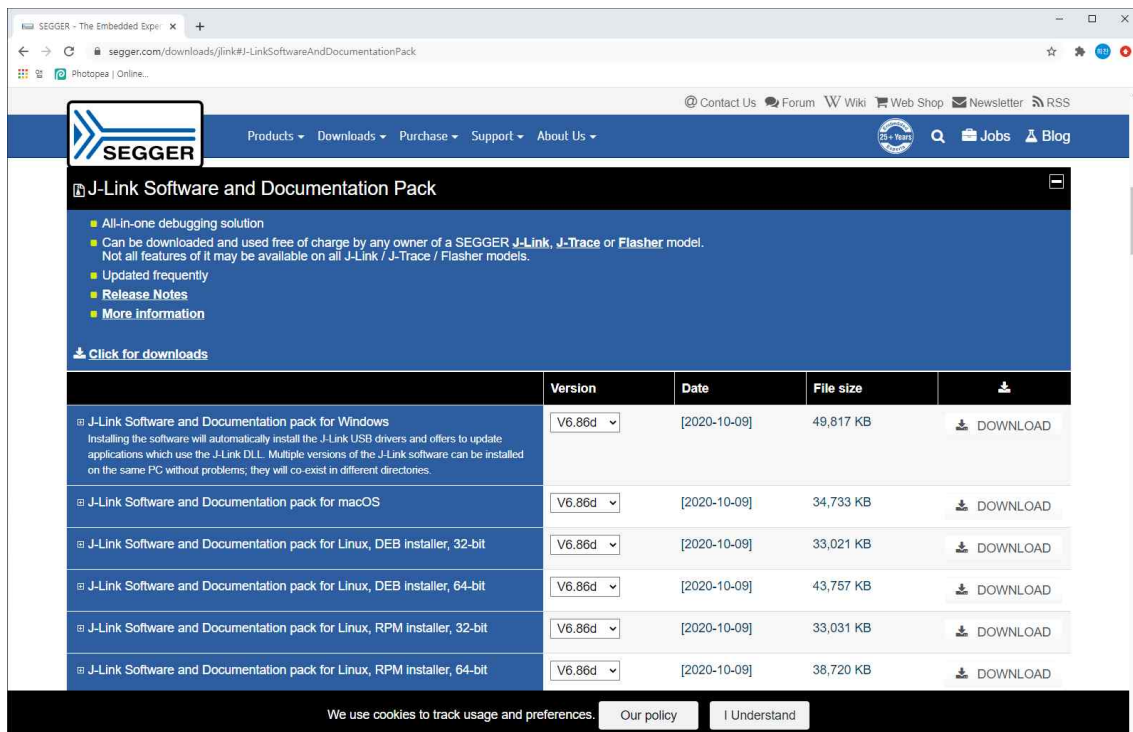


제작 순서

- pcb 제작 -> MCU 부트로더 다운 -> 코딩 -> 하우징 제작

1. MCU chip에 부트로더 및 usb 인식 프로그램 깔기

- 아래 링크로 들어가 Segger J-Link를 다운 받는다.
- <https://www.segger.com/downloads/jlink>

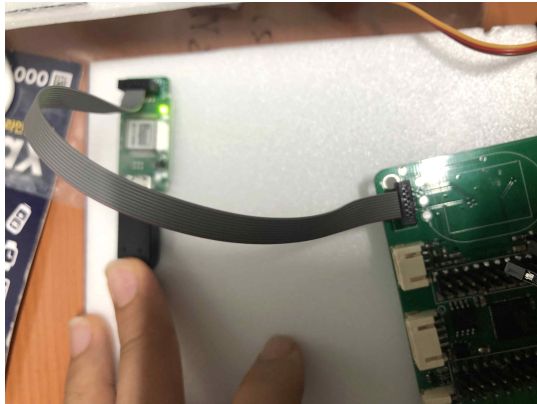


자신의 운영체제에 맞는 버전을 다운.

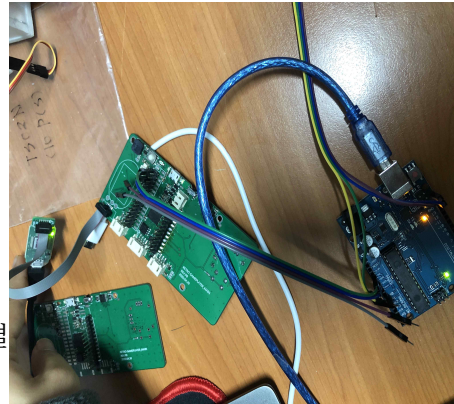
Package Content



Atmel core의 펌웨어를 집어넣기 위해서는 j-link라는 장치가 있어야 한다.



