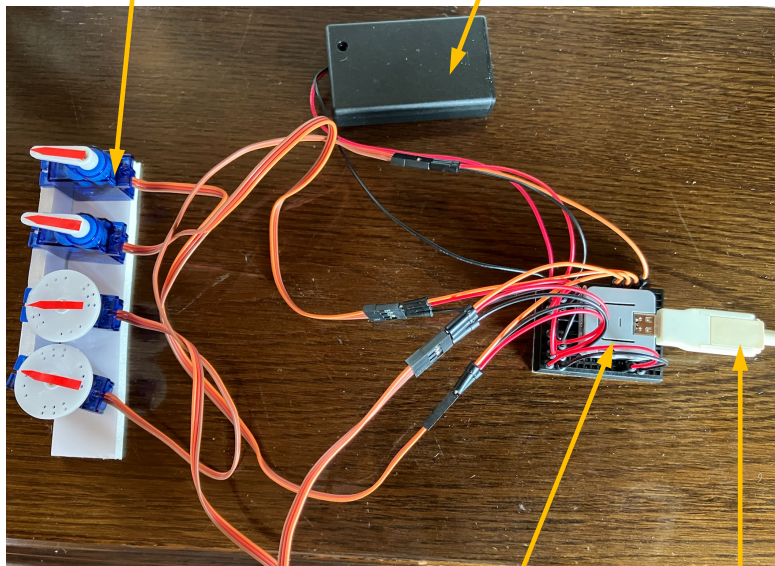


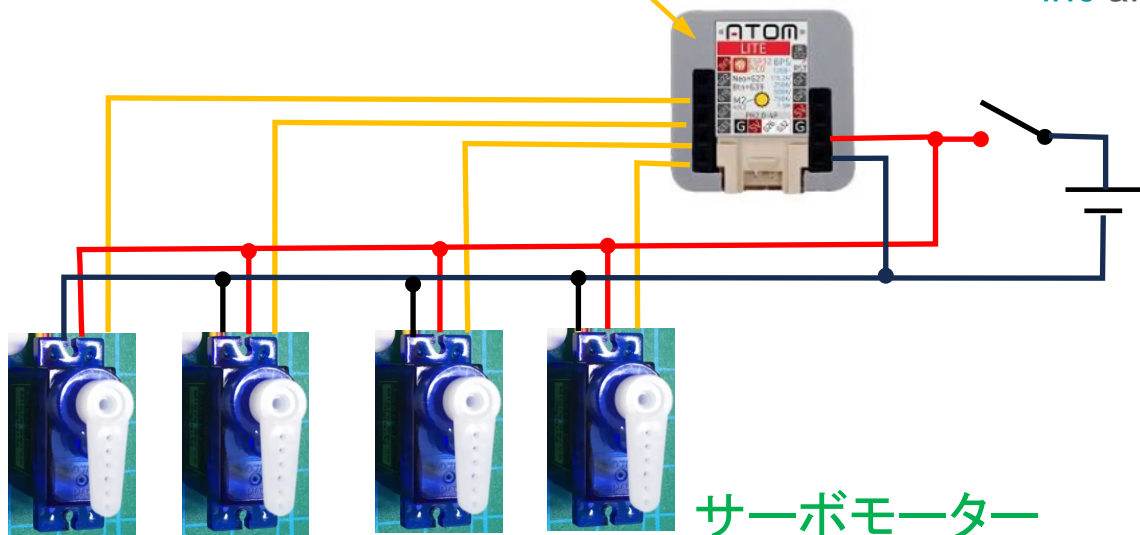
サーボモーター

電池ボックス



M5Atom

USB Type C



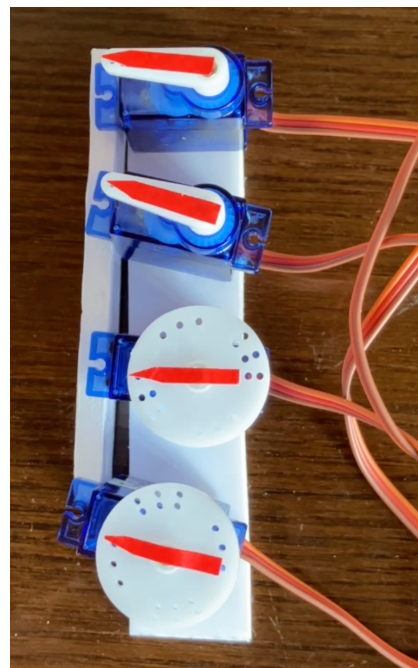
サーボモーター

No.2

No.1

No.3

No.0



90

90

90

90

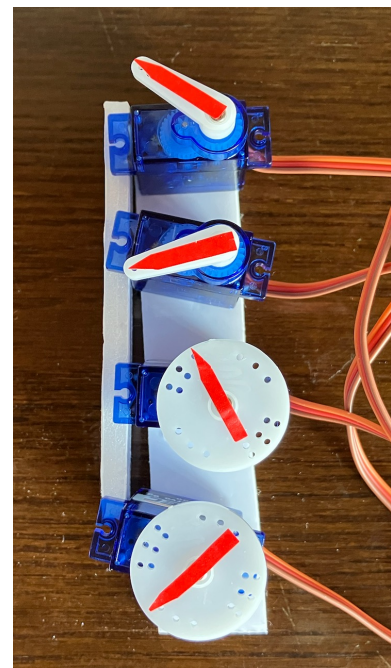


