

---

# SEANLAB Tech Documentation

*Release REL*

**sean**

**Dec 13, 2018**



---

## Contents

---

<b>1</b>	<b>Part 1: 2018.11.15 ~</b>	<b>3</b>
1.1	chapter 1: minecraft . . . . .	3
1.1.1	1.1 All tech books . . . . .	3
1.1.2	1.1 minecraft+scratch . . . . .	3
1.1.3	1.2 A4S . . . . .	4
1.1.4	1.3 Scratch 3.0 . . . . .	5
1.2	chapter 2: A4S . . . . .	6
1.2.1	raspberrypi . . . . .	6
1.3	chapter 3: S4A . . . . .	6
<b>2</b>	<b>Part 2: 2</b>	<b>7</b>
2.1	Chapter 0: About . . . . .	7
2.1.1	Thanks to . . . . .	7
2.1.2	SEAN's Paradise . . . . .	7
2.2	chapter 1: S4A . . . . .	7
2.3	chapter 2: Snap4Arduino . . . . .	8
2.4	chapter 3: ScratchX . . . . .	8
2.4.1	mblock . . . . .	9
2.4.2	. . . . .	9
2.4.3	minecraft scratch . . . . .	9
2.5	chapter 4: johy-five . . . . .	21
2.6	chapter 5: Arduino Uno . . . . .	21
2.7	chapter 6: mblock . . . . .	22
2.8	chapter 7: code.org . . . . .	23
2.9	chapter 8: AVR GCC . . . . .	24
2.10	chapter 9: Veyon . . . . .	24
2.11	chapter 10: blockly . . . . .	25
2.12	chapter 11: Ware house . . . . .	25
2.13	chapter 12: Docker . . . . .	26
2.14	chapter 13: Microbit . . . . .	26
2.15	chapter 14: Minecraft . . . . .	27
2.16	chapter 15: Ardublock . . . . .	27
2.17	chapter 16: Raspberry Pi . . . . .	27
2.18	chapter 17: Minitank . . . . .	27
2.19	chapter 18: MakeyMakey . . . . .	28
2.20	chapter 19: WAVGAT Uno . . . . .	28

2.21	chapter 20: Cloud9 . . . . .	28
<b>3</b>	<b>Part 3: 3</b>	<b>29</b>
<b>4</b>	<b>Part 4: 4</b>	<b>31</b>
<b>5</b>	<b>Part 5: 5</b>	<b>33</b>

SEANLAB TECH

written by sean base on following books



Github | <https://github.com/newsteinking/seanlabtech.git>



# CHAPTER 1

---

Part 1: 2018.11.15 ~

---

## 1.1 chapter 1: minecraft

### 1.1.1 1.1 All tech books

<http://www.allitebooks.com/>

.

Beginning Blockchain Bitcoin Essentials Bitcoin For Dummies Bitcoin for the Befuddled Building Games with  
Ethereum Smart Contracts Decentralized Applications Mastering Bitcoin, 2nd Edition

### 1.1.2 1.1 minecraft+scratch

<https://github.com/mpatrascu/ScratchMC>

### ScriptCraft

<https://github.com/walterhiggins/ScriptCraft/tree/master>

<https://github.com/tclavier/docker-scriptcraft>

### python 3.7 install

```
sudo apt-get update -y
sudo apt-get install build-essential tk-dev libncurses5-dev libncursesw5-dev
↳ libreadline6-dev libdb5.3-dev libgdbm-dev libsqlite3-dev libssl-dev libbz2-dev
↳ libexpat1-dev liblzma-dev zlib1g-dev libffi-dev -y
```

```
wget https://www.python.org/ftp/python/3.7.0/Python-3.7.0.tar.xz
```

(continues on next page)

(continued from previous page)

```
tar xf Python-3.7.0.tar.xz

cd Python-3.7.0
./configure
make -j 4
sudo make altinstall

sudo ln -sf /usr/local/bin/python3.7 /usr/bin/python

/usr/local/lib/python3.5/dist-packages/mcpi/minecraft.py

/home/pi/Desktop/mcpi-scratch-server.desktop

Exec=lxterminal --command="/bin/bash -c 'python2 -i /home/pi/Documents/MCPiScratch/
↪mcpi-scratch/mcpi-scratch-server.py' "
```

## python pip install

```
sudo apt-get install python3-pip
```

To get the Python 2 version:

```
sudo apt-get install python-pip
```

For example, the following command installs the Unicorn HAT library for Python 3:

pip3 install unicornhat The following command installs the Unicorn HAT library for Python 2:

```
pip install unicornhat
```

## denosawr

<https://denosawr.github.io/mcpi-scratch/>

```
curl -s https://denosawr.github.io/mcpi-scratch/install.sh | sh
```

```
mcpi-scratch python2 /home/pi/Desktop/mcpi-scratch-server.desktop
```

## 1.1.3 1.2 A4S

<https://github.com/damellis/A4S>

## raspberrypi

```
git clone https://github.com/damellis/A4S
```

```
git clone https://github.com/firmata/processing
```

```
>./build.sh
```

```
sudo apt-get install librtx-java
```

- USB

```
dmesg | tail *run.sh
```



```
-Djava.library.path=/usr/lib/jni
java -d32 -Djava.library.path=/usr/lib/jni -jar A4S.jar $@
*
./run.sh /dev/ttyUSB0
```

### 1.1.4 1.3 Scratch 3.0

#### github

<https://github.com/LLK/scratch-gui>

#### electron

```
npm install -g asar
asar pack app app.asar
asar extract app.asar destfolder
```

#### scratch extension

<https://scratch.mit.edu/discuss/topic/289503/>

node\_modulesscratch-blocks

npm install

npm link

You can also use the command `python build.py` to build scratch blocks, but from my experience, changes aren't reflected in the GUI unless you run `npm link`.

