RPI BASED AUTONOMOUS CAR

FULLSTACK EMBEDDED (2017)

FREDERIC AFADJIGLA

CONTENTS

LIST OF FIGURES

ABSTRACT

The purpose of this document is to give the reader an overview about the RpiAutonomousCar's hardware. This will be the learning platform for the FSE 2017 Workshop in Lomé and Accra. The principal hardware components of the board and interfaces will be briefly described.

1 INTRODUCTION

Goal of FSE 2017 will be to build an autonomous car with obstacle avoidance capability with the students. The robot should be able to drive 2 brushed DC motors, an ultrasonic sensor for distance measurement and a servo motor. To be able to fullfill these requirements with a Raspberry Pi, an additional board will be designed.

2 RPIAUTONOMOUSCAR'S MECHANIC

As shown in Figure 2 and 1, the mechanical structure is simple and easy to build. The kit includes:

- 1 x Car Chassis
- 2 x Gear Motor(1:48)
- 2 x Car Tire
- 2 x Speed Encoder
- 2 x Fastener
- 1 x Universal Wheel
- 1 x Battery Box
- All Necessary Screw And Nut

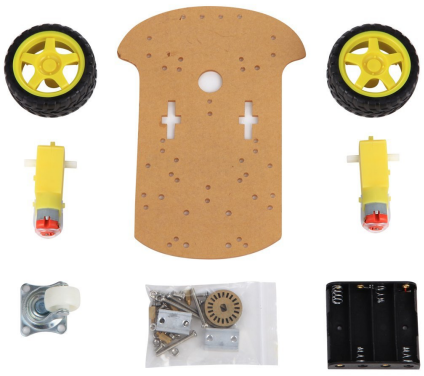

Figure 1: Robot's components

Figure 2: Assembled robot

3 RPIAUTONOMOUSCAR'S ELECTRONIC

The Board's electronic is kept simple to ease training. It is designed so that students can use the same board to design other embedded system examples.

3.1 *Led*

The RpiAutonomousCar's board has one led connected to GPIO21.

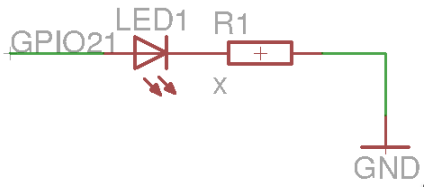

Figure 3: Led

3.2 *Analog Inputs for Raspberry Pi Using the MCP3004*

The MCP3004 is a low cost 4-channel 10-bit analog to digital converter. The MCP3004 connects to the Raspberry Pi using a serial peripheral interface (SPI) serial connection. You can use either the hardware SPI bus (remember to connect GPIO07 and GPIO25 together via a jumper cable if you have the first version of the board), or any four GPIO pins and software SPI to communicate to the MCP3004. Software SPI is a little more flexible since it can work with any pins on the Pi, whereas hardware SPI is slightly faster but less flexible because it only works with specific pins. The board gives you the possibility to try both methods since it is also wired to the Raspberry Pi hardware SPI.

SPI Pins	Raspberry Pi pin
SCLK (Serial Clock)	GPIO11
MOSI (Master Out Slave In)	GPIO10
MISO (Master In Slave Out)	GPIO09
CS (Chip Select)	GPIO25

Table 1: MCP3004 SPI connection

Voltage that are allowed to be connected to the MCP3004 have to be selected so that V_{out} muss be less than 3.3V(V_{ref}). V_{out} is the voltage that can be measured after the voltage dividers. All channels except channel

