

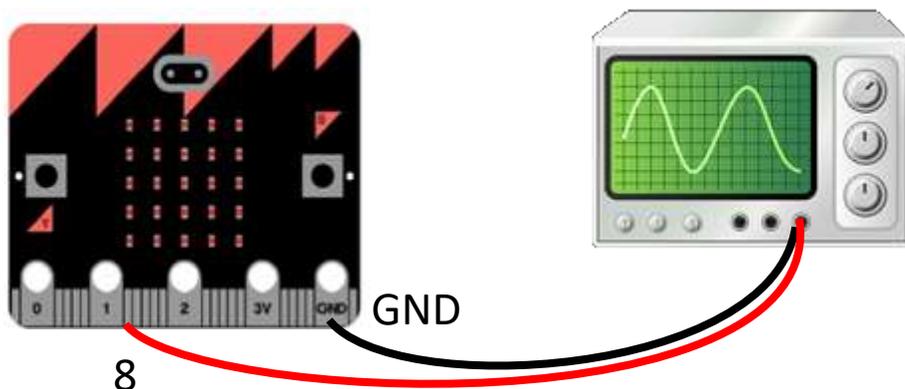
micro:bit pin output test

測定の目的

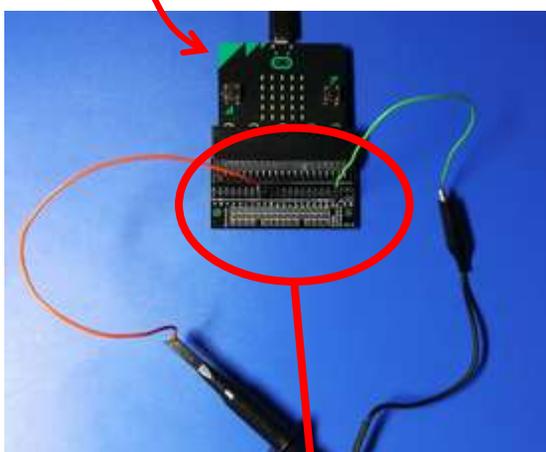
micro:bitにおいてpythonプログラムでP8に対して操作すると、P8ピンに電圧変化が生ずることを確かめる。