J-link 연결

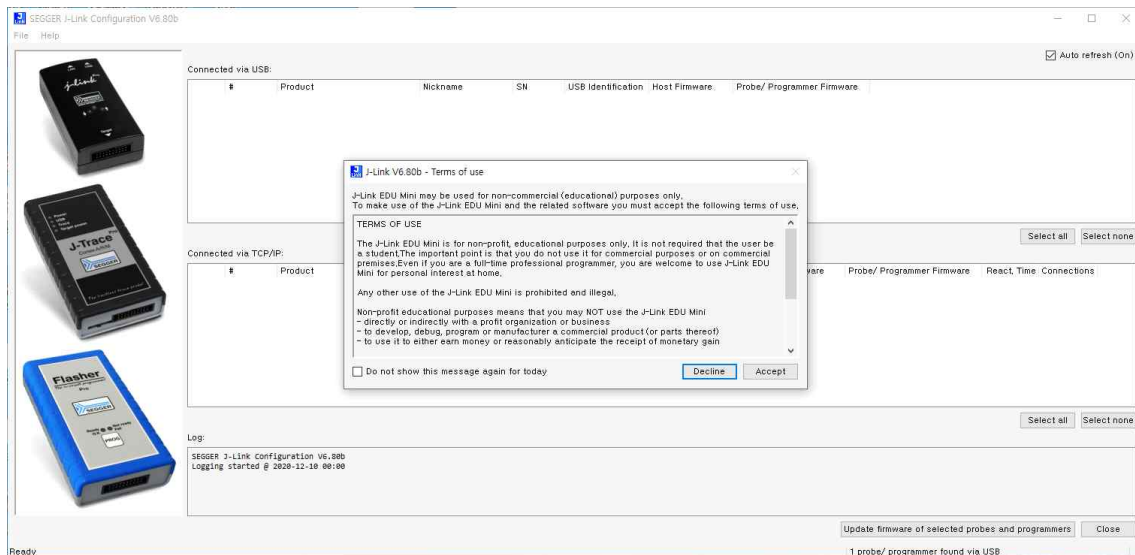


VCC 3.3V 와 GND 연결

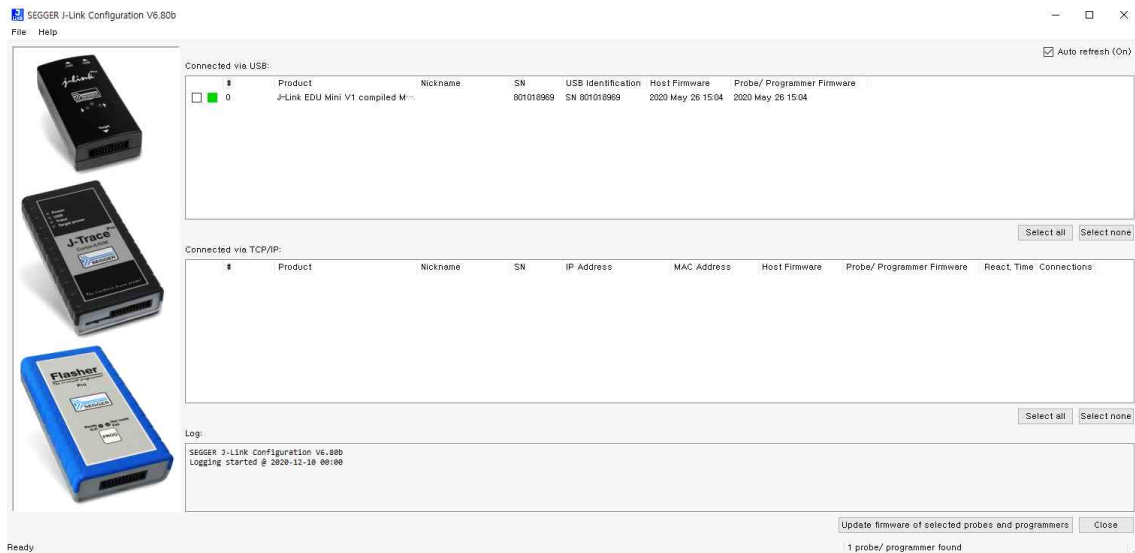
설치를 다 하였으면 j-link configurator를 실행시켜준다.



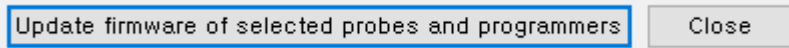
실행시키면 아래와 같이 Terms of use 안내창이 나오는데
do not show this message again for today를 체크해주고 -> Accept를 눌러준다.



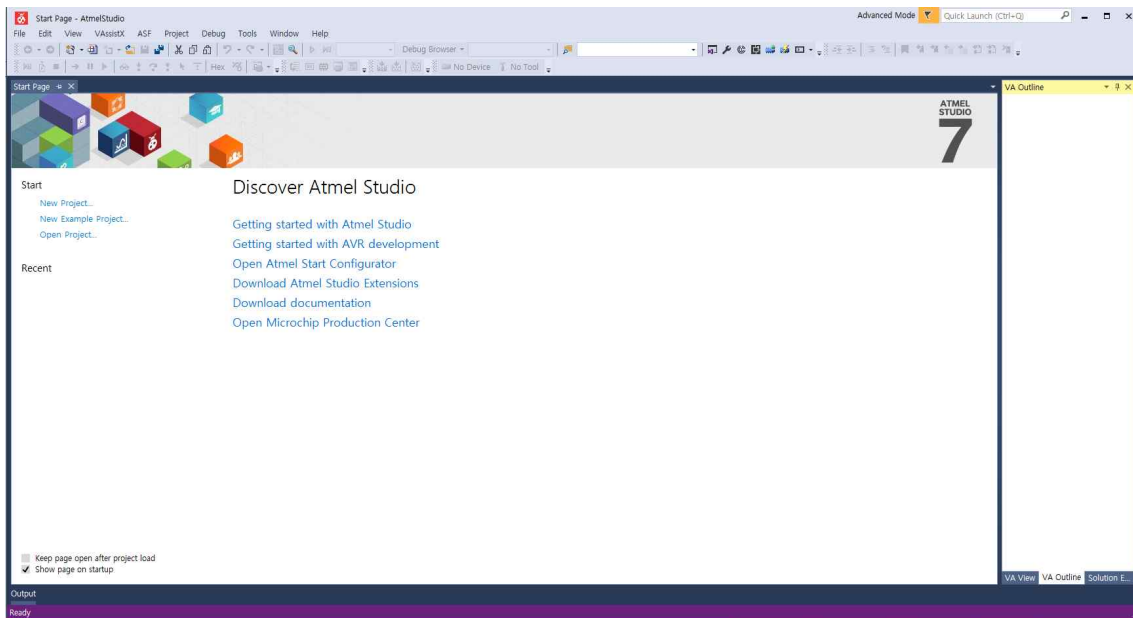
그러면 연결을 스스로 하는 것을 볼 수 있다.



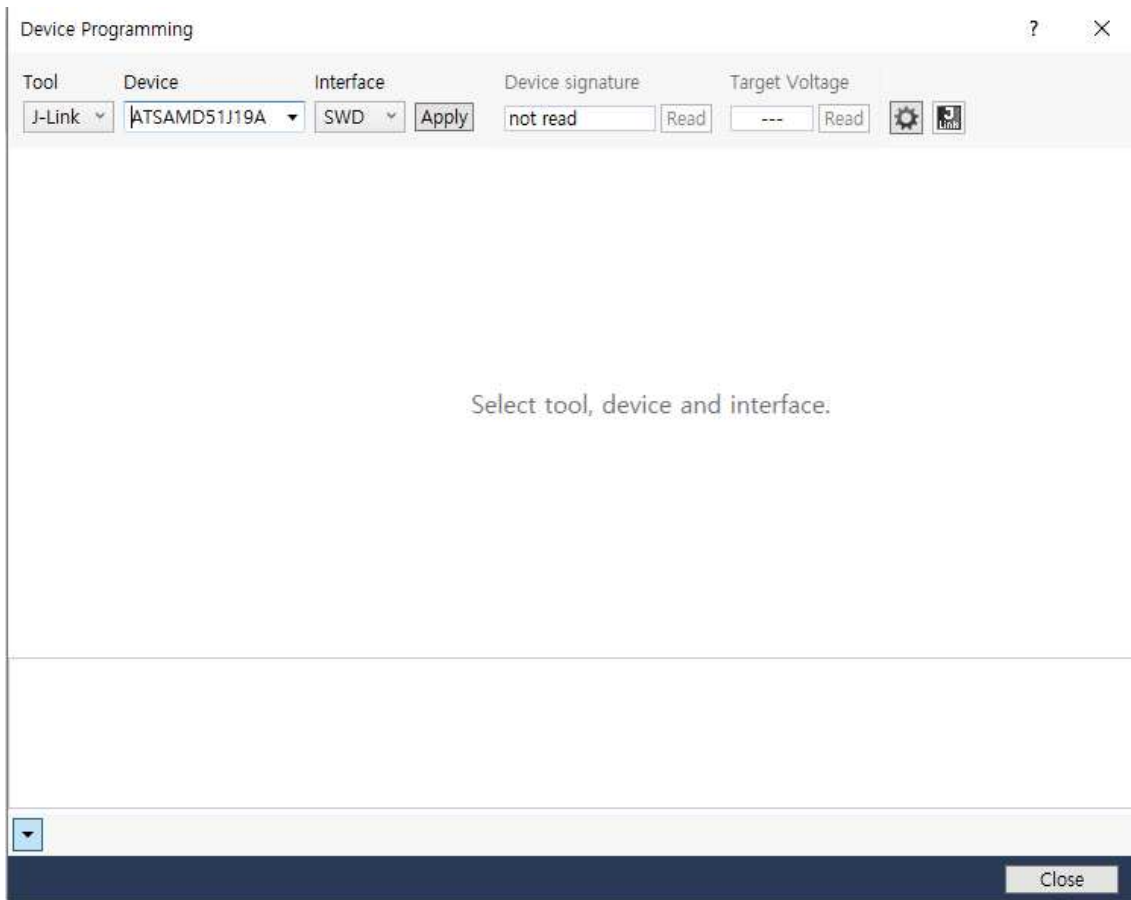
그 후 업데이트 버튼을 눌러주어 최신버전으로 업데이트를 해준다.