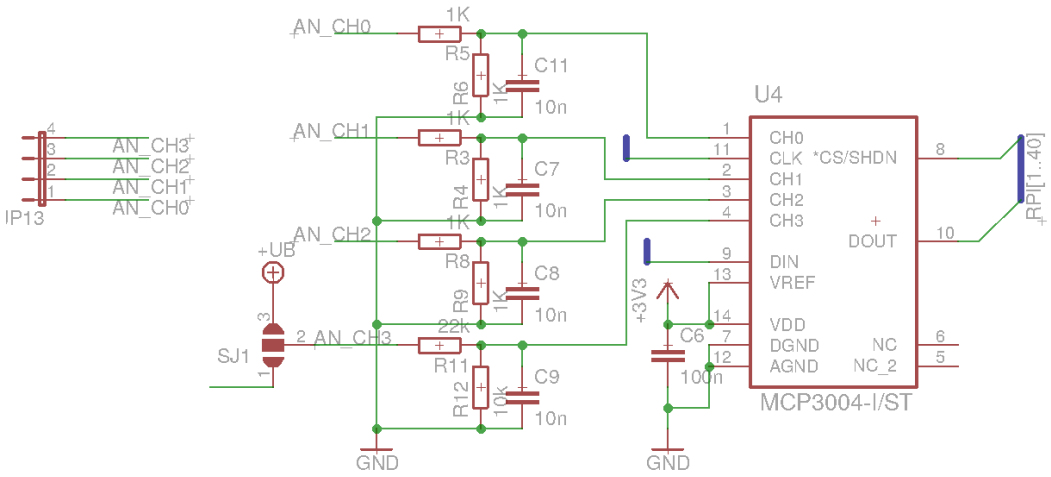

Figure 4: MPC3004 schematic

3 are designed to have a measuring range of 0-5V DC. Channel 3 can measure up to 10V. $V_{out[0-2]}$ for channels 0,1,2 can be calculated as:

$$V_{out[0-2]} = \frac{R_6}{R_5 + R_6} * V_{in} = 0.5 * V_{in}$$

In case of channel 3:

$$V_{out[3]} = 0.316 * V_{in}$$

Please note that the analog channel 3 is connected per default via a jumper (SJ1) to the robot's supply voltage. Remove the jumper to be able to connect another voltage source to channel 3.

3.3 *8-channel Bi-directional Logic Level Converter - TXB0108*

Precautions have to be taken when connecting 5V devices to the raspberry pi since the Pi is not 5V tolerant. There are many simple ways to handle this issue like voltage dividers.Handling bidirectional signals or high speed transfers can be tough. That's where this lovely chip, the TXB0108 bi-directional level converter comes in! This chip performs bidirectional level shifting from pretty much any voltage to any voltage and will auto-detect the direction.

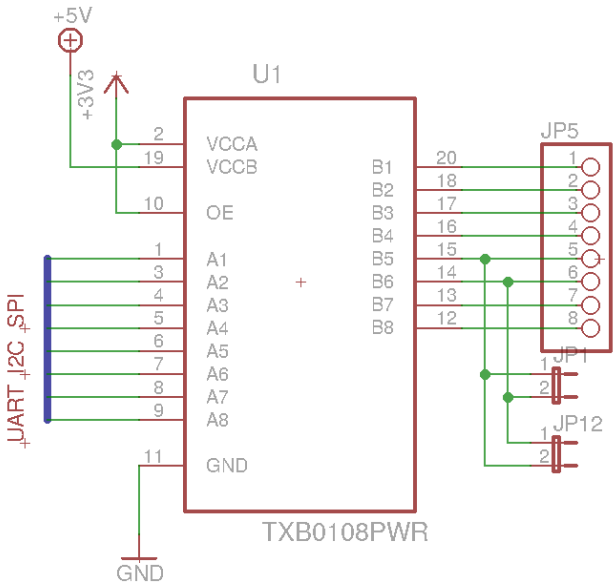

Figure 5: TXB0108PWR schematic

Pins	RPi pin
1	GPIO10
2	GPIO09
3	GPIO11
4	GPIO08
5	GPIO02
6	GPIO03
7	GPIO14
8	GPIO15

Table 2: TX0108PWR connection

3.4 *DC Motor driver - TB6612FNG*

The TB6612FNG motor driver can control up to two DC motors at a constant current of 1.2A (3.2A peak). Two input signals (IN1 and IN2) can be used to control the motor in one of four function modes - clockwise (CW), counter-clockwise (CCW), short-brake, and stop. The two motor outputs (A and B) can be separately controlled, the speed of each motor is controlled via a PWM input signal with a frequency up to 100kHz. The STBY pin should be pulled high to take the motor out of standby mode.

Logic supply voltage (VCC) can be in the range of 2.7-5.5VDC, while the motor supply (VM) is limited to a maximum voltage of 15VDC. The output current is rated up to 1.2A per channel (or up to 3.2A for a short,

single pulse).

