

モーター駆動回路
モータードライバで駆動
モータードライバの仕様:
駆動回路
駆動方法
プログラムのコード
実行結果
参考文献
ソフトについて

モーター駆動回路

モータードライバで駆動

この文章のリンク : [gitpage](#)

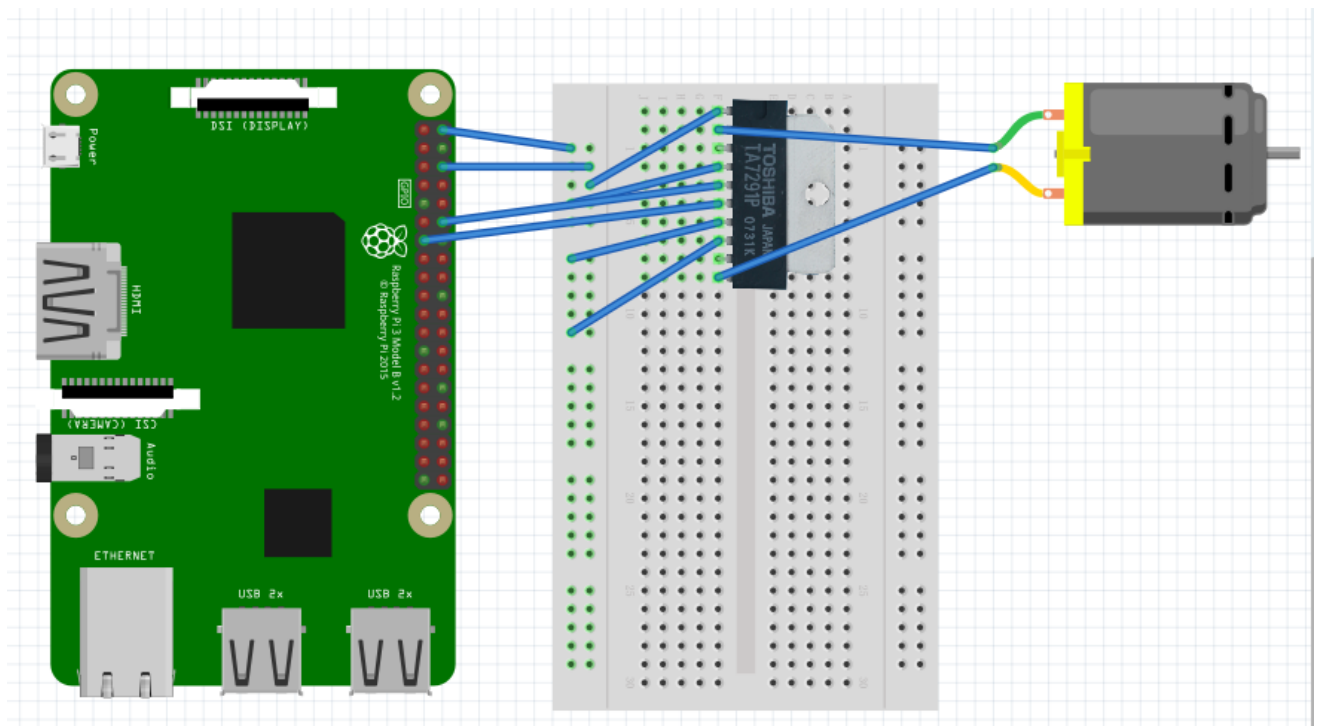
モータードライバの仕様:

型番: TA7291P

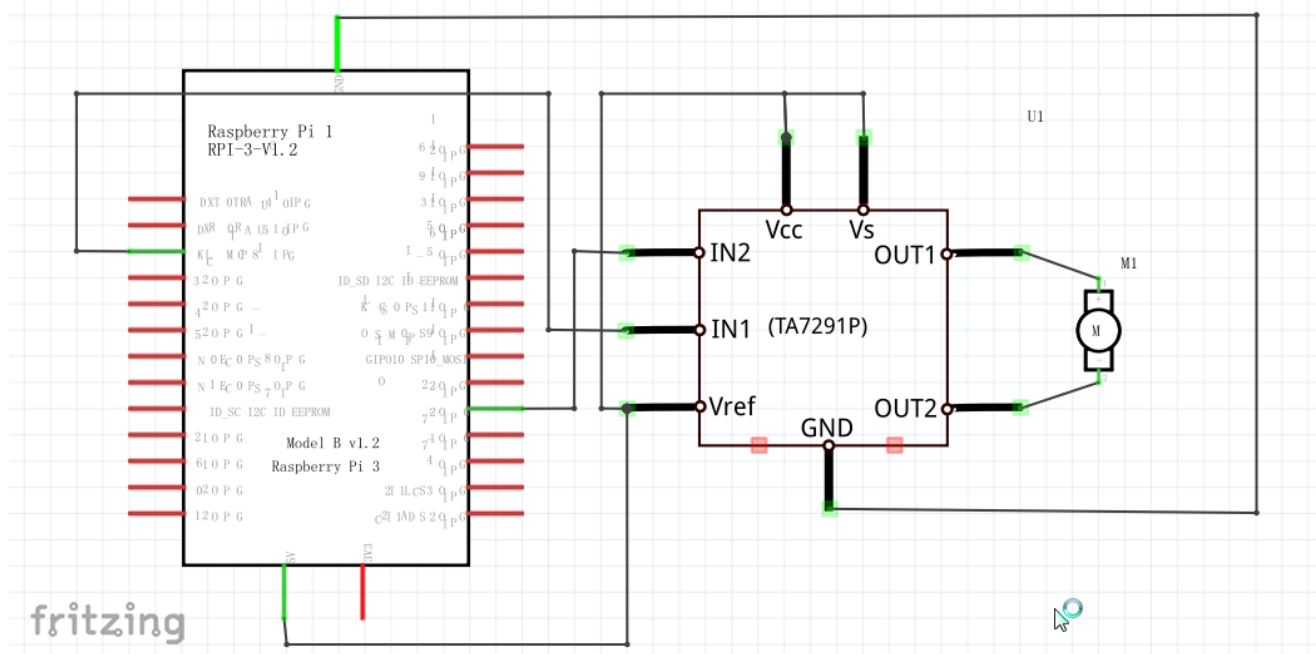
DataSheet: <https://toshiba.semicon-storage.com/info/docget.jsp?did=16128&prodName=TA7291P>