120

60

150

30



`int angZero[] = {90, 90, 90, 90};`

No.0 1 2 3

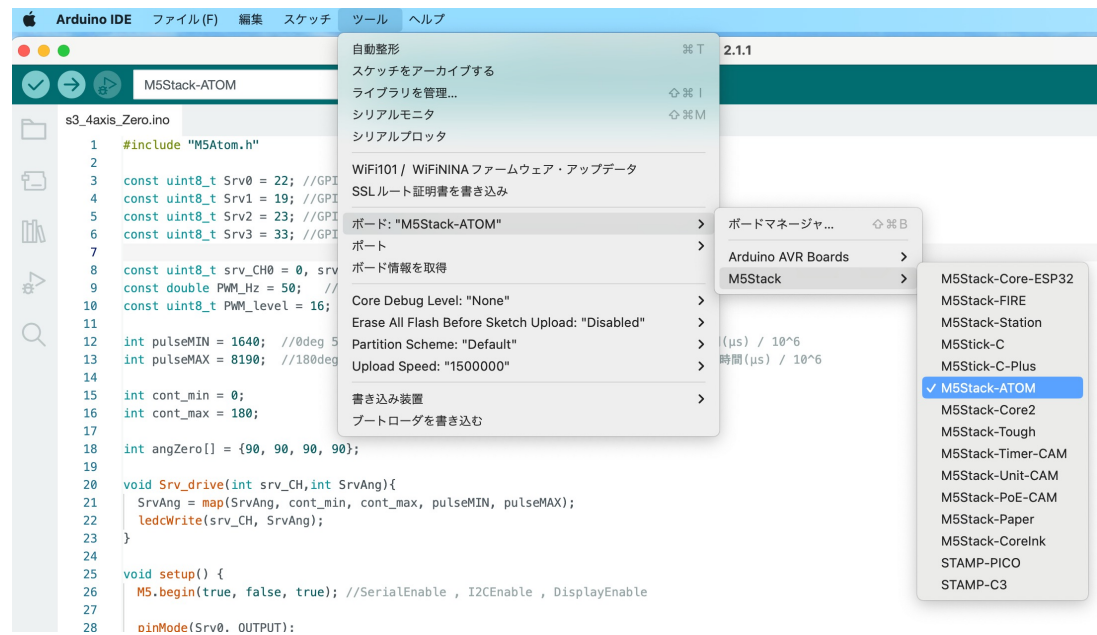


`int angZero[] = {30, 60, 120, 150};`

No.0 1 2 3

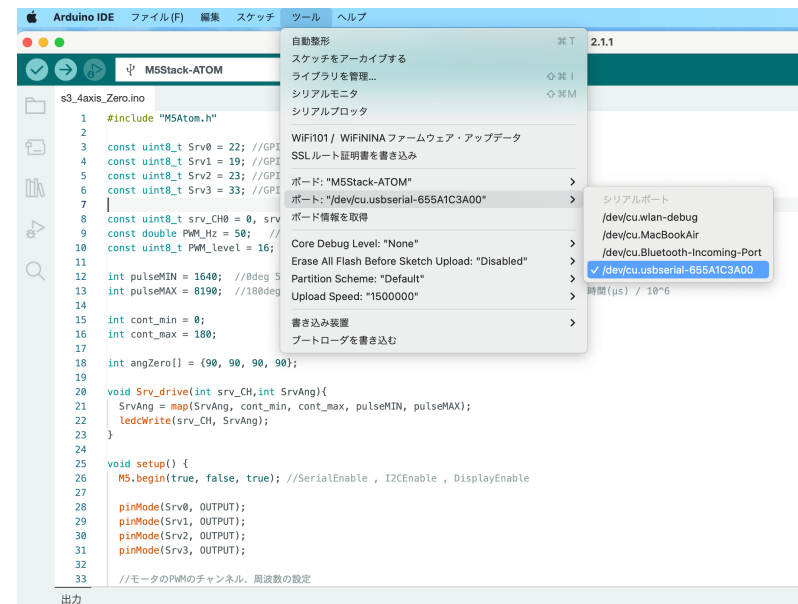
① Arduino IDEを起動後、コードを入力

② ツール→ ボード→ M5Stack Arduino→ M5Stack ATOM