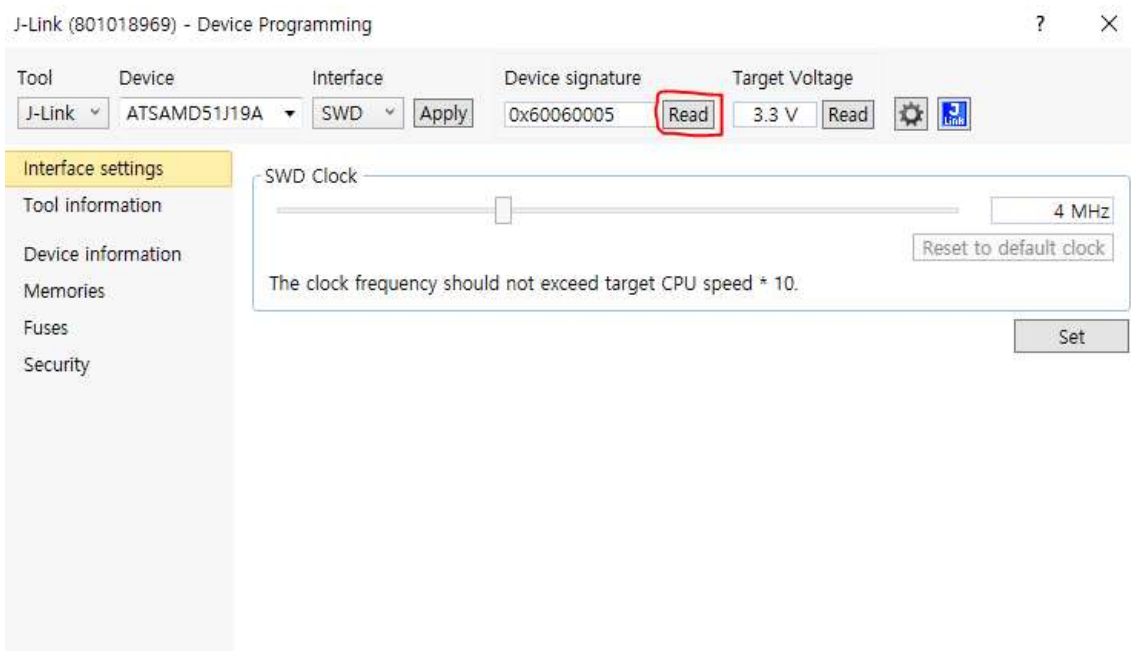
그리고 Atmel Studio를 다운받아 열어준다.



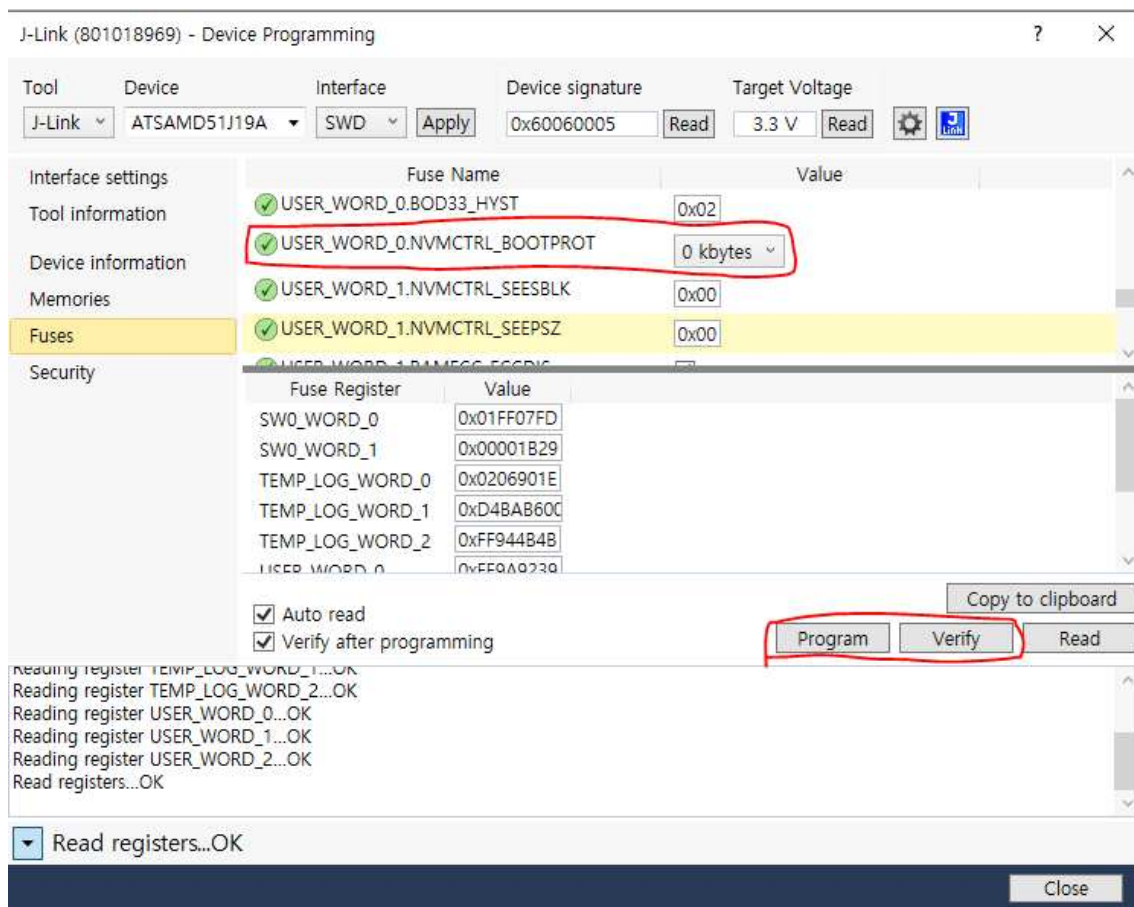
우리는 이곳에서 메뉴 상단에 Tools - Device Programming에 들어간다.