【1】測定のための結線

P8のピンの電圧の変化を測定するために、このように結線したい。



micro:bit

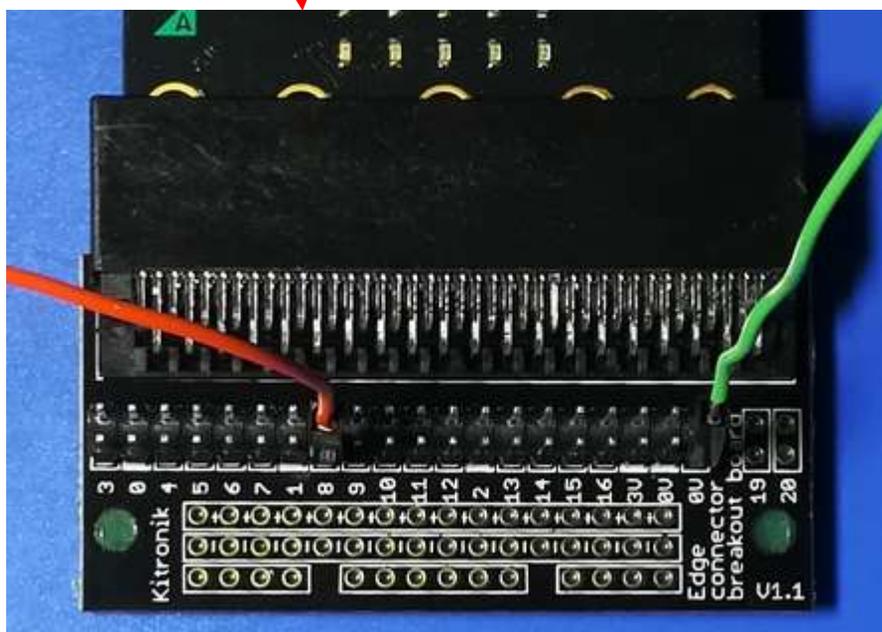


micro:bitにエッジコネクタを取り付け、エッジ端子にオシロスコープを付けて観察できるようにした。

GNDはエッジコネクタの0Vと書かれているピン

測定箇所は8と書かれているピン
このピンはソフトウェア上はpin8となっている。

部分拡大



「8」につながっている

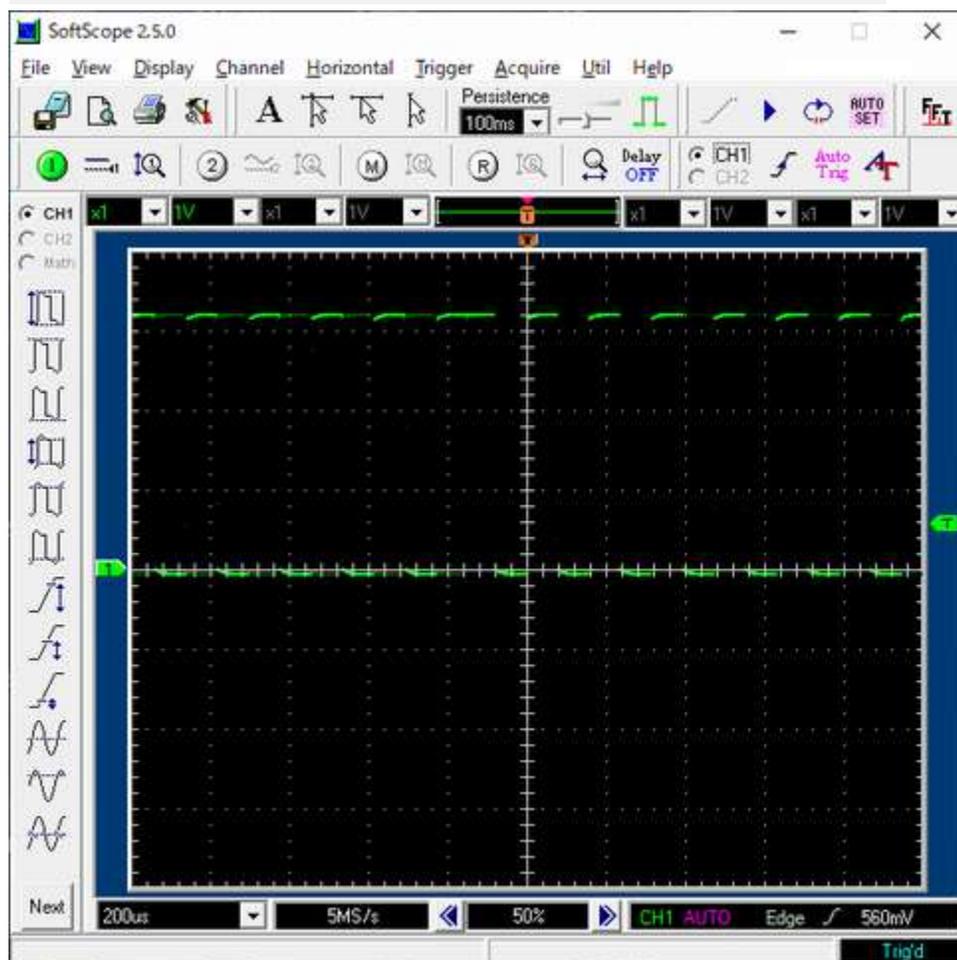
「0V」につながっている

micro:bit pin output test

【2】P8にdigital出力(0, 1を交互に出力)

```
#test python program  
from micro:bit import *
```

```
while True: #次の複数行を無限に繰り返しなさいの意味  
    pin8.write_digital(1) #pin8に1を出力しなさいの意味  
    pin8.write_digital(0) #pin8に0を出力しなさいの意味
```



時間軸(横軸) 200 μ sec/div 点線の間隔が200 μ secということ
電圧(縦軸) 1V/div 点線の間隔が1Vということ
参考 1 μ sec(マイクロ秒)= 10^{-6} 秒 1msec(ミリ秒)= 10^{-3} 秒