③ M5AtomとパソコンとUSB Type Cで接続

④ ツール→ シリアルポート→ /dev/cu.usbserial-XXXX)

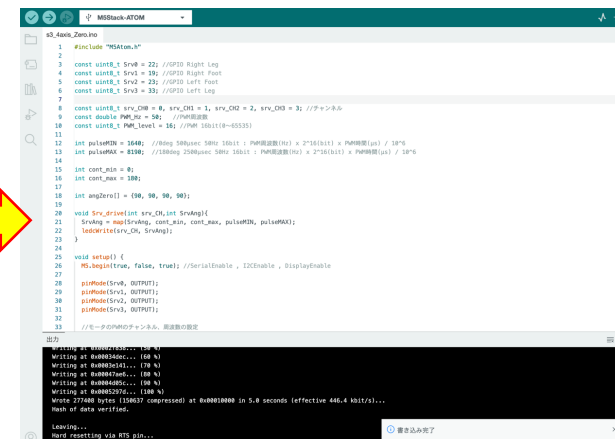
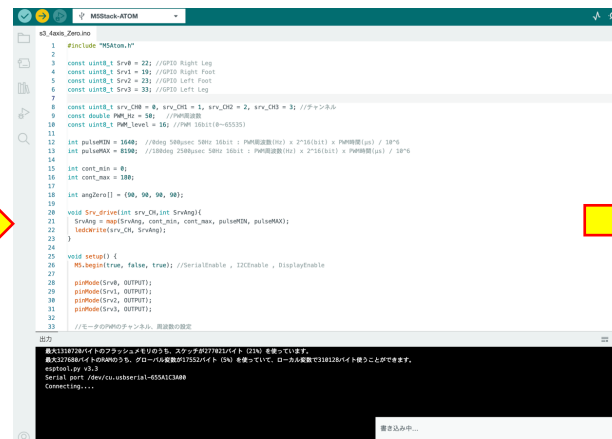
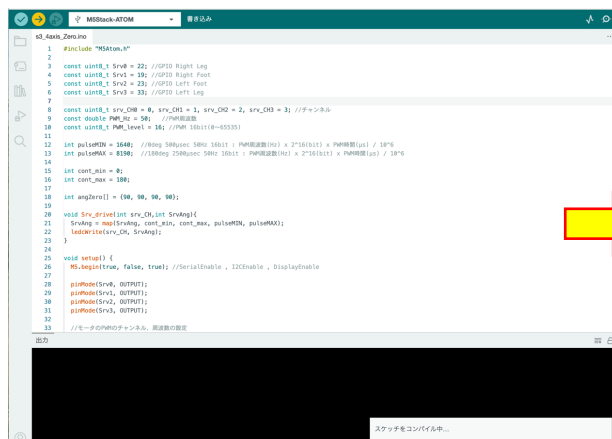
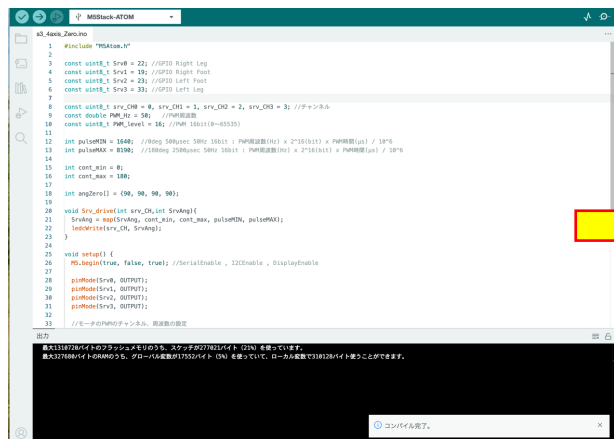