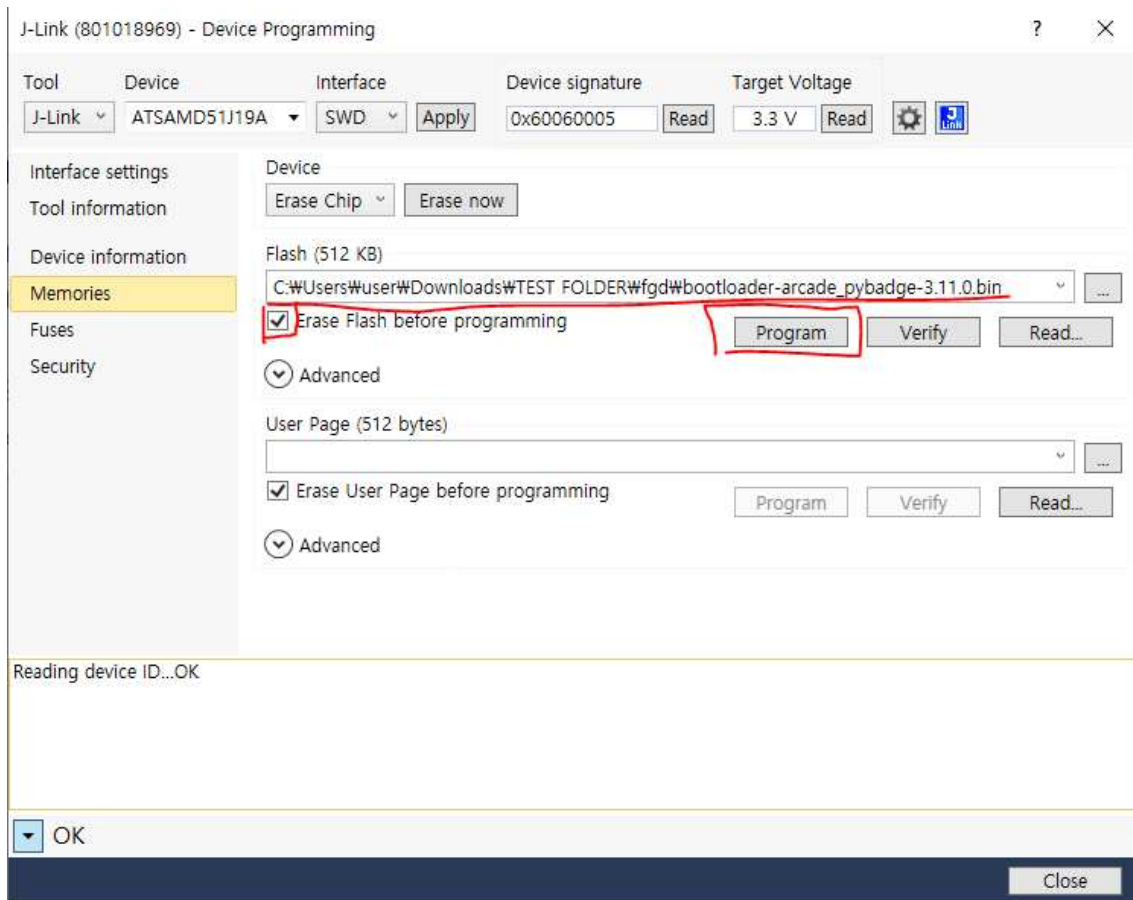
j-link와 ATSAMD51J19A를 설정하고 Apply를 눌러준다.
Apply를 누르면 자신의 J-link 고유 번호가 위에 나온다.



read 버튼을 눌러 Device signature 와 Target Voltage 값을 확인한다.
Reading device ID...OK



Fuse Tab을 클릭 후 USER_WORD_0.NVMCTRL_BOOTPROT 0kbytes 로 설정



Memories 카테고리를 눌러준다.

여기서 Flash에 Bootloader 파일을 넣어서 구울 예정인데 Bootloader 파일은 <https://github.com/adafruit/uf2-samdx1/releases> 로 가서 아래 파일을 다운 받으면 된다.

	bootloader-arcade_pybadge_lc-3.11.0.bin	16 KB
	bootloader-arcade_pygamer-3.11.0.bin	16 KB

위의 주소가 없어졌으면

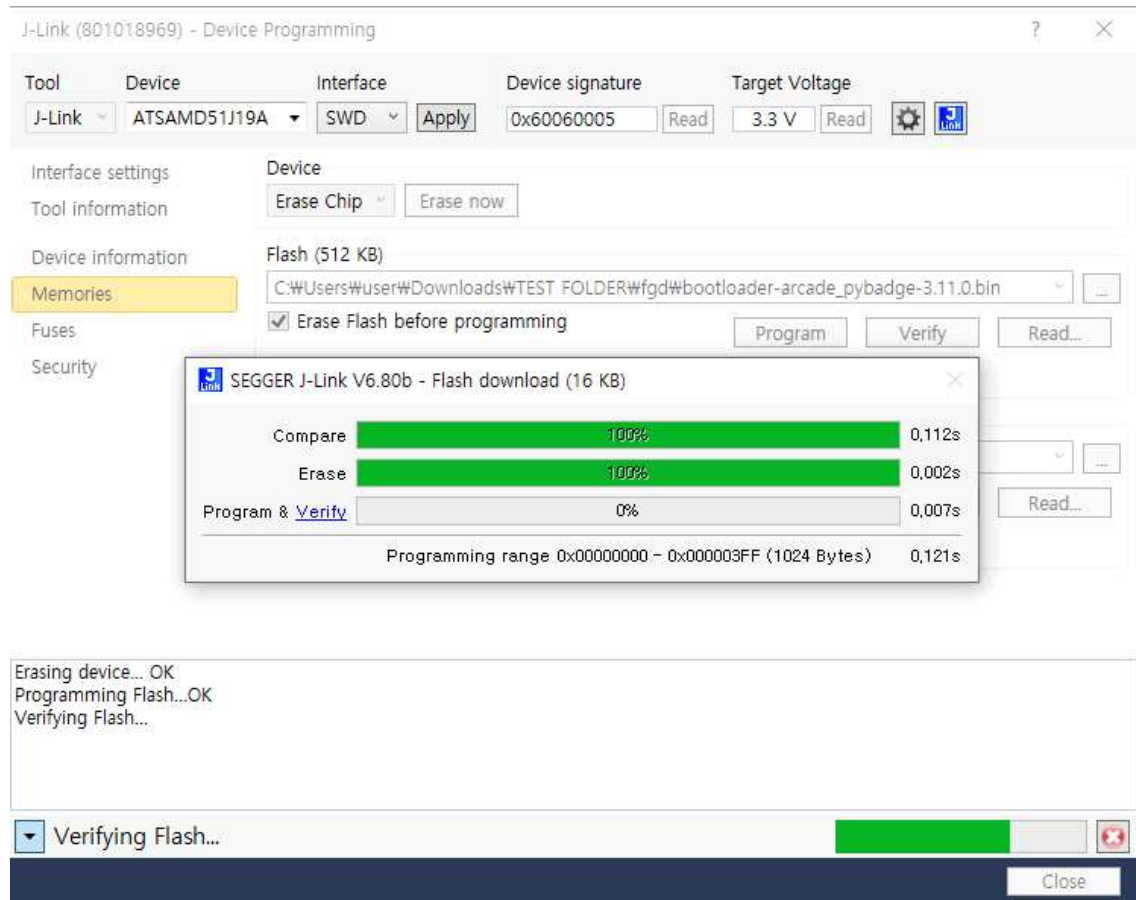
<https://learn.adafruit.com/how-to-program-samd-bootloaders?view=all> 주소의 버튼을 클릭하면 넘어가진다.

