M1-bit robot

Notes i made during the build of this Robot, based on Motorola MC14500B 1 bits microprocessor with some dutch tekst :-(

Hardware I/O

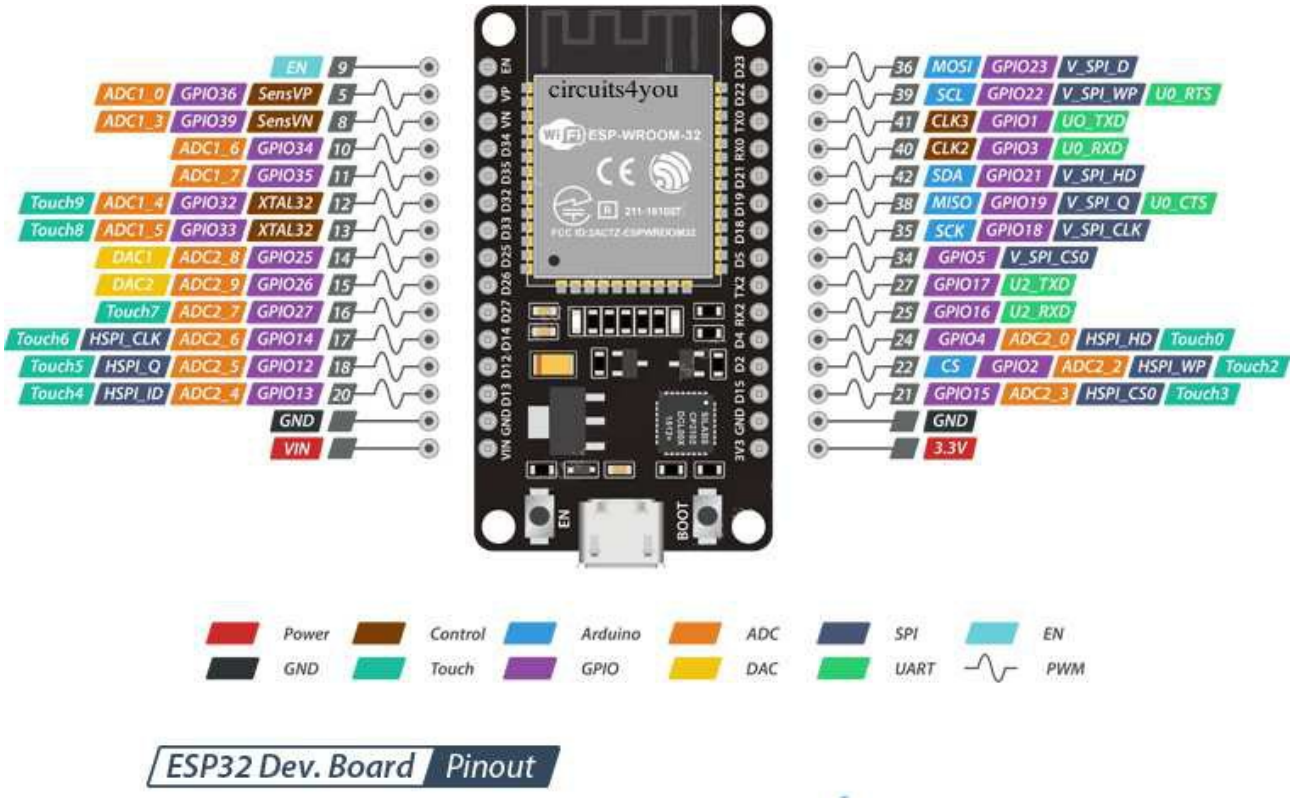
UIT:	Functie	Signaal
0	Motor_Left	High is Motor ON
1	Motor_Right	High is Motor ON
2	(Test Led_1)	Not connected
3	(Test Led_2)	Not connected
4	M1	Memory bit to IN 4
5	M2	Memory bit to IN 5
6	Timer OUT	Timer to IN 7
7		

IN	Functie	Signaal
0	Line Sensor Links	High is black line / Low is white line
1	Line Sensor Rechts	High is black line / Low is white line
2	Vaste Plus	Always high
3		
4	M1	Memory bit from OUT 4
5	M2	Memory bit from OUT 5
6	Timer IN	Timer from OUT 7
7	RR	Result Register

