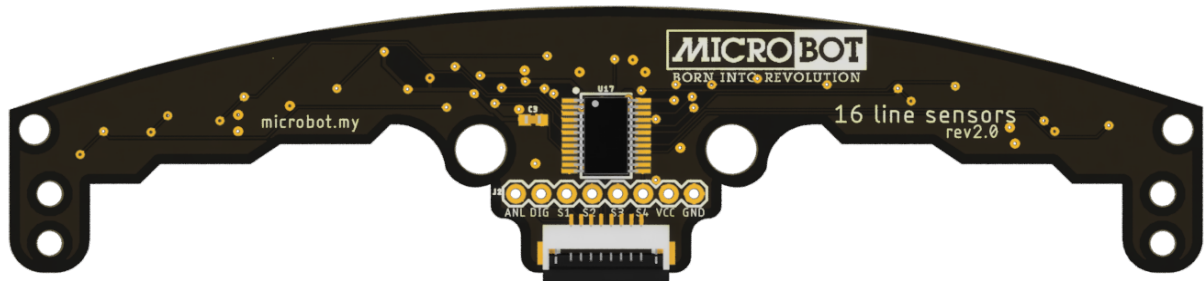


16 LINE SENSORS

THE “16 LINE SENSORS” is an array of sensors that can be read digitally or analog way. For this purpose, an analog multiplexer is used with which the specific sensor to be read is determined.

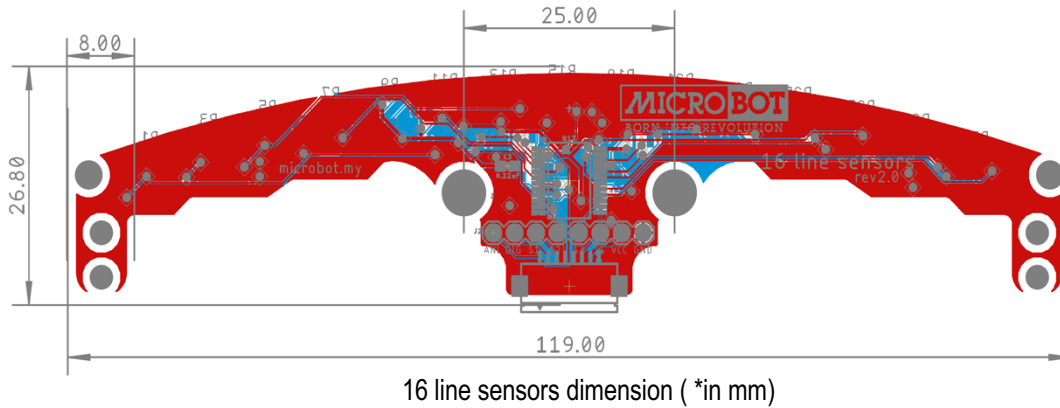


Pins	Description
VCC	3.3 to 5V
GND	0V reference
ANL	0 to Vcc Analog Output
DIG	Digital I/O
S1	Multiplexer output
S2	
S3	
S4	

Pins definition

S1	S2	S3	S4	sensors
0	0	0	0	15
1	0	0	0	14
0	1	0	0	13
1	1	0	0	12
0	0	1	0	11
1	0	1	0	10
0	1	1	0	9
1	1	1	0	8
0	0	0	1	7
1	0	0	1	6
0	1	0	1	5
1	1	0	1	4
0	0	1	1	3
1	0	1	1	2
0	1	1	1	1
1	1	1	1	0

Multiplexer logic table



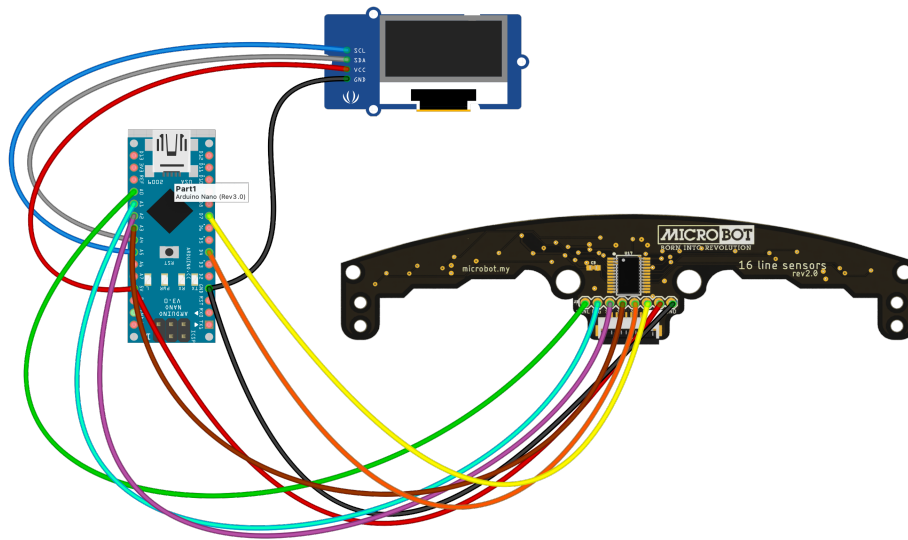
Characteristic	Value
Weight	5g
Current consumption at 5V	175mA
Current consumption at 3.3V	80mA

How to use 16 line sensors module?

To carry out the test of the module, it will be necessary to have the following elements at your disposal:

- > Arduino nano
- > OLED I2C Display
- > 16 line sensors module
- > Jumper wires.

Having this ready, proceed to carry out the assembly as shown in the following diagram.



Wiring diagramme

The operational voltage for this module must be within 3.3V to 5V, as shown in the previous diagram, in this way, the digital output voltage through the DIG pin will be the same as the supply voltage while the analog output (ANL) will have only values between 0V to VCC.

The *multiplexer*, shortened to “MUX” or “MPX”, is a combinational logic circuit designed to switch one of several input lines through to a single common output line by the application of a control signal. Multiplexers operate like very fast acting multiple position rotary switches connecting or controlling multiple input lines called “channels” one at a time to the output.

For this purpose, the sensor must be selected using pins (S1, S2, S3, S4), bearing in mind that only digital values of 1 (HIGH) or 0 (LOW) are recognized and in this way the address is constructed by setting at level 1 or level 0 as appropriate in the truth table. That is, if you want to read the analog value and the digital value of sensor 15, you must set 1=LOW, 2=LOW, 3=LOW, 4=LOW and for sensor 14 it will be 1=HIGH, 2=LOW, 3=LOW, 4=LOW, and so on.

In this way, to make programming easier, a function is created with the pins as an arrangement, in this way the number 0, 1, 2 ... etc will be sent as appropriate and the function will take care of putting the respective pins high or low.