<https://github.com/larjohn/rpi-scratch3-mqtt>

<https://github.com/larjohn/python3-pigpio-scratch3-mqtt>

scratch-guinode\_modulesscratch-blocksblocks\_vertical extensions.js default\_toolbox.js

scratch-guinode\_modulesscratch-blocksmediaextensions seanpen-block-icon.svg

scratch-guinode\_modulesscratch-blocksi18n

scratch-guinode\_modulesscratch-110nscrip build-data.js build-i18n-src.js

scratch-guinode\_modulesscratch-vmsrcextensions

scratch3\_seanpen index.js Scratch3PenSEANBlocks

scratch-guinode\_modulesscratch-vmsrcextension-support

extension-manager.js const Scratch3PenSEANBlocks = require('../extensions/scratch3\_seanpen');

scratch-guisreliblibrariesextensions

index.jsx id id

```
const builtinExtensions = { pen: Scratch3PenBlocks, seanpen: Scratch3PenSEANBlocks, wedo2:
  Scratch3WeDo2Blocks, music: Scratch3MusicBlocks, microbit: Scratch3MicroBitBlocks, text2speech:
  Scratch3Text2SpeechBlocks, translate: Scratch3TranslateBlocks, videoSensing: Scratch3VideoSensingBlocks,
  speech2text: Scratch3Speech2TextBlocks, ev3: Scratch3Ev3Blocks
```

```
};  
scratch-guisrclibrariesextensions  
index.jsx  
import seanpenImage from './seanpen.png'; import seanpenInsetImage from './seanpen-small.svg';  
cratch-guisrclibrariesextensions  
scratch-guitranslationsmessagessrclibrariesextensions index.json
```

### **mblock bluetooth**

<https://github.com/inker610566/MBlocky>

### **Microbit docker**

```
docker pull kadok0520/pxt-microbit  
docker run -d -p 80:80 --name pxt-microbit01 kadok0520/pxt-microbit
```

## **1.2 chapter 2: A4S**

<https://github.com/damellis/A4S>

### **1.2.1 raspberrypi**

```
git clone https://github.com/damellis/A4S  
git clone https://github.com/firmata/processing  
>./build.sh  
sudo apt-get install librx-tx-java  
run.sh  
  • USB  
dmesg | tail
```

## **1.3 chapter 3: S4A**

## 2.1 Chapter 0: About

SEANLAB Tech Note

by sean

### 2.1.1 Thanks to



- sean
- Mr Ju SS
- OSS Members

### 2.1.2 SEAN's Paradise

I think that My Life as Software Engineer was terrible , but it's role for social is important so, I keep going for better life & software development

## 2.2 chapter 1: S4A

1.S4AFirmware16.ino download

2. S4A

<http://s4a.cat/>

2 digital inputs (digital pins 2 and 3),

3 analog outputs (digital pins 5, 6 and 9), 3 digital outputs (pins 10, 11 and 13) and 4 special outputs to connect Parallax continuous rotation servomotors (digital pins 4, 7, 8 and 12).

basic example 1.button & led

Digital 2-3 LED 10~13 OK

2.Photoresistor and LED

Analog S=0 G V=5v

## 2.3 chapter 2: Snap4Arduino

1.Snap4Arduino - Web Install Version

install standard firmata -

**chrome extension** browser chrome://extensions

Snap4Arduino connector install

drag & drop plugin

2.connect 3.

netsblox

mongo

docker run -d -p 27017:27017 -v /home/sean/netsblox/NetsBlox/db mongo

docker run -it -p 8080:8080 -e MONGO\_URI='mongodb://192.168.42.130:27017/netsblox' -v /home/sean/netsblox/NetsBlox/media/blob-data netsblox/server

## 2.4 chapter 3: ScratchX

Graphical Language for Arduino

<http://blog.ardublock.com/2012/06/11/setup-ardublock-development-in-eclipse/>

\*Redbot controller for scratchx <https://github.com/MrYsLab/rb4s>

nodejs +scratch x

<https://www.npmjs.com/package/http-server>

**nodejs** 1.npm install http-server -g 2.http-server ./public

IE11 – OK example-firmata- standard firmata - upload IE —close - plugin

Arduino Uno +ScratchX =>OK Arduino Mega=ScratchX ==> OK

<<original>>

[http://scratchx.org/?url=http://khanning.github.io/scratch-arduino-extension/arduino\\_extension.js=ko#scratch](http://scratchx.org/?url=http://khanning.github.io/scratch-arduino-extension/arduino_extension.js=ko#scratch)

IE11 — OK D:SEANLAB2ScratchXrealworld\_gitnode-scratchx

node app.js node scratchx.js

[http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/arduino\\_extension.js=ko#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/arduino_extension.js=ko#scratch)

scratch\_microbit extension OK

scratch-microbit-extension/firmware/ makecode-microbit-scratch-extension.hex install

D:SEANLAB2ScratchXgit\_downscratch-microbit-extension node index.js ==> Green light

[http://scratchx.org/?url=https://jaafreitas.github.io/scratch-microbit-extension/scratch\\_microbit.js&lang=en#scratch](http://scratchx.org/?url=https://jaafreitas.github.io/scratch-microbit-extension/scratch_microbit.js&lang=en#scratch)

[http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/scratch\\_microbit.js=en#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/scratch_microbit.js=en#scratch)

[http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/makecode-microbit-scratch-extension.js=en&url=http://localhost:8080/scratch\\_extensions/examples/Compass.sbx#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/makecode-microbit-scratch-extension.js=en&url=http://localhost:8080/scratch_extensions/examples/Compass.sbx#scratch)

[http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/makecode-microbit-scratch-extension.js=en&url=http://localhost:8080/scratch\\_extensions/examples/Button.sbx#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/makecode-microbit-scratch-extension.js=en&url=http://localhost:8080/scratch_extensions/examples/Button.sbx#scratch) [http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/makecode-microbit-scratch-extension.js=en&url=http://localhost:8080/scratch\\_extensions/examples/Display.sbx#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/makecode-microbit-scratch-extension.js=en&url=http://localhost:8080/scratch_extensions/examples/Display.sbx#scratch) [http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/makecode-microbit-scratch-extension.js=en&url=http://localhost:8080/scratch\\_extensions/examples/Tilt.sbx#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/makecode-microbit-scratch-extension.js=en&url=http://localhost:8080/scratch_extensions/examples/Tilt.sbx#scratch)

## 2.4.1 mblock

[http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/arduino\\_extension.js=ko#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/arduino_extension.js=ko#scratch)

[http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/arduino\\_extension.js=ko&url=http://localhost:8080/scratch\\_extensions/examples/ArduinoBasic.sbx#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/arduino_extension.js=ko&url=http://localhost:8080/scratch_extensions/examples/ArduinoBasic.sbx#scratch)

[http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/arduino\\_extension.js=ko&url=http://localhost:8080/scratch\\_extensions/makeblock.js=ko&url=http://localhost:8080/scratch\\_extensions/examples/demo.sbx#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/arduino_extension.js=ko&url=http://localhost:8080/scratch_extensions/makeblock.js=ko&url=http://localhost:8080/scratch_extensions/examples/demo.sbx#scratch)

[http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/random\\_wait\\_extension.js=ko&url=http://localhost:8080/scratch\\_extensions/examples/ArduinoBasic.sbx#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/random_wait_extension.js=ko&url=http://localhost:8080/scratch_extensions/examples/ArduinoBasic.sbx#scratch)

