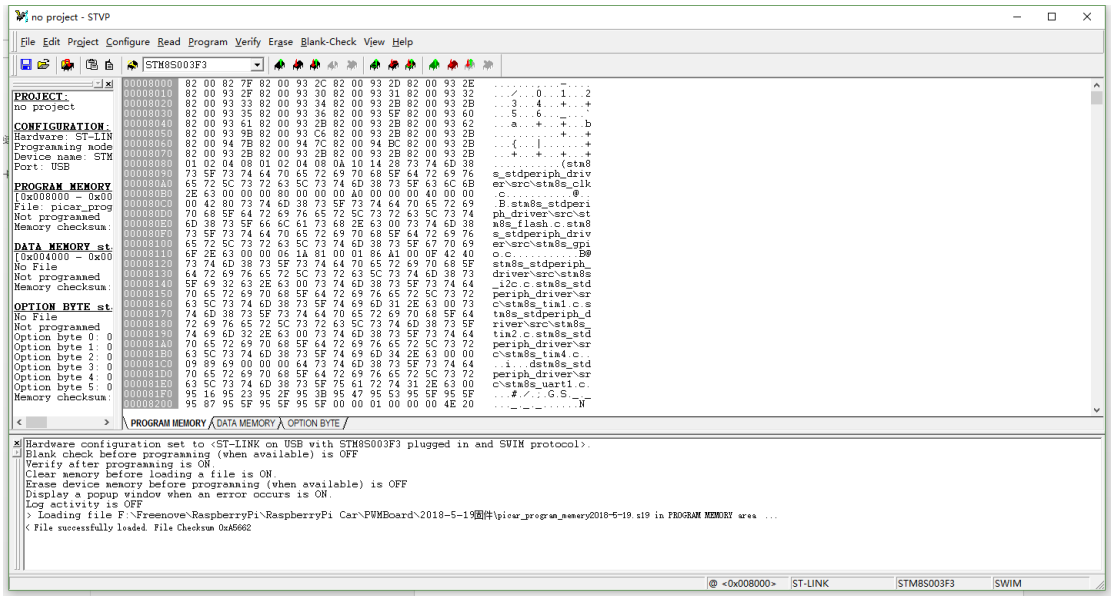
1. Preparation Items: burning software, hardware, firmware

Item	Type	Link
Burning Software	STVP	http://www.st.com/content/st_com/en/products/development-tools/software-development-tools/stm8-software-development-tools/stm8-programmers/stvp-stm8.html
Hardware	ST-Link or other Stm8s program burner	Self-provide
Program Firmware	picar_program_memery.s19	https://github.com/Freenove/Freenove_Three-wheeled_Smart_Car_Kit_for_Raspberry_Pi
Selective Byte Firmware	picar_option_byte.s19	

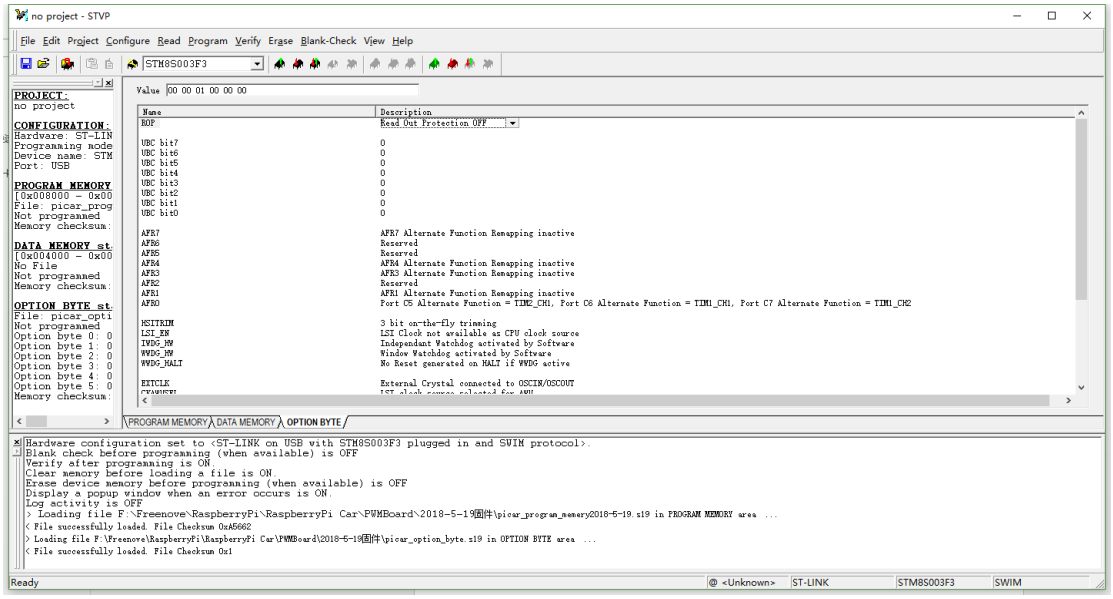
2. Steps

   Open STVP, and select MCU STM8S003F3. On PROGRAM MEMERY tag page, load the program firmware “picar_program_memery.s19”, and on OPTION BYTE label page, load the selected byte firmware ”picar_option_byte.s19”, and click the burn button.

Load program firmware:



Load the option byte firmware:



Click “Program all tabs (on active sectors if any)”, then the burning succeeds.