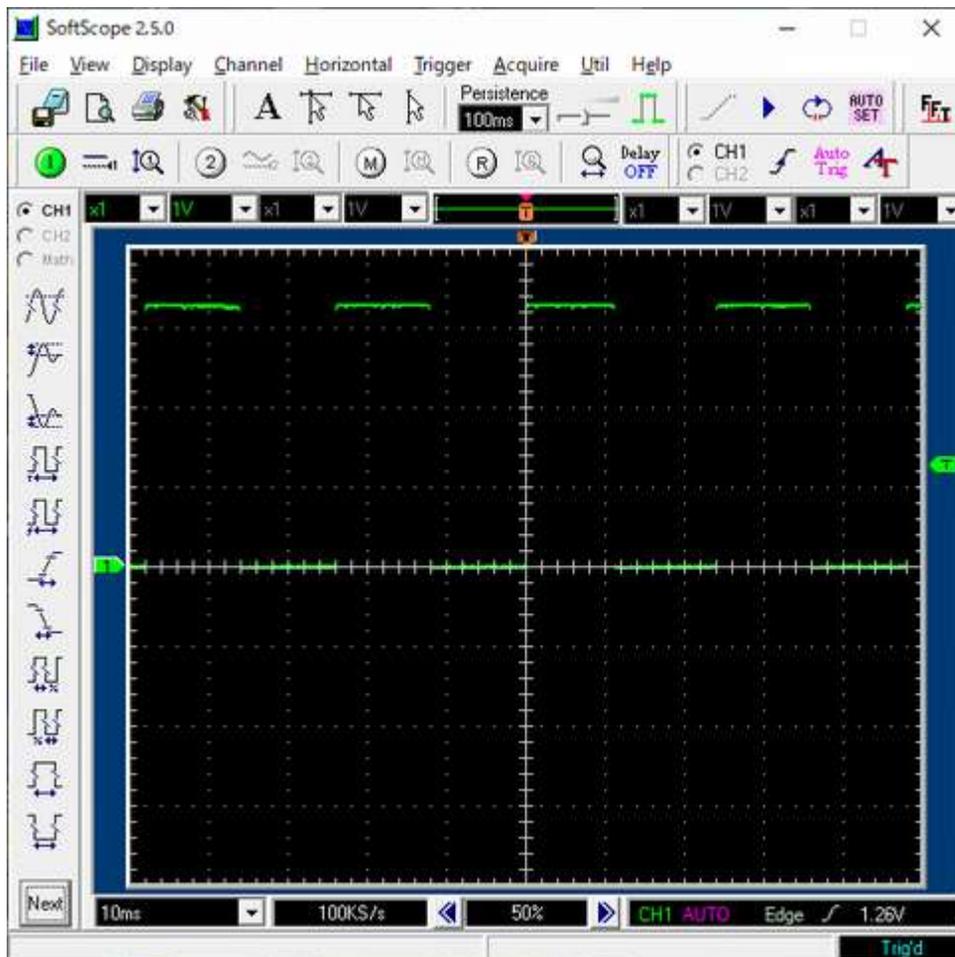
H(3.3V)の期間72 μ sec, L(0V)の期間84 μ secの矩形波(0Vと3.3Vを繰り返している)が観測される。周期は156 μ secということになる。
しかし、頻繁に時間幅は変化する。(この測定でもHの部分が長くなっているところが見えている)
プログラムではH(値1)とL(値0)を交互に出力し続けている。
OSが途中で様々な作業を割り込ませていて、プログラムの動作を中断させる結果と思われる。

micro:bit pin output test

【3】P8にdigital出力(0, 1を交互にゆっくりと出力)

```
#test python program
from micro:bit import *

while True:    #次の複数行を無限に繰り返しなさいの意味
    pin8.write_digital(1)
    sleep(10)  #10msecお休み
    pin8.write_digital(0)
    sleep(10)  #10msecお休み
```



時間軸(横軸) 10msec/div 点線の間隔が10msecということ
電圧(縦軸) 1V/div 点線の間隔が1Vということ
参考 1 μ sec(マイクロ秒)=10⁻⁶秒 1msec(ミリ秒)=10⁻³秒