ESP32 Bord M1-bit:

ESP-WROOM-32		GPIO:	Functie:	Opm:
links	rechts	d0	BOOT - Pin	
En reset	d23	1 tx0		
d36 vp	d22	2	Niet Gebruiken.	Defect ??
d39 vn	1 tx0	3 rx0		
d34	3 rx0	4	Uitgang naar Timer_Test	Tijdenlijk !!!!
d35	d21	5	Niet Gebruiken.	Defect ??
d32	d19	6 F clk		NIET gebruiken
d33	d18	7 F do		NIET gebruiken
d25	d5	8 F d1		NIET gebruiken
d26	17 tx2	9 F d2		NIET gebruiken
d27	16 rx2	10 F d3		NIET gebruiken
d14	d4	11 F cmd		NIET gebruiken
d12	d2	12	EPROM - D0	Adresbus - 0
d13	d15	13	EPROM - D1	Adresbus - 1
sd2	d0	14	EPROM - D2	Adresbus - 2
cmd	sd0	16 rx2	EPROM - D4	Instructionbus - 0
gnd	clk	17 tx2	EPROM - D5	Instructionbus - 1
vin	3V3	18	EPROM - D6	Instructionbus - 2
		19	EPROM - D7Instructionbus	Instructionbus - 3
		21 sda	OLED LCD	
		22 scl	OLED LCD	
		23	Clock - IN vanuit MC14500B	LET OP !! via R 180 Ohm
		25 DAC		
		26 DAC		
		27	Flag-F IN vanuit MC14500B	LET OP !! via R 180 Ohm
		32	Reset - UIT naar MC14500B	
		33	Ws2812 LED	
		34		Alleen ingang
		35		Alleen ingang
		36		Alleen ingang
		39		Alleen ingang

The sign below corresponds to my sign:




```
FastLED.addLeds<NEOPIXEL, DATA_PIN>(leds, NUM_LEDS);  // GRB ordering is assumed
FastLED.addLeds<WS2812, DATA_PIN, RGB>(leds, NUM_LEDS);  // GRB ordering is
// FastLED.addLeds<WS2812, DATA_PIN, GRB>(leds, NUM_LEDS);  // GRB ordering is orgineel !!
FastLED.setBrightness(24);
```

https://randomnerdtutorials.com/esp32-pinout-reference-gpios/

GPIO	Input	Output	Notes
0	pulled up	OK	outputs PWM signal at boot
1	TX pin	OK	debug output at boot
2	OK	OK	connected to on-board LED
3	OK	RX pin	HIGH at boot
4	OK	OK	
5	OK	OK	outputs PWM signal at boot
6	x	x	connected to the integrated SPI flash
7	x	x	connected to the integrated SPI flash
8	x	x	connected to the integrated SPI flash
9	x	x	connected to the integrated SPI flash
10	x	x	connected to the integrated SPI flash
11	x	x	connected to the integrated SPI flash
12	OK	OK	boot fail if pulled high
13	OK	OK	
14	OK	OK	outputs PWM signal at boot
15	OK	OK	outputs PWM signal at boot
16	OK	OK	
17	OK	OK	
18	OK	OK	
19	OK	OK	
21	OK	OK	
22	OK	OK	
23	OK	OK	
25	OK	OK	
26	OK	OK	
27	OK	OK	
32	OK	OK	
33	OK	OK	
34	OK		input only
35	OK		input only
36	OK		input only
39	OK		input only

