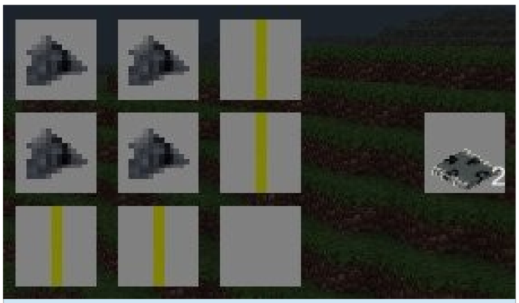
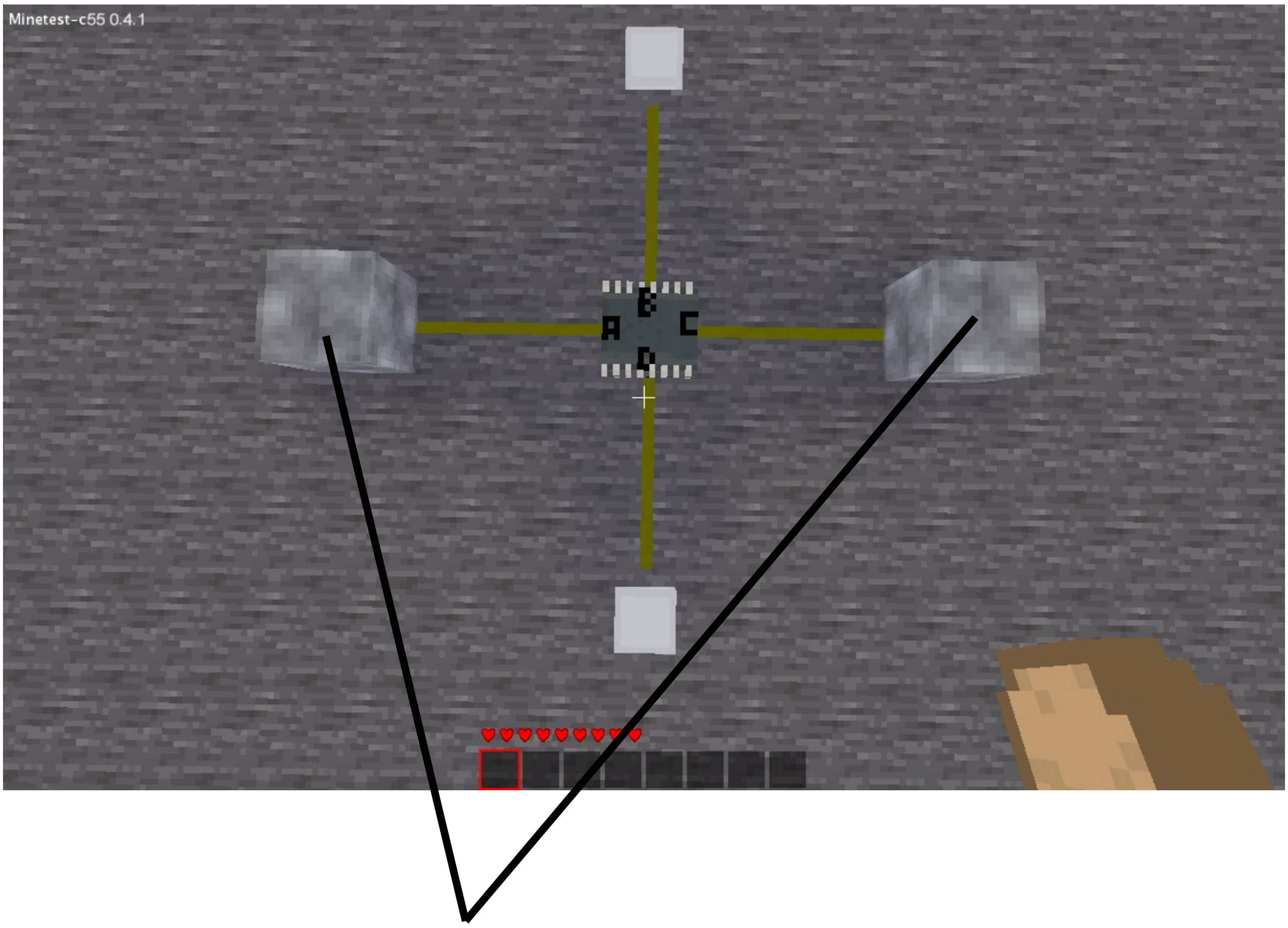
Mesecon Microcontroller coding

in 1 Hour (or two)