Hの期間12msec, Lの期間12msecの矩形波が観測される。

周期は24mということになる。

プログラムではH(値1)を出力して10msecそのまま放置し, L(値0)を出力して10msec放置する, という動作を繰り返しているが, 実際にはもう少し舞台裏で作業しているため12msecの動作になっている。

測定していると, 時間幅は頻繁に1msecほど変化する。

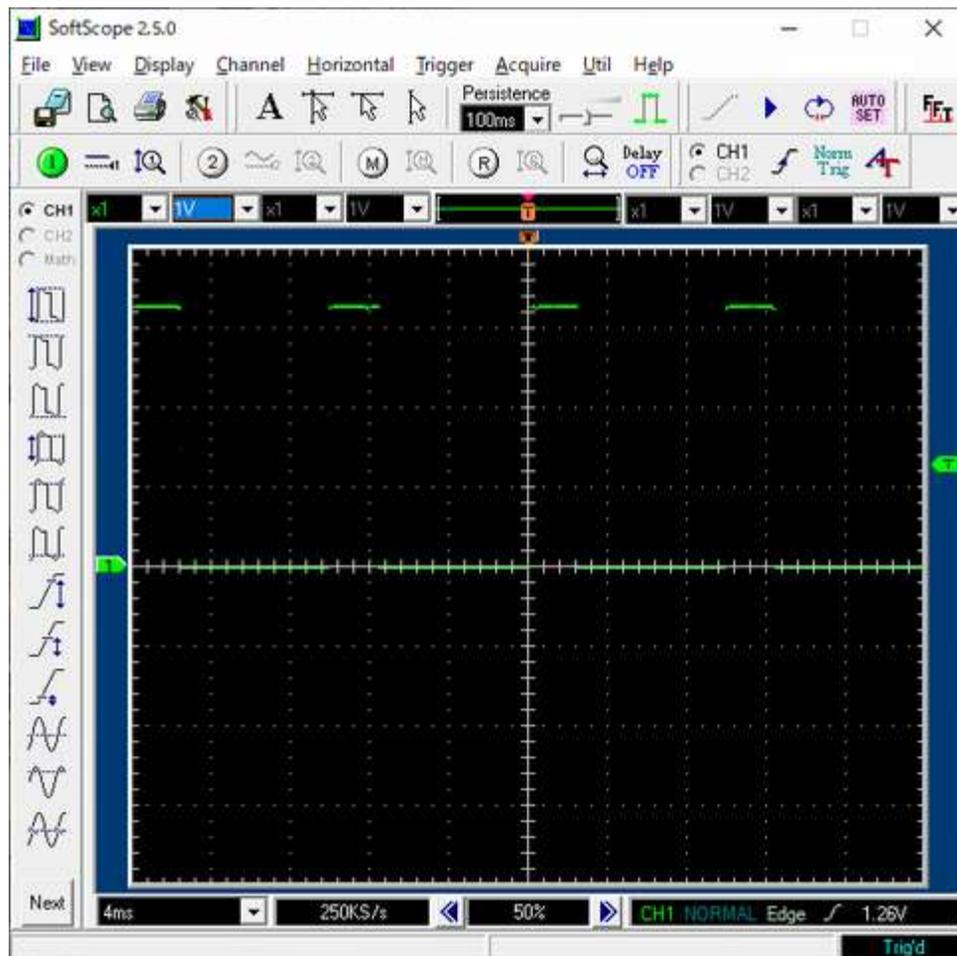
OSが途中で様々な作業を割り込ませている結果と思われる。

micro:bit pin output test

【4】P8にアナログ出力 その1

```
#test python program
from micro:bit import *

#pin8 PWMrperiodを10msecに設定
pin8.set_analog_period(10)
pin8.write_analog(256) #パルス幅256/1024を指示
#このプログラムはここで終了してしまうが、パルスは出力され続ける
```



時間軸(横軸) 4msec/div 点線の間隔が4msecということ
電圧(縦軸) 1V/div 点線の間隔が1Vということ
参考 1 μ sec(マイクロ秒)=10⁻⁶秒 1msec(ミリ秒)=10⁻³秒