Getting started with the ESP32, what do I need:

https://randomnerdtutorials.com/install-esp32-filesystem-uploader-arduino-ide/

https://randomnerdtutorials.com/esp32-ota-over-the-air-arduino/

https://randomnerdtutorials.com/esp32-ssd1306-oled-display-arduino-ide/

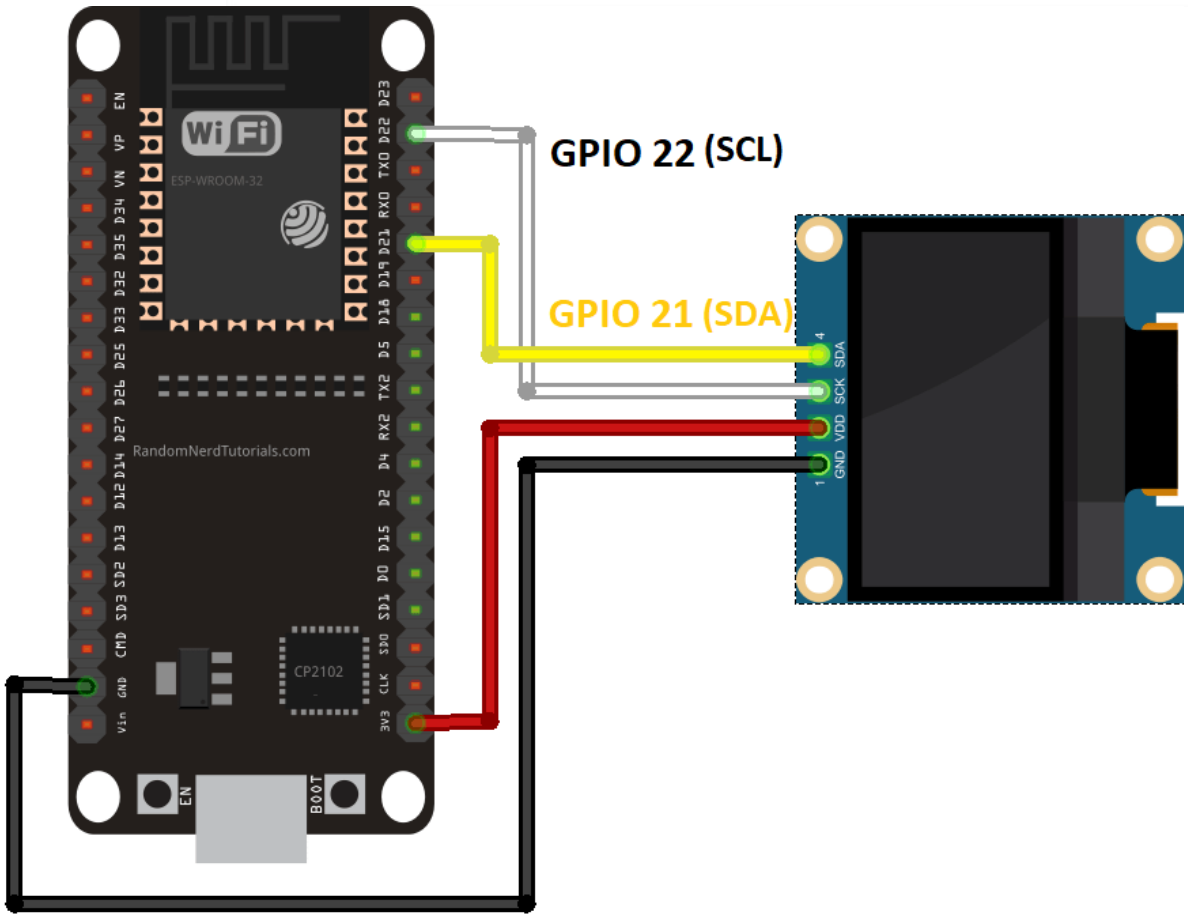
Motorola MC74F240N