[http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/scratch\\_microbit.js=ko&url=http://localhost:8080/scratch\\_extensions/examples/ArduinoBasic.sbx#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/scratch_microbit.js=ko&url=http://localhost:8080/scratch_extensions/examples/ArduinoBasic.sbx#scratch)

[http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/arduino\\_extension.js=ko&url=http://localhost:8080/scratch\\_extensions/examples/scratchx.sbx#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/arduino_extension.js=ko&url=http://localhost:8080/scratch_extensions/examples/scratchx.sbx#scratch)

1 - led A -pin 9 - set led a on

## 2.4.2

<https://github.com/amandaghassaei/ScratchSerialExtension> XXX

[http://localhost:8081/?url=http://localhost:8081/scratch\\_extensions/serial\\_port\\_extension.js=ko#scratch](http://localhost:8081/?url=http://localhost:8081/scratch_extensions/serial_port_extension.js=ko#scratch) XXXX

## 2.4.3 minecraft scratch

<http://scratchx.org/?url=https://mojang.github.io/cc-scratch.js#scratch> [http://localhost:8080/?url=http://localhost:8080/scratch\\_extensions/cc-scratch.js=ko#scratch](http://localhost:8080/?url=http://localhost:8080/scratch_extensions/cc-scratch.js=ko#scratch)

```
(function(ext) {
  var poller = null;
  var device = null;
  var status = false;
  var _selectors = {};
  var _buffer = [];
  var _isParseStartIndex = 0;
```

(continues on next page)

(continued from previous page)

```
var _isParseStart = false;
var ports = {
  Port1: 1,
  Port2: 2,
  Port3: 3,
  Port4: 4,
  M1:9,
  M2:10,
  'on board':7,
  'light sensor on board':8
};
var slots = {
  Slot1:1,
  Slot2:2
};
var switchStatus = {
  On:1,
  Off:0
};
var buttonStatus = {
  pressed:0,
  released:1
}
var shutterStatus = {
  Press:0,
  Release:1,
  'Focus On':2,
  'Focus Off':3,
};
var axis = {
  'X-Axis':1,
  'Y-Axis':2,
  'Z-Axis':3
};
var ircodes = { "A":69,
  "B":70,
  "C":71,
  "D":68,
  "E":67,
  "F":13,
  "↑":64,
  "↓":25,
  "←":7,
  "→":9,
  "R0":22,
  "R1":12,
  "R2":24,
  "R3":94,
  "R4":8,
  "R5":28,
  "R6":90,
  "R7":66,
  "R8":82,
  "R9":74};
var __irCodes = [];
for(var key in ircodes){
  __irCodes.push(ircodes[key]);
}
```

(continues on next page)

(continued from previous page)

```

}
var tones = {"B0":31,"C1":33,"D1":37,"E1":41,"F1":44,"G1":49,"A1":55,"B1":62,
             "C2":65,"D2":73,"E2":82,"F2":87,"G2":98,"A2":110,"B2":123,
             "C3":131,"D3":147,"E3":165,"F3":175,"G3":196,"A3":220,"B3":247,
             "C4":262,"D4":294,"E4":330,"F4":349,"G4":392,"A4":440,"B4":494,
             "C5":523,"D5":587,"E5":659,"F5":698,"G5":784,"A5":880,"B5":988,
             "C6":1047,"D6":1175,"E6":1319,"F6":1397,"G6":1568,"A6":1760,"B6":1976,
             "C7":2093,"D7":2349,"E7":2637,"F7":2794,"G7":3136,"A7":3520,"B7":3951,
             "C8":4186,"D8":4699};
var beats = {"Half":500,"Quater":250,"Eighth":125,"Whole":1000,"Double":2000,"Zero
→":0};

function onParse(byte){
    position = 0
    value = 0
    _buffer.push(byte);
    var len = _buffer.length;
    if(len>= 2){
        if (_buffer[len-1]==0x55 && _buffer[len-2]==0xff){
            _isParseStartIndex = len-2
            _isParseStart = true;
        }
        if (_buffer[len-1]==0xa && _buffer[len-2]==0xd && _isParseStart == true){
            _isParseStart = false;

            var position = _isParseStartIndex+2;
            var extId = _buffer[position];
            position+=1;
            var type = _buffer[position];
            position+=1;
            var value = 0;
            // 1 byte 2 float 3 short 4 len+string 5 double

            if (type == 1){
                value = _buffer[position];
            }
            if (type == 2){
                value = readFloat(position);
                if(value<-255 || value>1023){
                    value = 0;
                }
            }
            if (type == 3){
                value = readShort(position);
            }
            if (type == 4){
                value = readString(position);
            }
            if (type == 5){
                value = readDouble(position);
            }
            if(type<=5){
                if(value!=null){
                    _selectors["value_"+extId] = value;
                }
                _selectors["callback_"+extId](value);
            }
        }
    }
}

```

(continues on next page)

(continued from previous page)

```

        _buffer = []
    }
}
function readFloat(position){
    var buf = new ArrayBuffer(4);
    var intView = new Uint8Array(buf);
    var floatView = new Float32Array(buf);
    for(var i=0;i<4;i++){
        intView[i] = _buffer[position+i];
    }
    return floatView[0];
}
function readShort(position){
    var buf = new ArrayBuffer(2);
    var intView = new Uint8Array(buf);
    var shortView = new Int16Array(buf);
    for(var i=0;i<2;i++){
        intView[i] = _buffer[position+i];
    }
    return shortView[0];
}
function readString(position){
    var l = _buffer[position]
    position+=1
    s = ""
    for(var i=0;i<l;i++){
        s += self.buffer[position+i].charAt(0)
    }
    return s
}
function readDouble(position){
    var buf = new ArrayBuffer(8);
    var intView = new Uint8Array(buf);
    var doubleView = new Float64Array(buf);
    for(var i=0;i<8;i++){
        intView[i] = _buffer[position+i];
    }
    return doubleView[0];
}
function short2array(v){
    var buf = new ArrayBuffer(2);
    var intView = new Uint8Array(buf);
    var shortView = new Int16Array(buf);
    shortView[0] = v;
    return [intView[0],intView[1]];
}
function float2array(v){
    var buf = new ArrayBuffer(4);
    var intView = new Uint8Array(buf);
    var floatView = new Float32Array(buf);
    floatView[0] = v;
    return [intView[0],intView[1],intView[2],intView[3]];
}
function string2array(v){
    var arr = v.split("");
    for(var i=0;i<arr.length;i++){