Grab a Bootloader

You'll also want to **download a compiled bootloader binary** (.bin file) for the board you're recovering. These can be found on the [Adafruit/uf2-samdx1 repository](https://github.com/adafruit/uf2-samdx1):

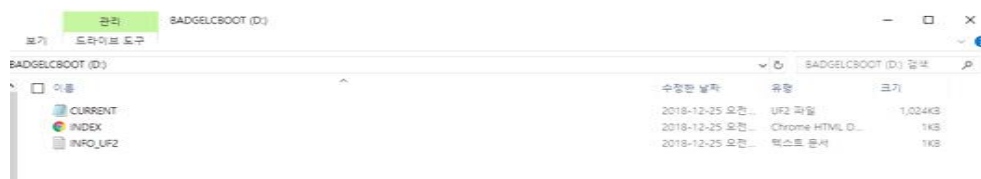


아래와 같이 경로를 입력한후 program을 눌러 구워준다.



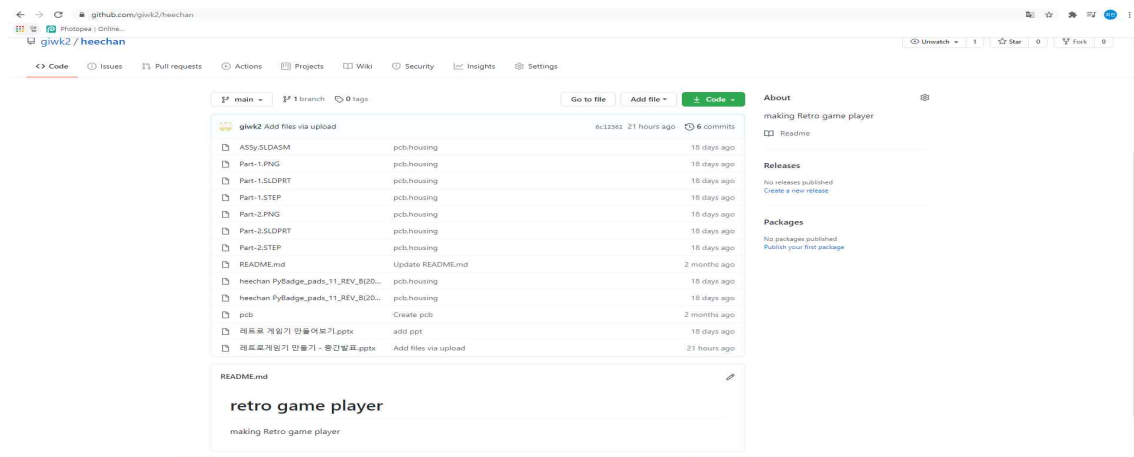
Erasing device... OK
Programming Flash...OK
Verifying Flash...OK

그러면 최종적으로 아래와 같이 부트로더에 진입할 수 있게 된다.

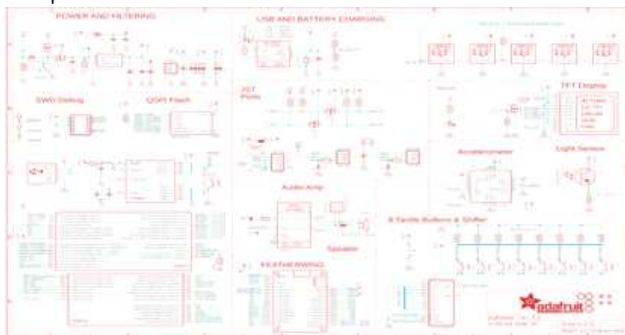


1. PCB 제작

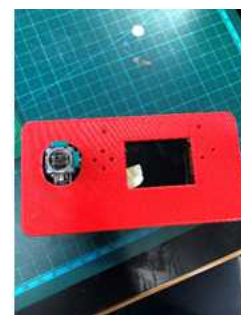
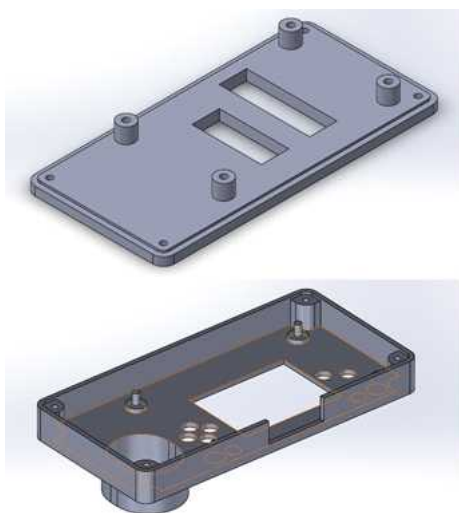
- 최종 제작 본은 <https://github.com/giwk2/heechan> 링크에 올려두었음.



- 위의 pcb를 변형하여 사용하거나 그대로 사용하여 제작.
회로도



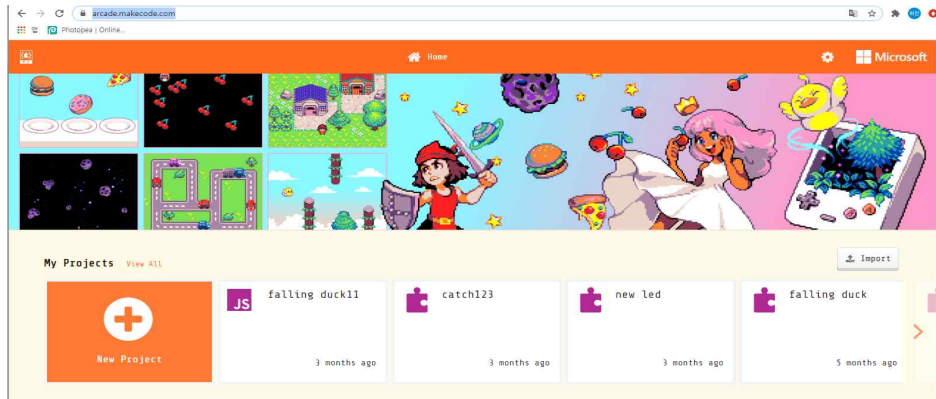
1. 하우징 제작



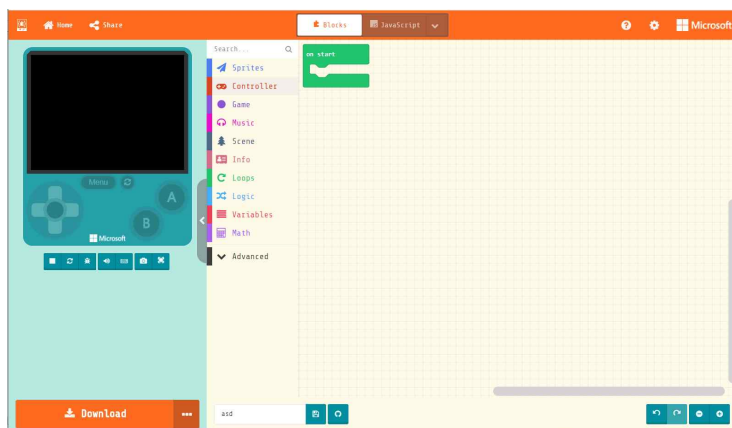
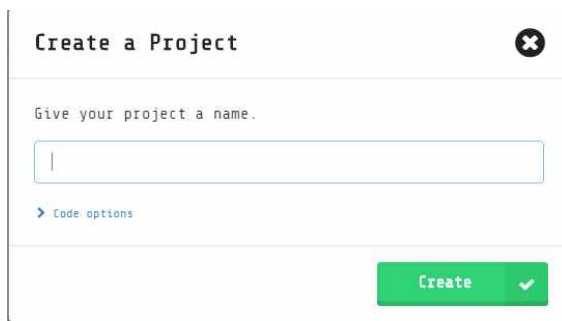
- 3d modeling -> 3d printer 제작