= 0ctal buffer + inverter

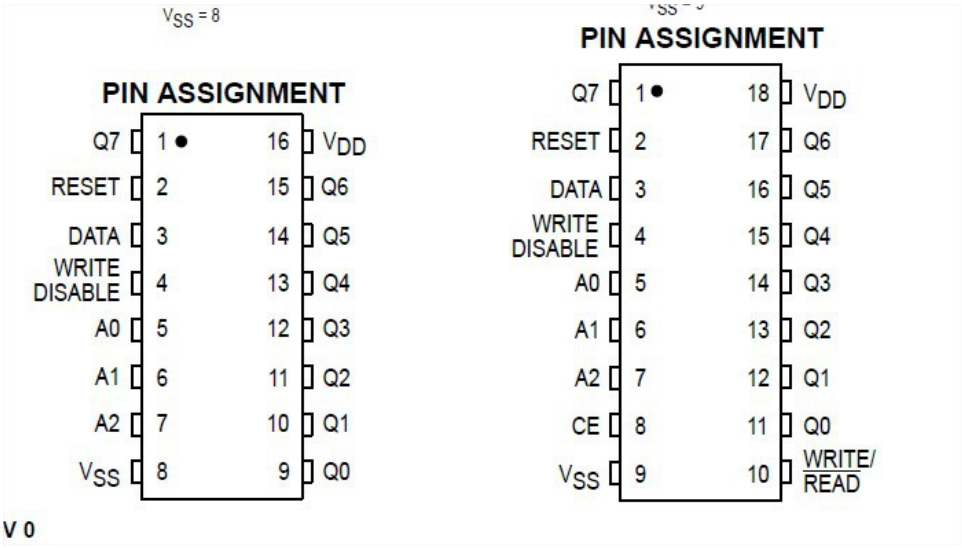
OLED Display SSD1306 Pin Wiring

Because the OLED display uses I2C communication protocol, wiring is very simple. You can use the following table as a reference.

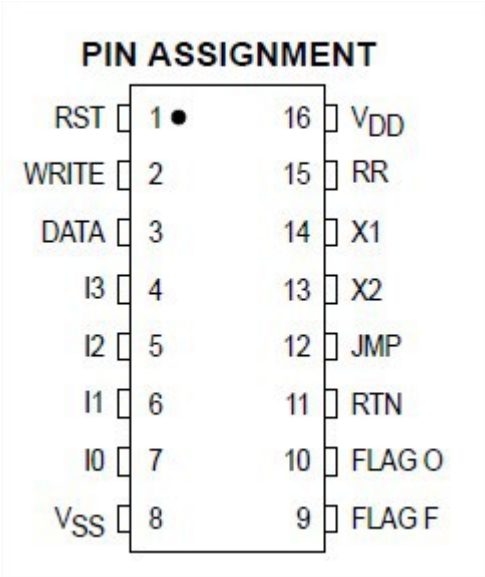
Pin	ESP32
Vin	3.3V
GND	GND
SCL	GPIO 22
SDA	GPIO 21