```

(continues on next page)



(continued from previous page)

```

        arr[i] = arr[i].charCodeAt(0);
    }
    console.log(arr);
    return arr;
}
function deviceOpened(dev) {
    // if device fails to open, forget about it
    if (dev == null) device = null;

    // otherwise start polling
    poller = setInterval(function() {
        if(device!=null){
            function callback(buffer){
                var buf = new Uint8Array(buffer);
                var len = buf[0];
                if(buf[0]>0){
                    for(var i=0;i<len;i++){
                        onParse(buf[i+1]);
                    }
                }
            }
            device.read(callback,30);
        }
    }, 20);
};
var lastWritten = 0;
var _buffers = [];
var _isWaiting = false;
function addPackage(buffer,callback){
    _buffers.push(buffer);
    var extId = buffer[4];
    setTimeout(function(){
        callback(_selectors["value_"+extId]);
    },100);
    writePackage();
}
function writePackage(){
    if(_buffers.length>0&&_isWaiting==false){
        _isWaiting = true;
        var buffer = _buffers[0];
        _buffers.shift();
        device.write(buffer,function(){
            setTimeout(function(){
                _isWaiting = false;
                writePackage();
            },20);
        });
    }
}
ext._getStatus = function() {
    return status?{status: 2, msg: 'Ready'}:{status: 1, msg: 'Not Ready'};
};
ext._deviceConnected = function(dev) {
    if(device) return;
    console.log("_deviceConnected");
    device = dev;
    device.open(deviceOpened);

```

(continues on next page)

(continued from previous page)

```

        status = true;
    };
    ext._deviceRemoved = function(dev) {
        if(device != dev) return;
        if(poller) poller = clearInterval(poller);
        device = null;
        status = false;
    };
    ext._shutdown = function() {
        if(poller) poller = clearInterval(poller);
        if(device) device.close();
        device = null;
        status = false;
    }
    var arrayBufferFromArray = function(data){
        var result = new Int8Array(data.length);
        for(var i=0;i<data.length;i++){
            result[i] = data[i];
        }
        return result;
    }

    //***** mBot Blocks *****//
    function genNextID(port, slot){
        var nextID = port * 4 + slot;
        return nextID;
    }
    ext.resetAll = function(){
        var data = [0x5,0xff, 0x55, 0x02, 0x0, 0x04];
        addPackage(arrayBufferFromArray(data), function(){
        })
    };
    ext.runBot = function(lSpeed,rSpeed){
        var deviceId = 5;
        var extId = 0;
        var data = [extId, 0x02, deviceId].concat(short2array(-lSpeed)).
        ↪concat(short2array(rSpeed));
        data = [data.length+3, 0xff, 0x55, data.length].concat(data);
        addPackage(arrayBufferFromArray(data), function(){
        });
    }
    ext.runMotor = function(port,speed){
        if(typeof port=="string"){
            port = ports[port];
        }
        var deviceId = 10;
        var extId = 0;
        var data = [extId, 0x02, deviceId, port].concat(short2array(speed));
        data = [data.length+3, 0xff, 0x55, data.length].concat(data);
        addPackage(arrayBufferFromArray(data), function(){
        });
    }
    ext.runServo = function(port,slot,angle){
        if(typeof port=="string"){
            port = ports[port];
        }
        if(typeof slot=="string"){

```

(continues on next page)

(continued from previous page)

```

        slot = slots[slot];
    }
    var deviceId = 11;
    var extId = 0;
    var data = [extId, 0x02, deviceId, port, slot, angle];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    addPackage(arrayBufferFromArray(data), function(){
    });
}
ext.runLedOnBoard = function(index,red,green,blue){
    if(index == "all"){
        index = 0;
    }
    runLed(7,2,index,red,green,blue)
}
ext.runLed = function(port,slot,index,red,green,blue){
    if(typeof port == "string"){
        port = ports[port];
    }
    if(typeof slot == "string"){
        slot = slots[slot];
    }
    if(port==ports["on board"]){
        slot = 2;
    }
    if(index == "all"){
        index = 0;
    }
    var deviceId = 8;
    var extId = 0;
    var data = [extId, 0x02, deviceId, port, slot, index, red*1, green*1, blue*1];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    addPackage(arrayBufferFromArray(data), function(){
    });
}
ext.runBuzzer = function(tone,beat){
    if(typeof tone=="string"){
        tone = tones[tone];
    }
    if(typeof beat=="string"){
        beat = beats[beat];
    }
    var deviceId = 34;
    var extId = 0;
    var data = [extId, 0x02, deviceId].concat(short2array(tone)).
    →concat(short2array(beat));
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    addPackage(arrayBufferFromArray(data), function(){
    });
};
ext.stopBuzzer = function(){
    runBuzzer(0,0);
};
ext.showCharacters = function(port,x,y,msg){
    if(typeof port == "string"){
        port = ports[port];
    }
}

```

(continues on next page)

(continued from previous page)

```

    var deviceId = 41;
    var extId = 0;
    var brightness = 6;
    var data = [extId, 0x02, deviceId, port,1,brightness,3].
    ↪concat(short2array(x)).concat(short2array(7+y)).concat([msg.length].
    ↪concat(string2array(msg)));
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    addPackage(arrayBufferFromArray(data), function(){
    });
}
ext.showTime = function(port,hour,dot,min){
    if(typeof port == "string"){
        port = ports[port];
    }
    var deviceId = 41;
    var extId = 0;
    var brightness = 6;
    var data = [extId, 0x02, deviceId, port,3,brightness,dot==" ":"?1:0].
    ↪concat(short2array(hour)).concat(short2array(min));
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    addPackage(arrayBufferFromArray(data), function(){
    });
}
ext.runSevseg = function(port,num){
    if(typeof port == "string"){
        port = ports[port];
    }
    var deviceId = 9;
    var extId = 0;
    var data = [extId, 0x02, deviceId, port].concat(float2array(num));
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    addPackage(arrayBufferFromArray(data), function(){
    });
}
ext.runLightSensor = function(port,status){
    if(typeof port == "string"){
        port = ports[port];
    }
    if(typeof status == "string"){
        status = switchStatus[status];
    }
    var deviceId = 3;
    var extId = 0;
    var data = [extId, 0x02, deviceId, port,status];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    addPackage(arrayBufferFromArray(data), function(){
    });
}
ext.runShutter = function(port,shutter){
    if(typeof port == "string"){
        port = ports[port];
    }
    if(typeof shutter == "string"){
        shutter = shutterStatus[shutter];
    }
    var deviceId = 20;
    var extId = 0;