⑤ 「レ」をクリックして  
コンパイル開始

⑥ 「→」をクリックして→ コンパイル→

書き込み開始

⑦ 書き込み終了と同時に  
サーボが動作



## PWM (Pulse Width Modulation)

```
#include "M5Atom.h"
```

```
const uint8_t Srv0 = 22; //GPIO Right Leg  
const uint8_t Srv1 = 19; //GPIO Right Foot  
const uint8_t Srv2 = 23; //GPIO Left Foot  
const uint8_t Srv3 = 33; //GPIO Left Leg
```

```
const uint8_t srv_CH0 = 0, srv_CH1 = 1, srv_CH2 = 2, srv_CH3 = 3; //  
チャンネル
```

```
const double PWM_Hz = 50; //PWM周波数  
const uint8_t PWM_level = 16; //PWM 16bit(0~65535)  
分解能
```

```
int pulseMIN = 1640;  
//0deg 500 μ sec 50Hz 16bit : PWM周波数(Hz) x 2^16(bit) x PWM時間(μ s) / 10^6
```

```
int pulseMAX = 8190;  
//180deg 2500 μ sec 50Hz 16bit : PWM周波数(Hz) x 2^16(bit) x PWM時間(μ s) / 10^6
```

```
int cont_min = 0;  
int cont_max = 180;
```

```
int angZero[] = {90, 90, 90, 90};
```

```
void Srv_drive(int srv_CH,int SrvAng){
```

```
  SrvAng = map(SrvAng, cont_min, cont_max, pulseMIN, pulseMAX); ← 16ビットの値に変換
```