1. 코딩

- 홈페이지 방문 <https://arcade.makecode.com/>



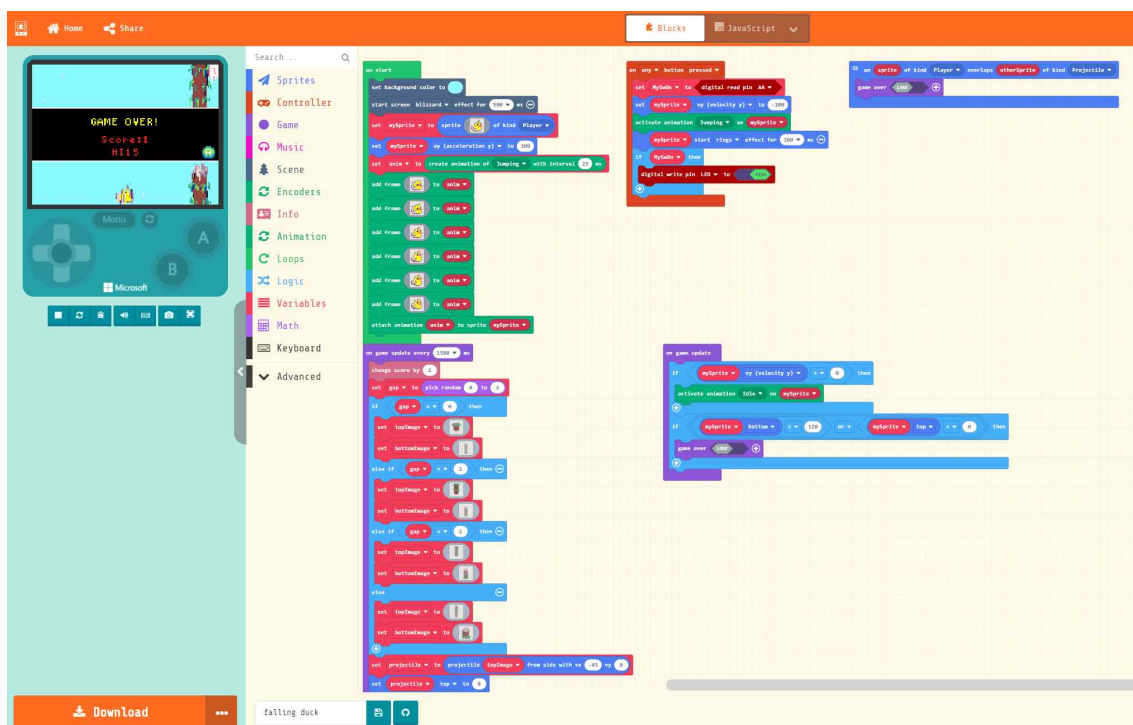
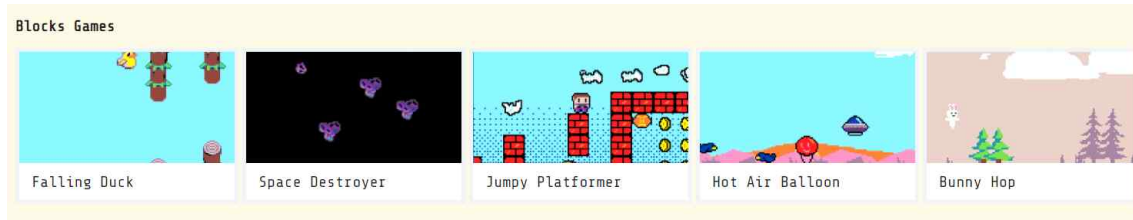
new project 클릭 -> project name 설정 -> Create 클릭



블록코드나 JavaScript를 활용하여 코드를 작성하면 됨.

예제를 실행해보기

1. falling duck 클릭 ->

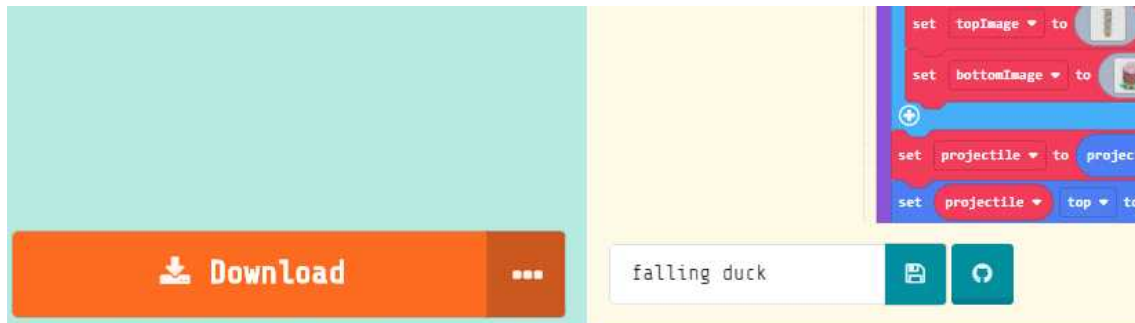


1. 시뮬레이터의 A,B 버튼을 눌러 오리 of 점프를 이용해보자.