```

(continues on next page)

(continued from previous page)

```

    var data = [extId, 0x02, deviceId, port, shutter];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    addPackage(arrayBufferFromArray(data), function(){
    });
}
ext.getButtonOnBoard = function(status, callback){
    if(typeof status=="string"){
        status = buttonStatus[status];
    }
    var deviceId = 31;
    var port = 7;
    var extId = genNextID(port, 0);
    var data = [extId, 0x01, deviceId, port];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    _selectors["callback_"+extId] = function(v){
        callback(status==1?v>500:v<500);
    }
    addPackage(arrayBufferFromArray(data), _selectors["callback_"+extId]);
}
var _lastTime = 0;
var _lastButtonStatus = [false, false];
ext.whenButtonPressed = function(status, callback){
    if(typeof status == "string"){
        status = buttonStatus[status];
    }
    if(new Date().getTime()-_lastTime>150){
        _lastTime = new Date().getTime();
        var deviceId = 31;
        var port = 7;
        var extId = genNextID(port, status);
        var data = [extId, 0x01, deviceId, port];
        data = [data.length+3, 0xff, 0x55, data.length].concat(data);
        _selectors["callback_"+extId] = function(v){
            _lastButtonStatus[0] = status==1?v>500:v<500;
            _lastButtonStatus[1] = !_lastButtonStatus[status];
        }
        addPackage(arrayBufferFromArray(data), _selectors["callback_"+extId]);
    }
    return _lastButtonStatus[status];
}
}
ext.getLightSensor = function(port, callback){
    if(typeof port=="string"){
        port = ports[port];
    }
    var deviceId = 3;
    var extId = genNextID(port, 0);
    var data = [extId, 0x01, deviceId, port];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    _selectors["callback_"+extId] = callback;
    addPackage(arrayBufferFromArray(data), _selectors["callback_"+extId]);
}
ext.getUltrasonic = function(port, callback){
    if(typeof port=="string"){
        port = ports[port];
    }
    var deviceId = 1;

```

(continues on next page)

(continued from previous page)

```

    var extId = genNextID(port,0);
    var data = [extId, 0x01, deviceId, port];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    _selectors["callback_"+extId] = function(v) {
        callback(Math.floor(v*100.0)/100.0);
    }
    addPackage(arrayBufferFromArray(data), _selectors["callback_"+extId]);
}
ext.getLinefollower = function(port,callback){
    if(typeof port=="string"){
        port = ports[port];
    }
    var deviceId = 17;
    var extId = genNextID(port,0);
    var data = [extId, 0x01, deviceId, port];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    _selectors["callback_"+extId] = callback;
    addPackage(arrayBufferFromArray(data), _selectors["callback_"+extId]);
}
ext.getJoystick = function(port,ax,callback){
    if(typeof port=="string"){
        port = ports[port];
    }
    if(typeof ax=="string"){
        ax = axis[ax];
    }
    var deviceId = 5;
    var extId = genNextID(port,ax);
    var data = [extId, 0x01, deviceId, port, ax];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    _selectors["callback_"+extId] = callback;
    addPackage(arrayBufferFromArray(data), _selectors["callback_"+extId]);
}
ext.getPotentiometer = function(port,callback){
    if(typeof port=="string"){
        port = ports[port];
    }
    var deviceId = 4;
    var extId = genNextID(port,0);
    var data = [extId, 0x01, deviceId, port];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    _selectors["callback_"+extId] = callback;
    addPackage(arrayBufferFromArray(data), _selectors["callback_"+extId]);
}
ext.getSoundSensor = function(port,callback){
    if(typeof port=="string"){
        port = ports[port];
    }
    var deviceId = 7;
    var extId = genNextID(port,0);
    var data = [extId, 0x01, deviceId, port];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    _selectors["callback_"+extId] = callback;
    addPackage(arrayBufferFromArray(data), _selectors["callback_"+extId]);
}
ext.getLimitswitch = function(port,slot,callback){
    if(typeof port=="string"){

```

(continues on next page)

(continued from previous page)

```

        port = ports[port];
    }
    if(typeof slot=="string"){
        slot = slots[slot];
    }
    var deviceId = 21;
    var extId = genNextID(port,slot);
    var data = [extId, 0x01, deviceId, port, slot];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    _selectors["callback_"+extId] = callback;
    addPackage(arrayBufferFromArray(data), _selectors["callback_"+extId]);
}
ext.getTemperature = function(port,callback){
    if(typeof port=="string"){
        port = ports[port];
    }
    var deviceId = 2;
    var extId = genNextID(port,0);
    var data = [extId, 0x01, deviceId, port];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    _selectors["callback_"+extId] = function(v){
        callback(Math.floor(v*100)/100);
    }
    addPackage(arrayBufferFromArray(data), _selectors["callback_"+extId]);
}
ext.getPirmotion = function(port,callback){
    if(typeof port=="string"){
        port = ports[port];
    }
    var deviceId = 6;
    var extId = genNextID(port,0);
    var data = [extId, 0x01, deviceId, port];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    _selectors["callback_"+extId] = callback;
    addPackage(arrayBufferFromArray(data), _selectors["callback_"+extId]);
}
ext.getIrRemote = function(code,callback){
    var deviceId = 14;
    if(typeof code=="string"){
        code = ircodes[code];
    }
    var port = 11;
    var slot = __irCodes.indexOf(code);
    var halfSize = __irCodes.length >> 1;
    if(slot >= halfSize){
        ++port;
        slot -= halfSize;
    }
    var extId = genNextID(port,slot);
    var data = [extId, 0x01, deviceId, 0, code];
    data = [data.length+3, 0xff, 0x55, data.length].concat(data);
    _selectors["callback_"+extId] = callback;
    addPackage(arrayBufferFromArray(data), _selectors["callback_"+extId]);
}
var descriptor = {
    blocks: [
        [" ", "move left %d.motorvalue right %d.motorvalue", "runBot", 100, 100],

```

(continues on next page)