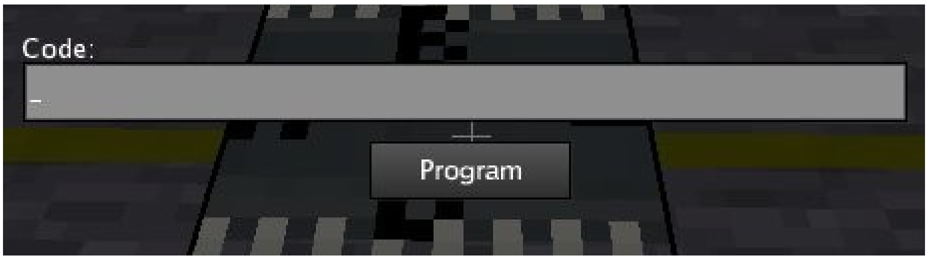
Step 1: Craft the Microcontroller

Step 2: Build this circuit:



Mesecon Switch

Step 3: Right-click on the microcontroller



Step 4: Learn coding it!

Basic information:

1) The Microcontroller has 4 Input/Output ports: A, B, C, and D
2) The Microcontroller runs at a speed of 2Hz
3) You can only code it by pressing the „Program“ button, not by closing the window with the esc-Key
4) Every time the Program button is hit, the microcontroller performs a reset, which means it turns off all the Ports
5) Don't mix lowercase and uppercase letters in your code!
6) When hovering on the microcontroller with your mouse it shows its state:
    - „Unprogrammed Microcontroller“ → It was just placed
    - „Code not valid“ → There is a bug in your code
    - „Programmed Microcontroller“ → The code works fine

Basic functions:

There are 2 basic functions that the microcontroller has:
`on(port1, port2, ...)` and `off(port1, port2, ...)`

You may start by entering

```
on(B)
```

into the Code field. After clicking „Program“ the lamp at Port B should be turned on.

You can also pass multiple parameters to these functions:

```
on(B, D)
```

will turn on both lamps.

Concatenation of functions:

You can simply put one function after the other.
For example

```
on(B, D)off(B)
```

will only turn on Port D, Port B will stay turned off.
Port B also won't quickly light up!
The code is first parsed, the port states are set afterwards.

Using the if-clause

If cause means that the parser/compiler checks whether a given condition applies or not. If the condition applies the functions between if and ; are executed, after that or if the condition does not apply the parser continues after ;

The structure for the if clause is as follows:

```
if(condition)functionA()functionB()...;functionC()
```

The condition

A condition applies if its value is 1, it does not apply if its value is 0. Any other value creates a „not valid“ error.

The letters A, B, C and D stand for the status of the corresponding „minetest-physical“-Ports.

Example:

```
if(A) on(B);
```

If you now turn on the switch connected to Port A, the microcontroller will turn on Port B. One thing you may discover: It does not turn off Port B after having turned it on.

The solution:

```
off(B) if(A) on(B);
```

You may notice that it takes half a second until the Microcontroller activates Port B: That's because the s in „Microcontroller“ stands for speed (ok, that's stupid and not the real reason, so simply accept it).

Not

The code from „The condition“ was pretty much boring, so now comes an inverter:

```
off(B) if(!A) on(B);
```

The ! Stands for not. „If Port A is not on then turn port B on“.

And

Let's make an and gate:

```
off(B) if(A&C) on(B);
```

The & stands for and.

Or

That's pretty much useless.

```
off(B) if(A|C) on(B);
```

The | stands for or.

Xor

Xor = exclusive or. Either A or B (but not both).

```
off(B) if(A~C) on(B);
```

The ~ stands for xor.

Equals (=)

```
off(B) if(A=C) on(B);
```

Port B will be turned on if Port A has the same state as Port B. (Either both off or both on)

More complicated code:

These are just some examples to see what is possible:

```
off(B)off(D)if(A~C)on(B);if(A&C)on(D);
```

This is an AND gate and a XOR gate in one:
Inputs for both are A and C, but the outputs are different:
B for XOR-Gate and D for AND-Gate.

```
off(B)if(A|B|C)on(B);if(A=B=C)off(B);
```

>> What does this code do?

Find it out and make your own creations!

For questions and mistakes in this tutorial (most is untested) feel free to eMail me:
norrepli [at] gmail.com or ask a question in the forum.

Jeija

MeseconMicroTut 0.1