駆動回路

1. BreadBorad:



2. 回路図:



駆動方法

●モータの回転方向はマイコンなどから出力する 0～5V のパルス信号で制御する

表5は、TA7291P の制御信号入力と出力の関係です。入力側の H レベルは 3.5～5V, L レベルは 0V とします。5V 動作のマイコンなら、信号をそのまま TA7291P に与えることができます。IN1 と IN2 が共に H レベルの場合、OUT1 と OUT2 どちらも L レベルとなり、モータにブレーキをかけることができます。

表 5 TA7291P の制御信号入力と出力の関係

入力		出力		動作
IN1	IN2	OUT1	OUT2	
L	L	Z	Z	停止(フリー)
H	L	H	L	正転(または逆転)
L	H	L	H	逆転(または正転)
H	H	L	L	ブレーキ

(注)Zはハイ・インピーダンス・出力側電源の+にもーにも繋がっていない状態

プログラムのコード

```
#!/usr/bin/env python
# -*- coding: utf-8 -

# import libraries
import os
import sys
import tty, termios
import RPi.GPIO
import time
```

```

# setup input pin for motor
In1_Motor=13    #GPIOXX
In2_Motor=19    #GPIOXX

# set up BCM GPIO numbering
RPi.GPIO.setmode(RPi.GPIO.BCM)

# set pins for Motor Diver Input Pin to output mode
RPi.GPIO.setup(In1_Motor, RPi.GPIO.OUT)
RPi.GPIO.setup(In2_Motor, RPi.GPIO.OUT)

## GPIOs Initial
RPi.GPIO.output(In1_Motor, False)
RPi.GPIO.output(In2_Motor, False)

try:
    print "Reading form keybord:"
    print "A=正方向 W=ブレーキ S=ストップ D=逆方向 Q=Quit"
    time.sleep(0.1)
# Main loop start here
    while True:
#get events of the controller
        fd=sys.stdin.fileno()
        old_settings=termios.tcgetattr(fd)
        try:
            tty.setraw(fd)
            ch=sys.stdin.read(1)
        finally:
            termios.tcsetattr(fd, termios.TCSADRAIN, old_settings)
            #print 'error'
        if ch=='a':
            print '-----'
            print '【move forward】'
            print '-----'
            RPi.GPIO.output(In1_Motor, True)
            RPi.GPIO.output(In2_Motor, False)
            print (RPi.GPIO.input(In1_Motor))
            print (RPi.GPIO.input(In2_Motor))
        elif ch=='w':
            print '-----'
            print '【Break】'
            print '-----'
            RPi.GPIO.output(In1_Motor, True)
            RPi.GPIO.output(In2_Motor, True)
        elif ch=='s':
            print '-----'
            print '【Stop】'
            print '-----'
            RPi.GPIO.output(In1_Motor, False)
            RPi.GPIO.output(In2_Motor, False)
        elif ch=='d':
            print '-----'

```

```

        print "【move backward】"
        print '-----'
        RPi.GPIO.output(In1_Motor, False)
        RPi.GPIO.output(In2_Motor, True)
    elif ch=='q':
        print '-----'
        print "【shutdown!】"
        print '-----'
        break
    elif ord(ch)==0x3:
        #这个是ctrl c
        print "shutdown"
        break
    print "Reading form keybord:"
    print "A=正方向 W=ブレーキ S=ストップ D=逆方向 Q=Quit"

finally:
    # this block will run no matter how the try block exits
    RPi.GPIO.cleanup() # clean up after yourself

```

実行結果

```

169.254.190.109 x
A=正方向 W=ブレーキ S=ストップ D=逆方向 Q=Quit
【Stop】
Reading form keybord:
A=正方向 W=ブレーキ S=ストップ D=逆方向 Q=Quit
【move backward】
Reading form keybord:
A=正方向 W=ブレーキ S=ストップ D=逆方向 Q=Quit
【move forward】
1
0
Reading form keybord:
A=正方向 W=ブレーキ S=ストップ D=逆方向 Q=Quit
【Break】
Reading form keybord:
A=正方向 W=ブレーキ S=ストップ D=逆方向 Q=Quit

```



参考文献

<https://www.marutsu.co.jp/contents/shop/marutsu/mame/81.html>
<http://www.geocities.jp/zattouka/GarageHouse/soft/Fritzing/FritzingPart.htm>

ソフトについて

1. Fritzing（部品図と回路図を描く）：<http://fritzing.org/home/>
2. TA7291Pのファイル：[ここ](#)でダウンロード
3. 回路のプロジェクトのファイル：[ここ](#)でダウンロード