(continued from previous page)

```

        [" ", "set motor%d.motorPort speed %d.motorvalue","runMotor", "M1", 0],
        [" ", "set servo %d.port %d.slot angle %d.servovalue","runServo", "Port1",
↪ "Slot1", 90],
        [" ", "set led %d.lport %d.slot %d.index red%d.value green%d.value blue%d.
↪ value","runLed","on board","Slot1","all",0,0,0],
        [" ", "play tone on note %d.note beat %d.beats","runBuzzer", "C4", "Half
↪ "],
        [" ", "show face %d.port x:%n y:%n characters:%s","showCharacters", "Port1
↪ ", 0,0,"Hello"],
        [" ", "show time %d.port hour:%n %m.points min:%n","showTime", "Port1",
↪ 10,":",20],
        [" ", "show drawing %d.port x:%n y:%n draw:%m.drawFace","showDraw", "Port1
↪ ", 0,0,"      "],
        ["-"],
        [" ", "set 7-segments display%d.port number %n","runSevseg", "Port1",
↪ 100],
        [" ", "set light sensor %d.aport led as %d.switchStatus","runLightSensor",
↪ "Port3", "On"],
        [" ", "set camera shutter %d.port as %d.shutter","runShutter","Port1",
↪ "Press"],
        ["-"],
        ["h", "when button %m.buttonStatus","whenButtonPressed","pressed"],
        ["R", "button %m.buttonStatus","getButtonOnBoard","pressed"],
        ["R", "light sensor %d.laport","getLightSensor","light sensor on board"],
        ["-"],
        ["R", "ultrasonic sensor %d.port distance","getUltrasonic","Port1"],
        ["R", "line follower %d.port","getLinefollower","Port1"],
        ["R", "joystick %d.aport %d.Axis","getJoystick","Port3","X-Axis"],
        ["R", "potentiometer %d.aport","getPotentiometer","Port3"],
        ["R", "sound sensor %d.aport","getSoundSensor","Port3"],
        ["R", "limit switch %d.port %d.slot","getLimitswitch","Port1","Slot1"],
        ["R", "temperature %d.port %d.slot °C","getTemperature","Port3","Slot1"],
        ["R", "pir motion sensor %d.port","getPirmotion","Port2"],
        ["-"],
        ["R", "ir remote %m.ircode pressed","getIrRemote","A"],
        ["-"],
        [" ", "send mBot's message %s","runIR", "hello"],
        ["R", "mBot's message received","getIR"],
        ["-"],
        ["R", "timer","getTimer", "0"],
        [" ", "reset timer","resetTimer", "0"]
    ],
    menus: {
        motorPort:["M1","M2"],
        slot:["Slot1","Slot2"],
        index:["all",1,2],
        Axis:["X-Axis","Y-Axis"],
        port:["Port1","Port2","Port3","Port4"],
        aport:["Port3","Port4"],
        lport:["led on board","Port1","Port2","Port3","Port4"],
        laport:["light sensor on board","Port3","Port4"],
        direction:["run forward","run backward","turn right","turn left"],
        points:[":", " "],
        note:["C2","D2","E2","F2","G2","A2","B2","C3","D3","E3","F3","G3","A3","B3
↪ ", "C4","D4","E4","F4","G4","A4","B4","C5","D5","E5","F5","G5","A5","B5","C6","D6",
↪ "E6","F6","G6","A6","B6","C7","D7","E7","F7","G7","A7","B7","C8","D8"],
        beats:["Half","Quater","Eighth","Whole","Double","Zero"],

```

(continues on next page)



(continued from previous page)

```

servovalue:[0,45,90,135,180],
motorvalue:[-255,-100,-50,0,50,100,255],
value:[0,20,60,150,255],
buttonStatus:["pressed","released"],
shutter:["Press","Release","Focus On","Focus Off"],
switchStatus:["Off","On"],
ircode:["A","B","C","D","E","F","↑","↓","←","→","Setting","R0","R1","R2",
↪ "R3","R4","R5","R6","R7","R8","R9"],
    }
};
var hid_info = {type: 'hid', vendor: 0x0416, product: 0xffff};
ScratchExtensions.register('Makeblock mBot', descriptor, ext, hid_info);
})();

```

## 2.5 chapter 4: johy-five

Open the Arduino IDE, select: File > Examples > Firmata > StandardFirmataPlus

StandardFirmataPlus is available in Firmata v2.5.0 or greater

\*LED

node eg/led-blink.js

\*motor

node eg/motor-directional.js

98:D3:32:11:1B:EA HC-06

## 2.6 chapter 5: Arduino Uno

D:SEANLAB2minitankAndroidArduino-Bluetooth-BasicArduino\_Bluetooth\_Basic

OK

BT Serial OK

D:SEANLAB2ArduinoBTSoftSerial\_OKarduino\_BT\_Serial\_OK

Serial OK BT Serial OK BT PIN 10 : Tx (Key HC-06 )

11 :Rx (Key HC-06 Tx)

1. D:SEANLAB2Arduino2.4TFT\_LCD

lcd & touch OK

2. D:SEANLAB2Arduino4.RFID\_OK

[http://wiki.keyestudio.com/index.php/Ks0205\\_Keyestudio\\_RC522\\_Sensor](http://wiki.keyestudio.com/index.php/Ks0205_Keyestudio_RC522_Sensor)

3.2 line lcd -OK D:SEANLAB2Arduino3.2line\_LCD\_OK <http://www.instructables.com/id/How-to-Connect-I2C-Lcd-Display-to-Arduino-Uno/>

zip lib - - zip VCC - 5V GND - GND SDA - A4 SCL -A5

4.RFID - OK

constrast

5. stepper motor -BYJ48 -OK

<http://www.instructables.com/id/BYJ48-Stepper-Motor/>

D:SEANLAB2Arduino5.Stepper\_motor\_OK0.sketch\_mar08b\_stepper\_motor\_ok

6. IR Recive <http://www.instructables.com/id/Arduino-Infrared-Remote-tutorial/> D:SEANLAB2Arduino6.IR\_Recive6.sketch\_mar08a\_I

7. DS1302\_RTC\_LCD —RTC sync fail

\*LCD VCC - 5V GND - GND SDA - A4 SCL - A5

\*RTC Arduino Uno D2 → DS1302 RTC RST

Arduino Uno D3 → DS1302 RTC DAT

Arduino Uno D4 → DS1302 RTC CLK

Arduino Uno D5 → DS1302 RTC GND

Arduino Uno D6 → DS1302 RTC VCC

8. 5461 LED – OK <http://docs.whiteat.com/?p=2078>

9. 05611 LED – XX <https://www.hacktronics.com/Tutorials/arduino-and-7-segment-led.html>

10. 1588 AS <https://create.arduino.cc/projecthub/SAwandter1/programming-8x8-led-matrix-23475a>

## 2.7 chapter 6: mblock

- windows

1. visual studio community 2017 install 2. windows- dev- dev command line 3. >npm install

npm install --production windows-build-tools

VC community 2017 install & C++ check

electron install >npm install electron npm install electron --save-dev --save-exact