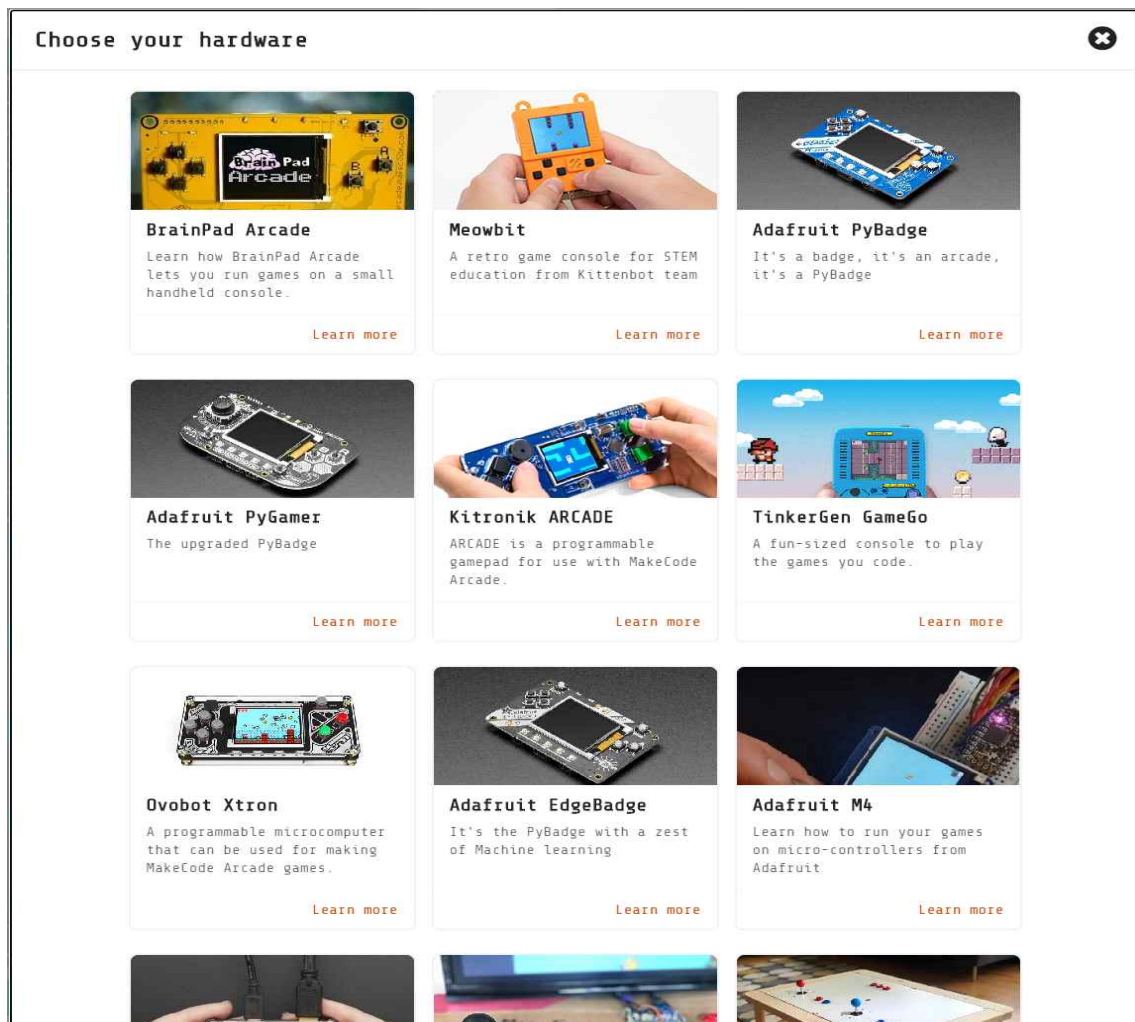
2. 우리의 게임기에 코드를 집어넣어 보자.



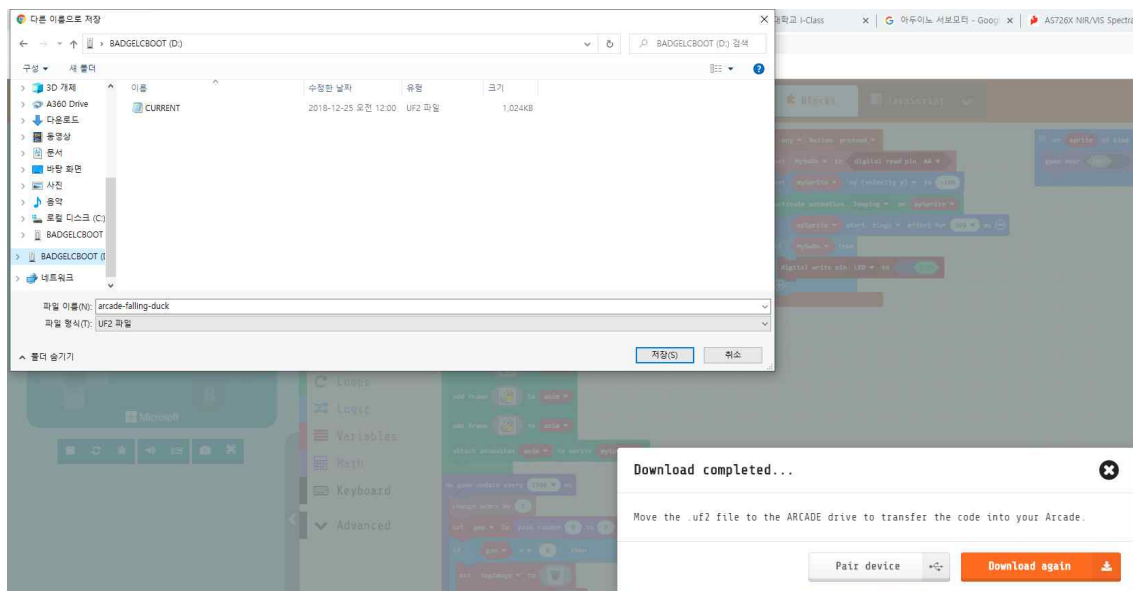
- 먼저 보드를 컴퓨터에 연결하여 BADGELCBOOT(부트로더)를 띄운다. 자동으로 열림.



- 다운로드 버튼 클릭

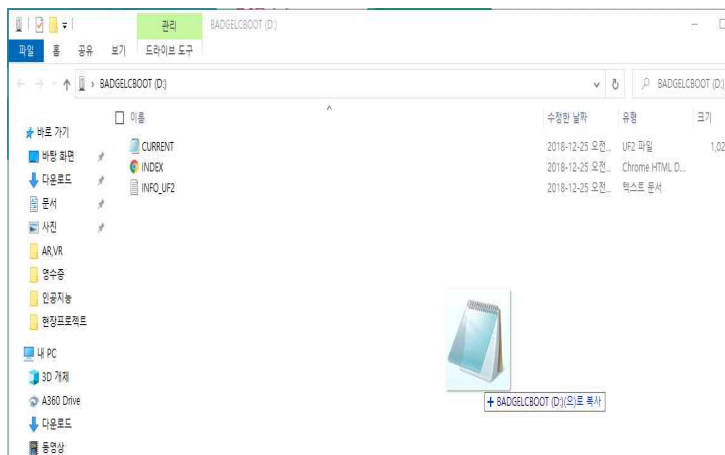


- Adafruit PyBadge를 클릭한다.



어디에 저장할지 물어본다.

- 여기서 주의할 점은 바로 부트로더에 저장을 하면 기존의 UF2 파일은 없어지고 지금의 파일이 덮어써짐 저장이 된다. 그리고 UF2 파일은 CURRENT 파일로 바뀌어 지게된다.
- 되도록 다른 폴더에 저장하여 따로 이동하여 넣는 것을 추천한다.
- 다운로드 폴더에 저장



arcade-falling-duck 2020-06-11 오전 9:18 UF2 파일 632KB

UF2 파일을 부트로더에 옮기면 자동으로 실행이 된다.



(혹시 이런 화면이 뜨고 d드라이브 이름이 "circuitpy"

라면 뒤쪽의 reset 버튼을 두 번 연속으로 누르면 bootloader mode로 진입할 수 있다.