Features:

- Power supply voltage: VM=15V max, VCC=2.7-5.5V
- Output current: Iout=1.2A(average) / 3.2A (peak)
- Standby control to save power
- CW/CCW/short brake/stop motor control modes
- Built-in thermal shutdown circuit and low voltage detecting circuit
- All pins of the TB6612FNG broken out to 0.1" spaced pins
- Filtering capacitors on both supply lines

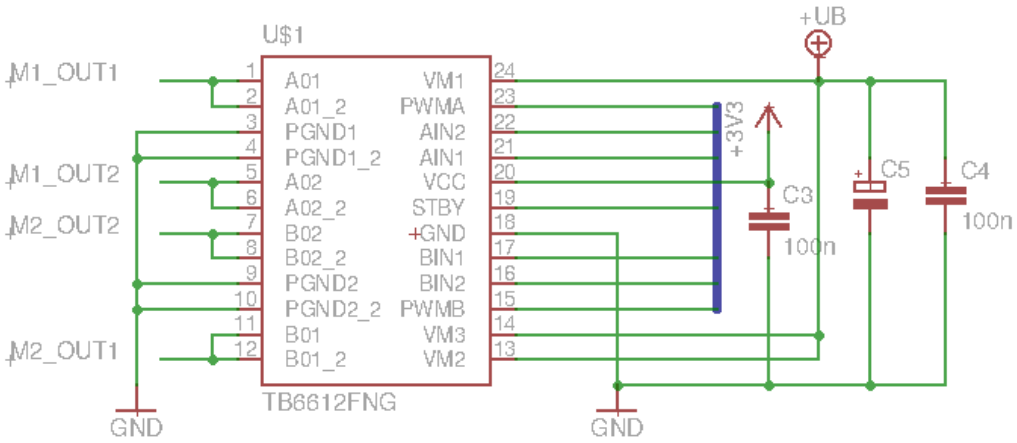

Figure 6: TB6612FNG schematic

Let's discuss the pinout for the TB6612FNG:

Pin Label	Function	Notes
VM	Motor Voltage	This is where you provide power for the motors (2.2V to 13.5V)
VCC	Logic Voltage	This is the voltage to power the chip and talk to the microcontroller (2.7V to 5.5V)
GND	Ground	Common Ground for both motor voltage and logic voltage (all GND pins are connected)
STBY	Standby	Allows the H-bridges to work when high (has a pulldown resistor so it must actively pulled high)
AIN1/BIN1	Input 1 for channels A/B	One of the two inputs that determines the direction
AIN2/BIN2	Input 2 for channels A/B	One of the two inputs that determines the direction
PWMA/PWMB	PWM input for channels A/B	PWM input that controls the speed
A01/B01	Output 1 for channels A/B	One of the two outputs to connect the motor
A02/B02	Output 2 for channels A/B	One of the two outputs to connect the motor

Table 3: TB6612FNG connection

When the outputs are set to High/Low your motor will run. When they are set to Low/High the motor will run in the opposite direction. In both cases, the speed is controlled by the PWM input.

In1	In2	PWM	Out1	Out2	Mode
H	H	H/L	L	L	Short brake
L	H	H	L	H	CCW
L	H	L	L	L	Short brake
H	L	H	H	L	CW
H	L	L	L	L	Short brake
L	L	H	OFF	OFF	Stop

Table 4: TB6612FNG Logic

4 WIRING THE RPICAR

4.1 *Connecting Robot's components*

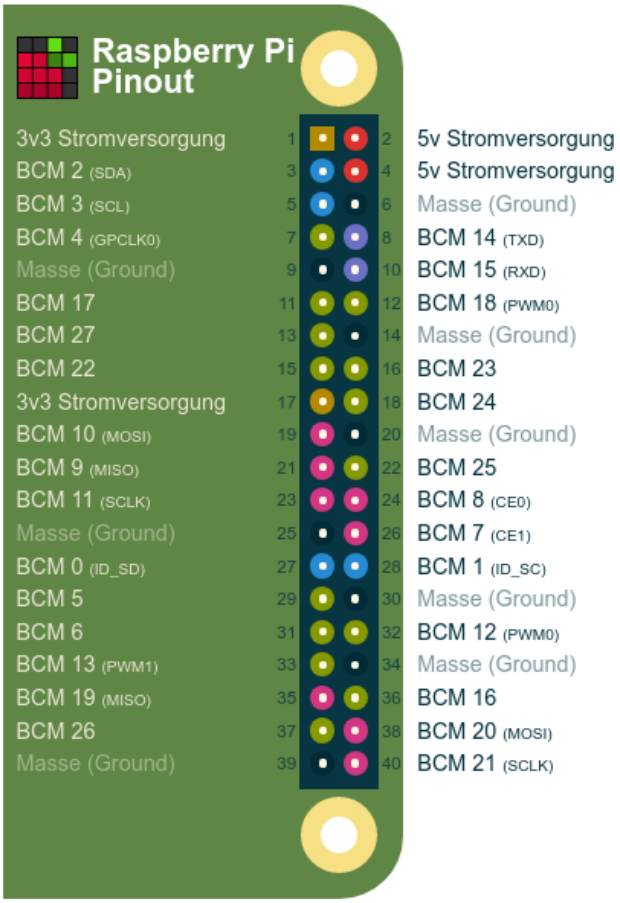

Figure 7: Raspberry Pi Pinout [1]