D:SEANLAB2mBlock2 ==>> OK npm install npm run rebuild-serialport npm run rebuild-hid npm run rebuild-bluetooth npm start

mblock ==>> OK D:SEANLAB2mBlock2Firmwarebaseboard\_firmware\_uno\_minitanak ==>Motor OK

hex file D:SEANLAB2mBlock2mBlocktoolshexuno.hex down

connect -com12-

board -uno

LED OK

mBlock OK

protocol

6-9 port

mblock Bluetooth ==> OK Bluetooth - Serial 8 ==>OK ( BT )

BTSerial.write("AT+PIN1234");

```
apt-get install libbluetooth-dev
```

```
sudo vim /usr/lib/systemd/system/bluetooth.service
```

and adding the `--compat` flag to the `ExecStart` value:

```
ExecStart=/usr/lib/bluetooth/bluetoothd--compat
```

Finally, restart the service:

```
systemctl daemon-reload systemctl restart bluetooth
```

```
sudo apt-get install libusb-1.0-0-dev
```

```
npm install electron
```

```
x-11 proxy mcookie c29ee074c51368664c164e7c5c0a747d xauth add localhost/unix:14 MIT-MAGIC-COOKIE-1
c29ee074c51368664c164e7c5c0a747d 08e4af70e0fcf25e927ff7683ab9ddb3
```

```
xauth add localhost/unix:14 MIT-MAGIC-COOKIE-1 08e4af70e0fcf25e927ff7683ab9ddb3
```

On my client PC I'm running Windows 10 and I have installed Xming and Xming Fonts.

I am running Putty and I have configured it to enable X11 Forwarding. I have tried leaving the X display location blank (as default) and also with the value `:0.0`.

On my server I'm running Ubuntu on AWS. I edited the `/etc/ssh/ssh_config` file to include the following two lines:

```
ForwardX11 yes ForwardX11Trusted yes
```

I also installed xauth using `sudo apt-get install xauth`

Whenever I try to run an application, like `xterm` & or `xclock` I get the same error:

```
PuTTY X11 proxy: Unsupported authorisation protocol Error: Can't open display: localhost:10.0
```

```
windowdown mblock
```

```
npm install --global --production windows-build-tools
```

## 2.8 chapter 7: code.org

```
sudo apt-get update sudo apt-get install -y git mysql-server mysql-client libmysqlclient-dev libxslt1-
dev libssl-dev zlib1g-dev imagemagick libmagickcore-dev libmagickwand-dev openjdk-9-jre-headless
libcairo2-dev libjpeg8-dev libpango1.0-dev libgif-dev curl pdftk enscript libsqlite3-dev phantomjs build-
essential redis-server rbenv ruby-build npm ruby2.3-dev
```

Hit enter and select default options for any configuration popups, leaving mysql passwords blank

**Install Node and Nodejs** Type `curl -sL https://deb.nodesource.com/setup_6.x | sudo -E bash -` And then `sudo apt-get install -y nodejs`

**Install Ruby 2.2.3 with rbenv** `rbenv install 2.2.3 rbenv global 2.2.3 rbenv rehash`

**Install yarn** First, type `curl -sS https://dl.yarnpkg.com/debian/pubkey.gpg | sudo apt-key add -` Then `echo "deb https://dl.yarnpkg.com/debian/ stable main" | sudo tee /etc/apt/sources.list.d/yarn.list` And lastly, `sudo apt-get update && sudo apt-get install yarn=0.23.2-1`

**Finally, configure your mysql to allow for a proper installation. You may run into errors if you did not leave mysql passwords blank**

```
Type echo "ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY
'';" | sudo mysql
```

```
echo "ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'king0733';" | sudo mysql
>mysql -uroot -p
```

```
>ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY ""; >FLUSH PRIVILEGES;
```

**Read the following notes, then go back up to the overview and run the commands there.** If, for any reason, you are forced to interrupt the rake install command before it completes, cd into dashboard and run `bundle exec rake db:drop` before trying rake install again rake install must always be called from the local project's root directory, or it won't work. Finally, don't worry if your versions don't match the versions in the overview if you're following this method; the installation should still work properly regardless

```
GRANT ALL PRIVILEGES ON . TO 'root'@'%' IDENTIFIED BY 'king0733' WITH GRANT OPTION;
```

```
FLUSH PRIVILEGES;
```

**1.error** ruby

mkmf.rb can't find header files for ruby at /usr/lib/ruby/include/ruby.h

```
==> sudo apt-get install ruby-dev
```

```
sudo apt-get update sudo apt-get build-essential
```

1.1 zlib is missing; necessary for building libxml2 `sudo apt-get install zlib1g-dev`

1.2 Can't install RMagick 2.15.4. Can't find Magick-config or pkg-config `sudo apt-get install libmagickwand-dev imagemagick`

1.3 Installing sqlite3 1.3.11 with native extensions Gem::Ext::BuildError: ERROR: Failed to build gem native extension.

```
apt-get install libsqlite3-dev
```

```
start
```

## 2.9 chapter 8: AVR GCC

```
sudo apt-get install arduino-mk
```

**\*install oracle jdk** `sudo add-apt-repository ppa:webupd8team/java sudo apt-get update`

```
sudo apt-get install oracle-java8-installer
```

```
export ARDUINO_DIR=/home/sean/Arduino/arduino-1.8.5 export ARDMK_DIR=/home/sean/Arduino/Arduino-
Makefile export AVR_TOOLS_DIR=/usr
```

## 2.10 chapter 9: Veyon

**1.installll** Host : install service & master

```
- ip mac
```

**2.master** - MS () - Master

3. master config

```
-> IP,Mac
```

raspberry pi

```
make -j4 make install veyon-server
```

## 2.11 chapter 10: blockly

D:SEANLAB2Blocklyseanlabarduino

python arduino\_web\_server.py

<https://github.com/google/blockly/wiki>

basic concept

D:SEANLAB2Blocklynode-blockly

node app.js

Disable arduino loop with once

C:Program Files (x86)Arduinohardwarearduinoavrcoresarduino main.cpp

```
// For Once execution for (int i=0;i<1;i++) {
    loop(); if (serialEventRun) serialEventRun();
}
/*
    for (;;) { loop(); if (serialEventRun) serialEventRun();
    }
*/
```

## 2.12 chapter 11: Ware house

1. python pip install

python path scripts

D:SEANLAB2IoTaWareHouseaWareHouseServer

pip install -r requirements.txt

2. install influxdb

ubuntu wget [https://dl.influxdata.com/influxdb/releases/influxdb\\_1.4.3\\_amd64.deb](https://dl.influxdata.com/influxdb/releases/influxdb_1.4.3_amd64.deb) sudo dpkg -i influxdb\_1.4.3\_amd64.deb

influxd -config /etc/influxdb/influxdb.conf

\$ sudo apt-get install python-pip python-dev build-essential \$ sudo pip install --upgrade pip \$ sudo pip install --upgrade virtualenv

veyon

X displays are protected by a “key” which you need to give in order to be able to connect. That key will generally be stored in the user’s ~/.Xauthority file.