가속도 센서 사용법(코드)

```
forever(function () {
```

```
  myx1 = controller.acceleration(ControllerDimension.X)
```

```
  myy1 = controller.acceleration(ControllerDimension.Y)
```

```
  if (myx1 > 400) {
```

```
    tempy = tempy + 2
```

```
    if (tempy > 155) {
```

```
      tempy = 155
```

```
    }
```

```
    basket.x = tempy
```

```
  } else {
```

```
    tempy = tempy - 2
```

```
    if (tempy < 0) {
```

```
      tempy = 0
```

```
    }
```

```
    basket.x = tempy
```

```
}
```

```
  if (myy1 < 400) {
```

```
    tempx = tempx + 2
```

```
    if (tempx > 100) {
```

```
      tempx = 100
```

```
    }
```

```
    basket.y = tempx
```

```
  } else {
```

```
    tempx = tempx - 2
```

```
    if (tempx < 0) {
```

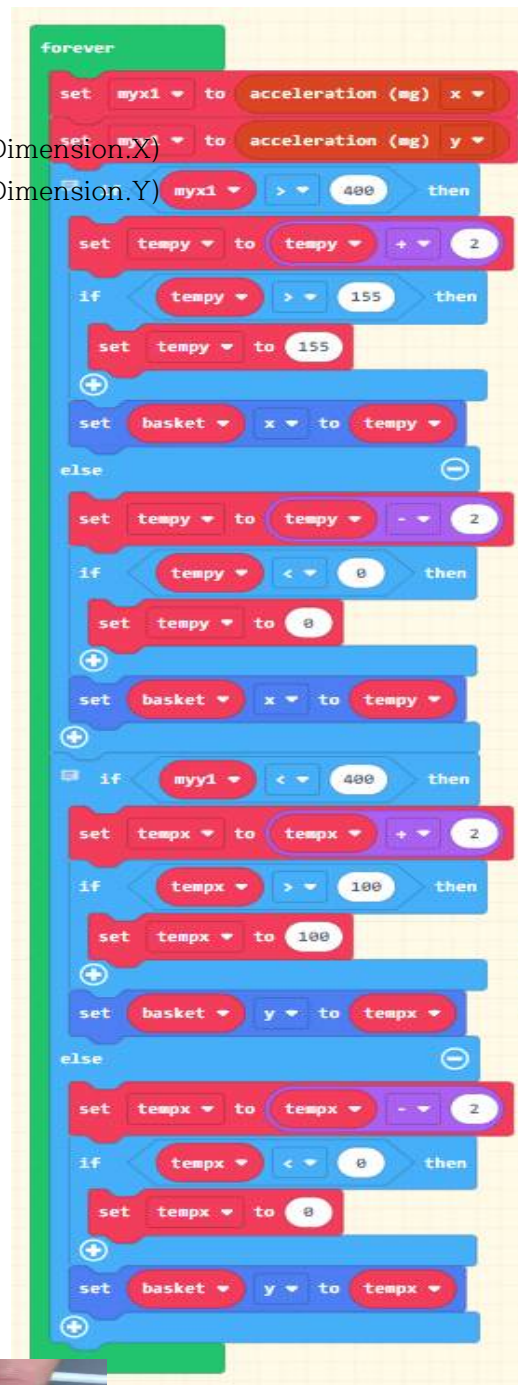
```
      tempx = 0
```

```
    }
```

```
    basket.y = tempx
```

```
}
```

```
})
```



가속도 센서 Test

조이스틱 사용법(코드)

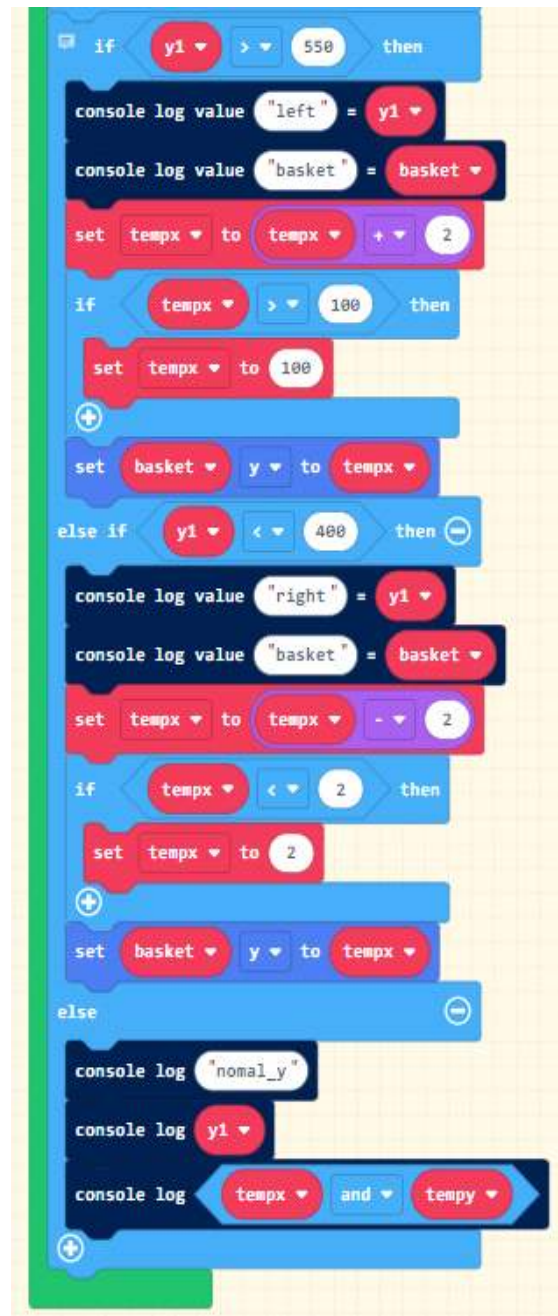
```
forever(function () {  
  x1 = pins.A2.analogRead()  
  y1 = pins.A3.analogRead()  
  if (x1 > 550) {  
    console.logValue("up", x1)  
    console.logValue("basket", basket)  
    tempy = tempy + 2  
    if (tempy > 155) {  
      tempy = 155  
    }  
    basket.x = tempy  
  } else if (x1 < 400) {  
    console.logValue("down", x1)  
    console.logValue("basket", basket)  
    tempy = tempy - 2  
    if (tempy < 0) {  
      tempy = 0  
    }  
    basket.x = tempy  
  } else {  
    console.log("nomal_x")  
    console.log(x1)  
  }  
  if (y1 > 550) {  
    console.logValue("left", y1)  
    console.logValue("basket", basket)  
    tempx = tempx + 2  
    if (tempx > 100) {  
      tempx = 100  
    }  
    basket.y = tempx  
  } else if (y1 < 400) {  
    console.logValue("right", y1)  
    console.logValue("basket", basket)  
    tempx = tempx - 2  
    if (tempx < 2) {  
      tempx = 2  
    }  
    basket.y = tempx  
  }  
}
```




```

    } else {
      console.log("nomal_y")
      console.log(y1)
      console.log(tempx && tempy)
    }
  })

```



Joystick Test

빛 센서 사용법(코드)

```
controller.onLightConditionChanged(ControllerLightCondition.Bright, function () {  
    basket.x = 50  
})
```

```
controller.onLightConditionChanged(ControllerLightCondition.Dark, function () {  
    basket.x = 80  
})
```



Light sensor Test