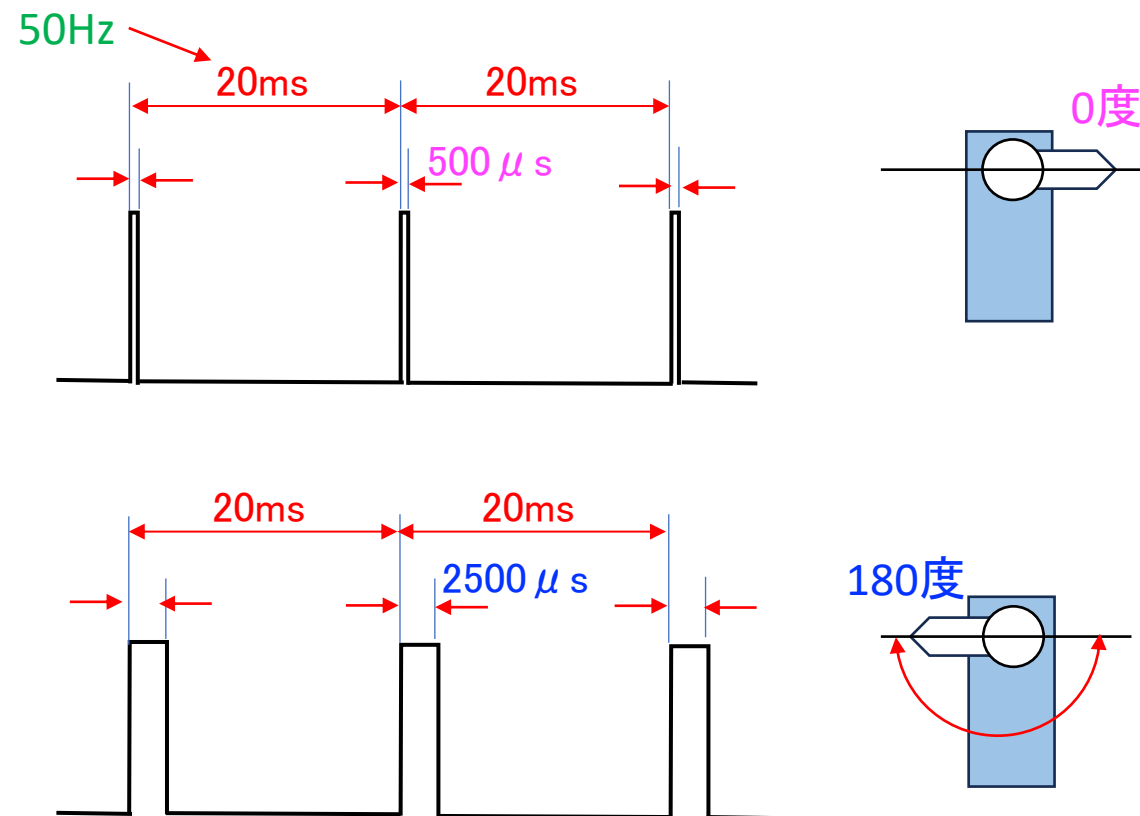
```
  ledcWrite(srv_CH, SrvAng);
```

```
}  
← パルス発生のコマンド (本来は、PWMで段階的に明るさを変えるコマンド)
```

パルス幅設定値 = 周波数[Hz] × 2<sup>16</sup> × 時間[μ s] ÷ 10<sup>6</sup>

サーボ角度0度 1640 = 50 × 2<sup>16</sup> × 500 / 10<sup>6</sup>

サーボ角度180度 8190 = 50 × 2<sup>16</sup> × 2500 / 10<sup>6</sup>



```
void setup() {  
  M5.begin(true, false, true); //SerialEnable , I2CEnable , DisplayEnable  
  
  pinMode(Srv0, OUTPUT);  
  pinMode(Srv1, OUTPUT);  
  pinMode(Srv2, OUTPUT);  
  pinMode(Srv3, OUTPUT);  
  
  //モータのPWMのチャンネル、周波数の設定  
  ledcSetup(srv_CH0, PWM_Hz, PWM_level);  
  ledcSetup(srv_CH1, PWM_Hz, PWM_level);  
  ledcSetup(srv_CH2, PWM_Hz, PWM_level);  
  ledcSetup(srv_CH3, PWM_Hz, PWM_level);  
  
  //モータのピンとチャンネルの設定  
  ledcAttachPin(Srv0, srv_CH0);  
  ledcAttachPin(Srv1, srv_CH1);  
  ledcAttachPin(Srv2, srv_CH2);  
  ledcAttachPin(Srv3, srv_CH3);  
  
  Srv_drive(srv_CH0, angZero[0]);  
  Srv_drive(srv_CH1, angZero[1]);  
  Srv_drive(srv_CH2, angZero[2]);  
  Srv_drive(srv_CH3, angZero[3]);  
}  
  
void loop() {  
  
}
```