To allow someone else’s application (like x11vnc) to connect to his DISPLAY, a user has to give him that key or grant him access to his ~/.Xauthority file. You can retrieve the key of your display by doing.

xauth list “\$DISPLAY”

You can grant access to your ~/.Xauthority by changing its permissions (group ownership or ACLs).

The other user can specify which authority file to use via the XAUTHORITY environment variable, or add a key to a specific display to his own ~/.Xauthority using xauth add.

## 2.13 chapter 12: Docker

\*ubuntu14.04 1.remove old version sudo apt-get remove docker docker-engine docker.io

2. install

**sudo apt-get install** apt-transport-https ca-certificates curl software-properties-common

curl -fsSL <https://download.docker.com/linux/ubuntu/gpg> | sudo apt-key add -

sudo apt-key fingerprint 0EBFCD88

sudo apt-get update

sudo apt-get install docker-ce

docker run -d -p 8083:8083 -p 8086:8086 -e ADMIN\_USER="root" -e INFLUXDB\_INIT\_PWD="root" -e PRE\_CREATE\_DB="warehouse" tutum/influxdb:latest docker run -d -p 8083:8083 -p 8086:8086 -e ADMIN\_USER="admin" -e INFLUXDB\_INIT\_PWD="admin" -e PRE\_CREATE\_DB="warehouse" tutum/influxdb:latest

docker run -it -d -p 8061:8061 tislamo/blockpy

##docker cloud9 docker run -it -d -p 8071:8071 -v /home/sean/docker\_workspace:/workspace/ kdelfour/cloud9-docker

docker run -it -d -p 80:80 -v /home/sean/docker\_workspace:/workspace/ kdelfour/cloud9-docker

sudo ufw allow 8071/tcp

sudo cat /proc/sys/net/ipv6/conf/all/disable\_ipv6

sudo systemctl -p

docker attach mynodered To stop the container:

docker stop mynodered To start the container:

docker start mynodered

## 2.14 chapter 13: Microbit

<https://github.com/Microsoft/pxt-microbit>

git clone <https://github.com/microsoft/pxt-microbit> cd pxt-microbit npm install -g pxt

npm install

pxt serve

Edublock

<https://microbit.edublocks.org/>

scratch+microbit

<https://github.com/MrYsLab/s2m>

>s2m

microbit offline

<https://www.microsoft.com/ko-kr/p/makecode-for-micro-bit/9pjc7sv48lcx?ocid=badge&rtc=1>

## 2.15 chapter 14: Minecraft

SEAN3.Note2.minecraft

#rasberry pi +minecraft java <https://www.raspberrypi.org/forums/viewtopic.php?t=186547>

D:SEANLABPYTHONPython3.7Libsite-packagesmcpiminecraft.py

/usr/lib/python3/dist-packages

////raspberrypi

/usr/local/lib/python3.5/dist-packages/mcpi

## 2.16 chapter 15: Ardublock

<https://github.com/taweili/ardublock>

## 2.17 chapter 16: Raspberry Pi

#Always force HDMI output and enable HDMI sound

hdmi\_force\_hotplug=1

hdmi\_drive=2

copy /home/pi/.Xauthority /root/.Xauthority

<https://pimylifeup.com/raspberry-pi-minecraft-server/>

minecraft server

wget <https://hub.spigotmc.org/jenkins/job/BuildTools/lastSuccessfulBuild/artifact/target/BuildTools.jar>

sudo java -jar BuildTools.jar -rev 1.12.2

sudo java -Xms512M -Xmx1008M -jar /home/minecraft/spigotlatestbuild.jar nogui

sudo java -Xms512M -Xmx1008M -jar /home/pi/minecraftserver/spigot-1.12.2.jar nogui

sudo nano eula.txt In here change false to TRUE, once done save and exit by pressing ctrl+x then y.

## 2.18 chapter 17: Minitank

1. D:SEANLAB3mblock1.mblock\_minitank2

<4 wheel> keystudion original D:SEANLAB3mblock1.1.mblock\_minitankArduinobaseboard\_firmware\_uno\_minitank\_BT\_Serial\_OK

<3 wheel> D:SEANLAB3mblock1.mblock\_minitank2Arduino4.baseboard\_firmware\_uno\_minitank3\_BT\_Serial\_OKbaseboard\_firmw  
==> OK

BT-05 — Serial port 8 mblock - bluetooth ->OK

motor joystic led

## 2.19 chapter 18: MakeyMakey

<http://learn.linksprite.com/arduino/use-makey-makey-style-touch-usb-shield-as-a-keyboard/>

If you have a compile error of library examples, try with an older Arduino IDE version like 1.05

<https://github.com/cefaloide/Arduino-Makey-Touch-Key-USB-SHIELD>

IDE 1.05 Compile —> OK

D:SEANLAB2MakeyMakey\_0\_Touch\_Key2\_OK ==> OK

A4 - space A5 - Enter

**HID** 1 - BT keyboard 2 - Makey keyboard

## 2.20 chapter 19: WAVGAT Uno

<https://forum.arduino.cc/index.php?topic=540421.0>

I also had the same issue at first but I have found the solution...

Step 1

Download the following RAR file from the link below; <https://drive.google.com/open?id=10gwrG9uTDwaEO-7EudsmBkfgdcyrcABI>

Step 2

Once downloaded, extract these and copy the content within the “update” folder (not the “update” folder itself just the content inside it)

Step 3

Paste content to the following path on your computer... C:\Users\Administrator\Documents\Arduino

Step 4

Restart Arduino IDE Software

Open Arduino IDE, select your COM port if required and Select your WAVGAT board type (WAVGAT boards are at the very bottom of the board list)

## 2.21 chapter 20: Cloud9

`docker run -it -d -p 8071:8071 -v /home/pi/cloud9/workspace/:/workspace/ kdelfour/cloud9-docker`

<https://calyfactory.github.io/docke%EC%97%90%EB%8C%80%ED%95%B4-%EC%95%8C%EC%95%84%EB%B3%B4%EC%9E%90/>



## CHAPTER 3

---

Part 3: 3

---



## CHAPTER 4

---

Part 4: 4

---



## CHAPTER 5

---

Part 5: 5

---