- Connecting M1 and M2
  M1 = Motor left and M2 = Motor right

1 https://de.pinout.xyz/pinout/

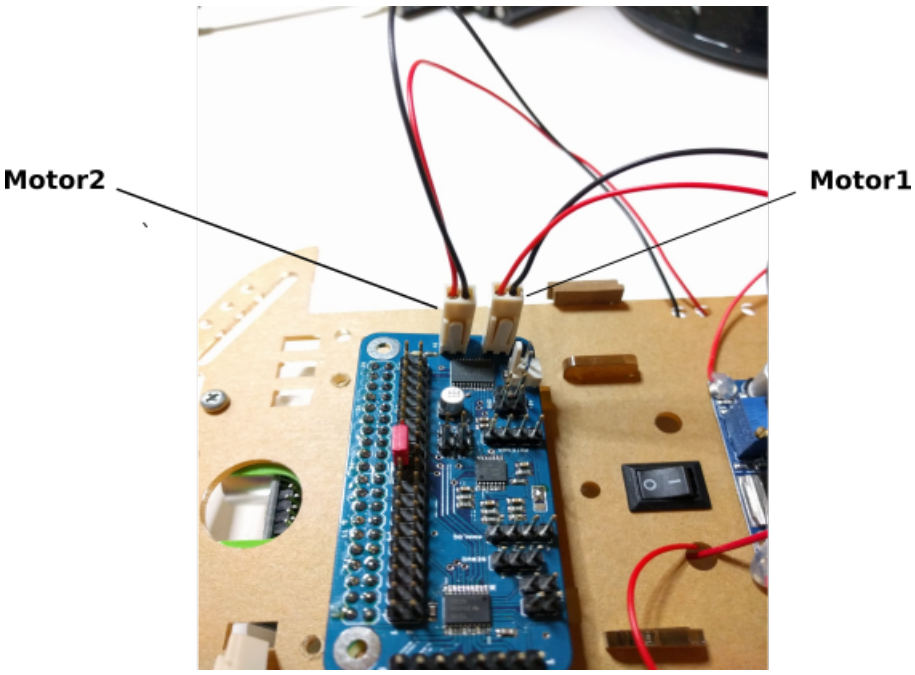

Figure 8: Connecting Motor1 and Motor2

- Connecting Motors PWR and Rpi PWR

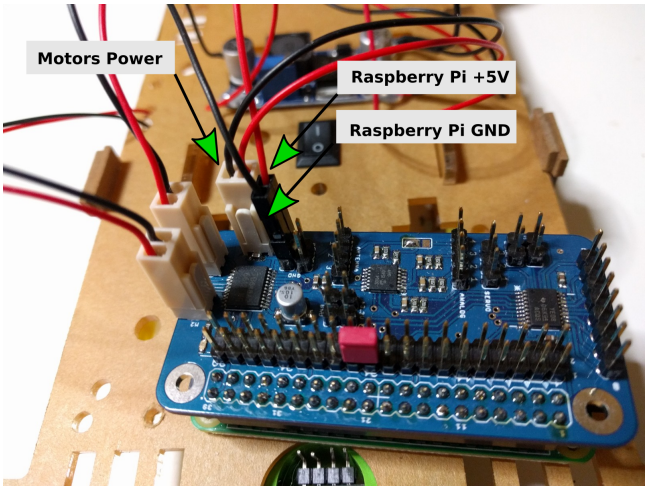

Figure 9: Connecting Motors PWR and Rpi PWR

- Connecting Motor Driver Pins
  Note: Wiring the motor driver is only needed vor V01 of the board. Later releases of the board will incorporate those changes.

  Please connect those pins below together:

  M1 PwmPinAnnex = 5 ..... m1pwmpin = 18

M2 PwmPinAnnex = 26 ..... m2pwmpin = 13

- Connecting the ultrasonic sensor

Function	Rpi Pin
VCC	+5V
TRIG	15 (Level converted)
ECHO	14 (Level converted)
GND	GND

- Connecting the IR sensor for line following
  IR sensor can be connected to any available GPIO when used in digital mode. If you want more accuracy analog mode can also be used. Therefore the MCP3004 chip has to be used. Connect the sensor output to the available pins of the MCP3004