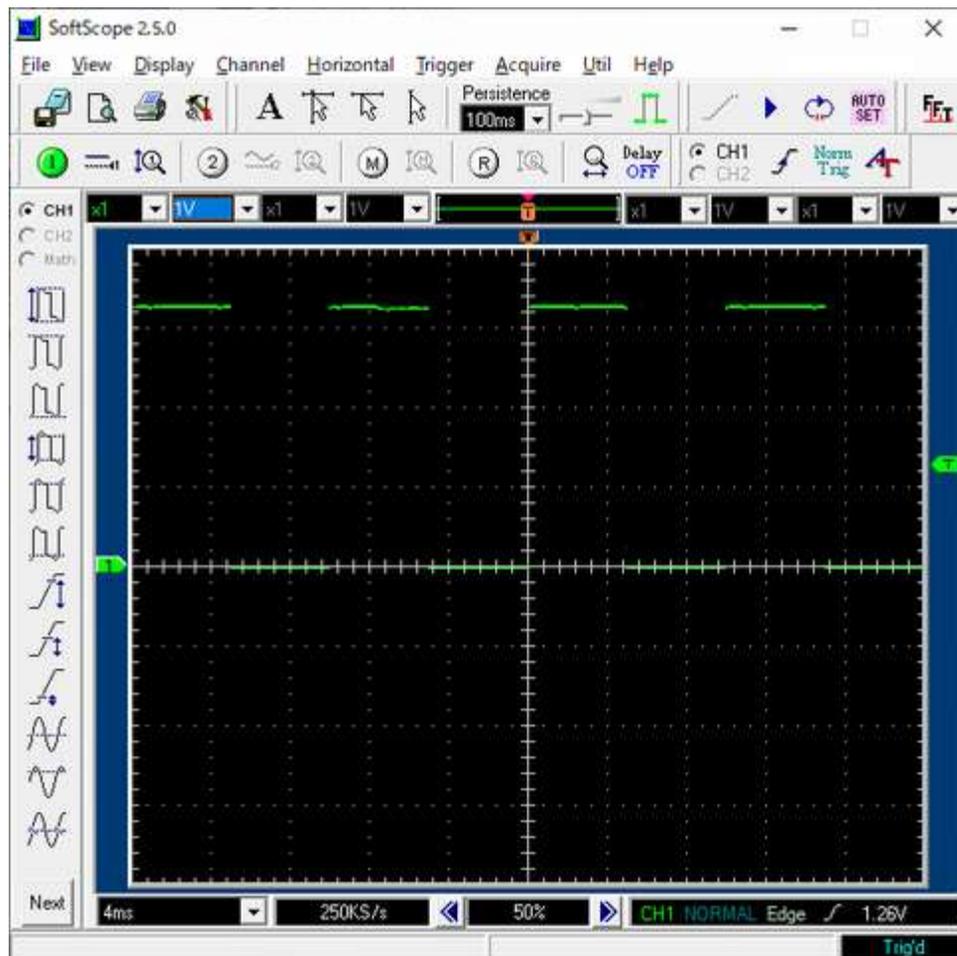
Hの期間2.5msec, Lの期間7.5msecの矩形波が観測される。
周期は10mということになる。
プログラムではanalog値を出力せよということだが、25%がHになる矩形波が出力されている。
このパルスは優先的に出力されているため、パルス幅が変化することはほとんどない。

micro:bit pin output test

【5】P8にアナログ出力 その2

```
#test python program
from micro:bit import *

#pin8 PWMrperiodを10msecに設定
pin8.set_analog_period(10)
pin8.write_analog(512) #パルス幅512/1024を指示
#このプログラムはここで終了してしまうが、パルスは出力され続ける
```



時間軸(横軸) 4msec/div 点線の間隔が4msecということ
電圧(縦軸) 1V/div 点線の間隔が1Vということ
参考 1 μ sec(マイクロ秒)=10⁻⁶秒 1msec(ミリ秒)=10⁻³秒

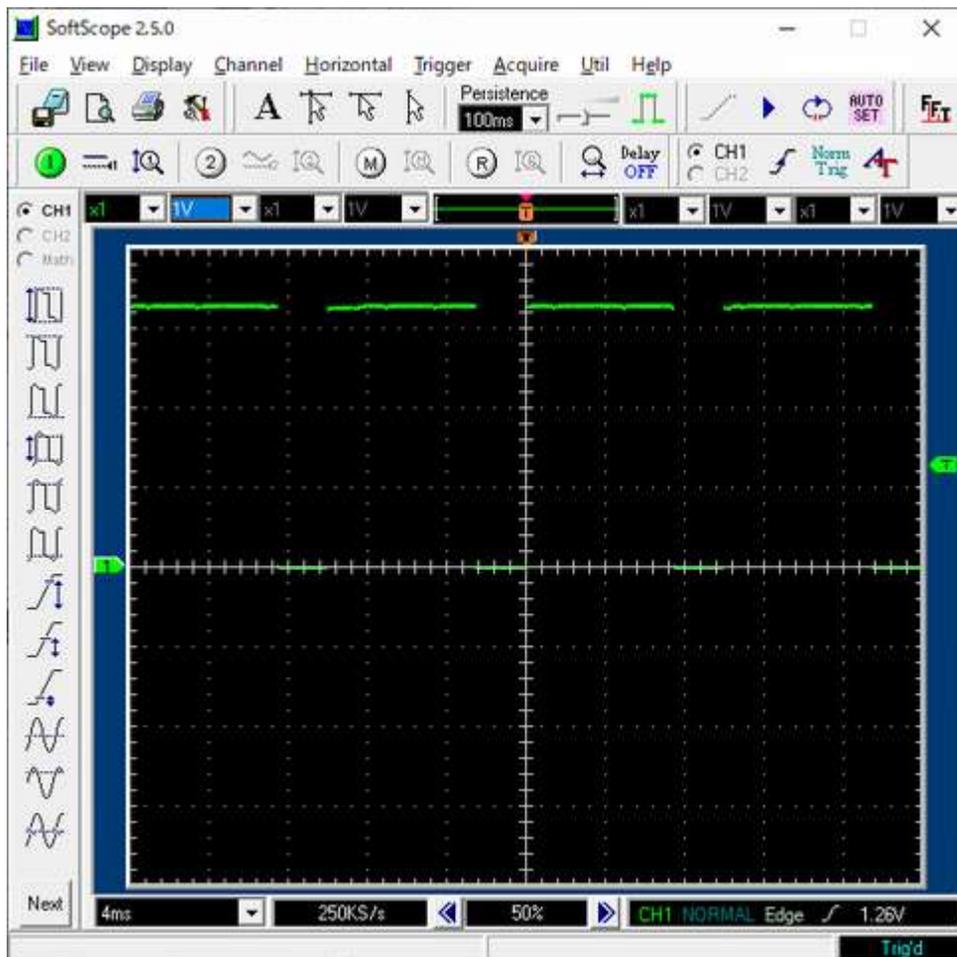
Hの期間5msec, Lの期間5msecの矩形波が観測される。
周期は10mということになる。
プログラムではanalo値を出力せよということだが, 50%がHIになる矩形波が出力されている。
このパルスは優先的に出力されているため, パルス幅が変化することはほとんどない。

micro:bit pin output test

【6】P8にアナログ出力 その3

```
#test python program
from micro:bit import *

#pin8 periodを10msecに設定
pin8.set_analog_period(10)
pin8.write_analog(768) #パルス幅768/1024を指示
#このプログラムはここで終了してしまうが、パルスは出力され続ける
```



時間軸(横軸) 4msec/div 点線の間隔が4msecということ
電圧(縦軸) 1V/div 点線の間隔が1Vということ
参考 1 μ sec(マイクロ秒)=10⁻⁶秒 1msec(ミリ秒)=10⁻³秒

Hの期間7.5msec, Lの期間2.5msecの矩形波が観測される。
周期は10mということになる。
プログラムではanalog値を出力せよということだが、75%がHになる矩形波が出力されている。
このパルスは優先的に出力されているため、パルス幅が変化することはほとんどない。

ところで、write_analogで指定できる値は0から1023までである。そのため100%Hを出し続けることはできない。最大で1023/1024までである。

このようにパルス周期を一定に保ち、Hになる時間の比(デューティ比)を変化させる方式はPWM(Pulse width Modulation)と呼ばれる。