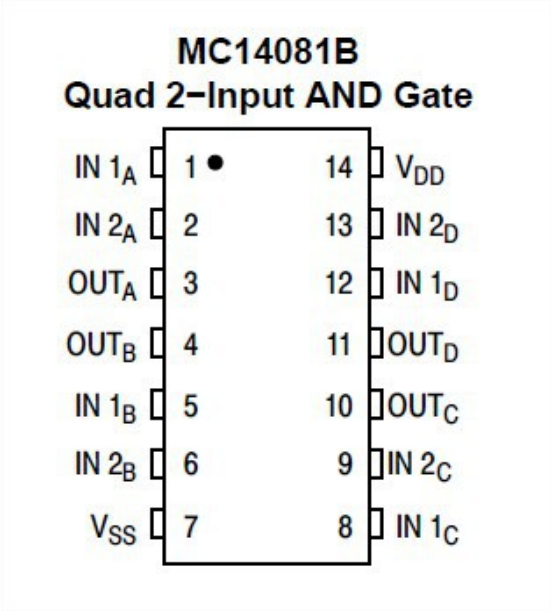
14099 / 14599



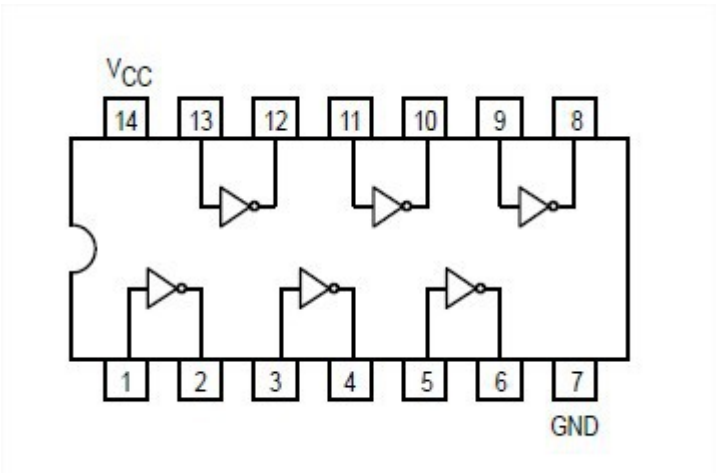
14500



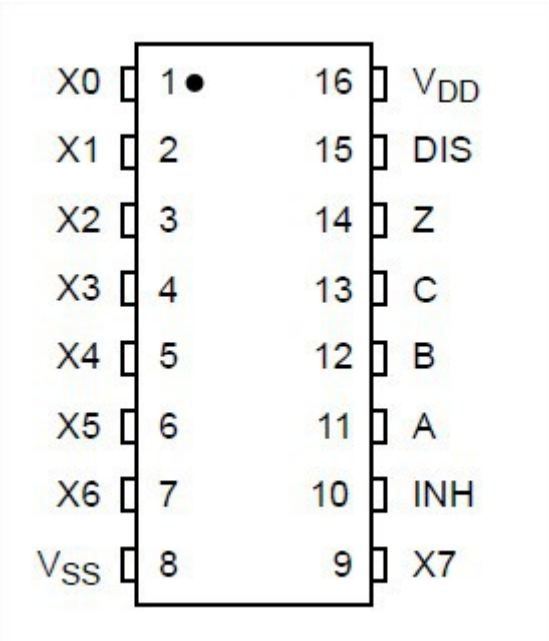
14081

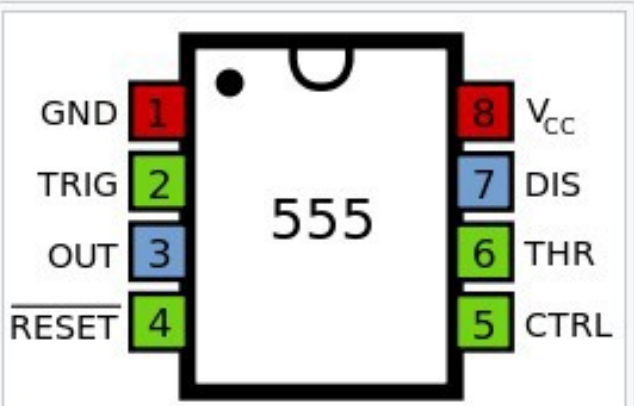


F74F04

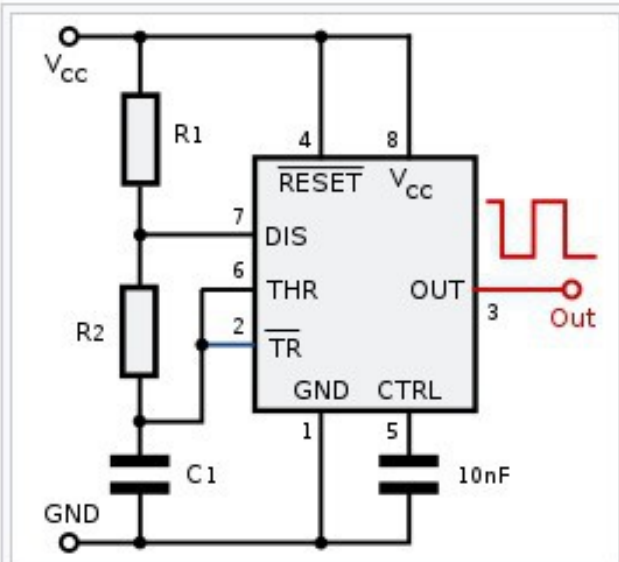


14512

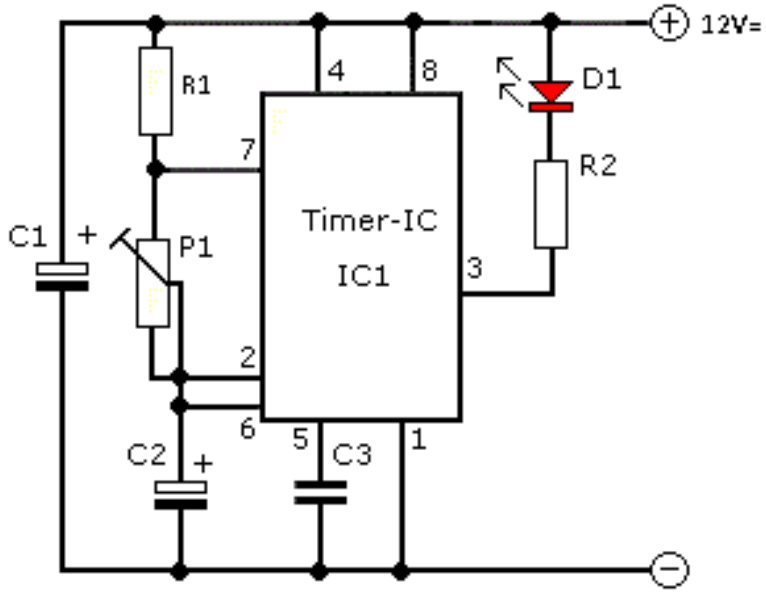


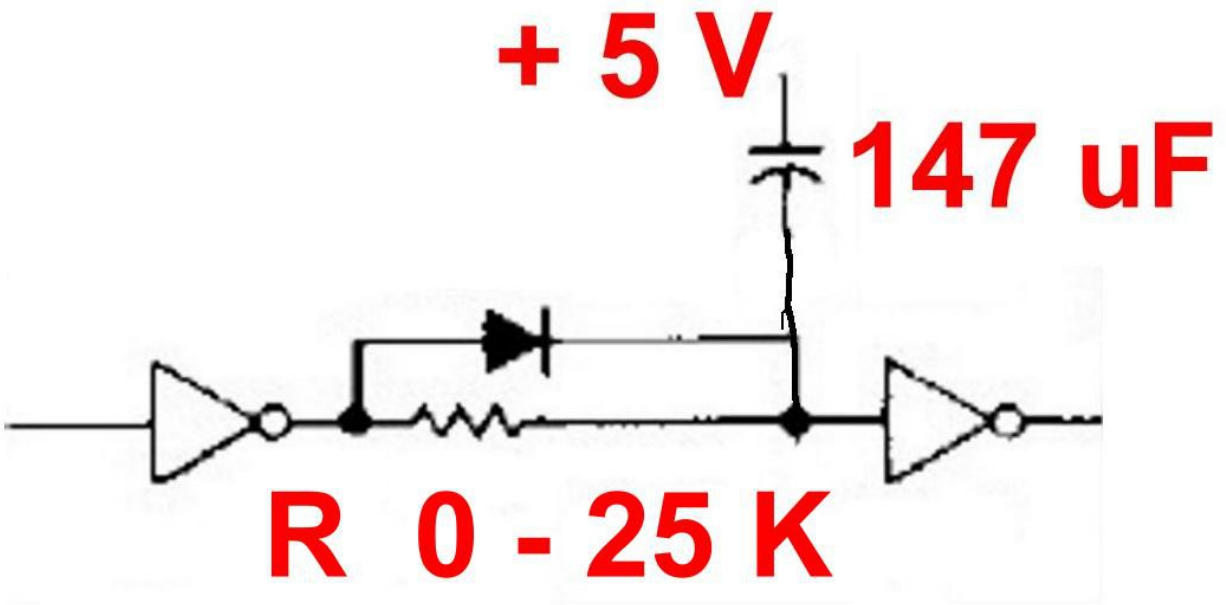


Aansluitingen van de NE555N (DIL8)



De 555 als astabiele multivibrator





The Timer has the following properties, multi-turn potentiometer of 25 K and two capacitors of 47 uF and 100 uF in parallel. This results in a very short delay of up to 1 second. If I calculate RC time, I already get 0.9 sec at an R of 25K and a C of 47 uF, but it seems that the input resistance of the 74F04 is throwing a spanner in the works. Or the diode that is parallel to the R accelerate the charging of the C.

All well and good, this works.

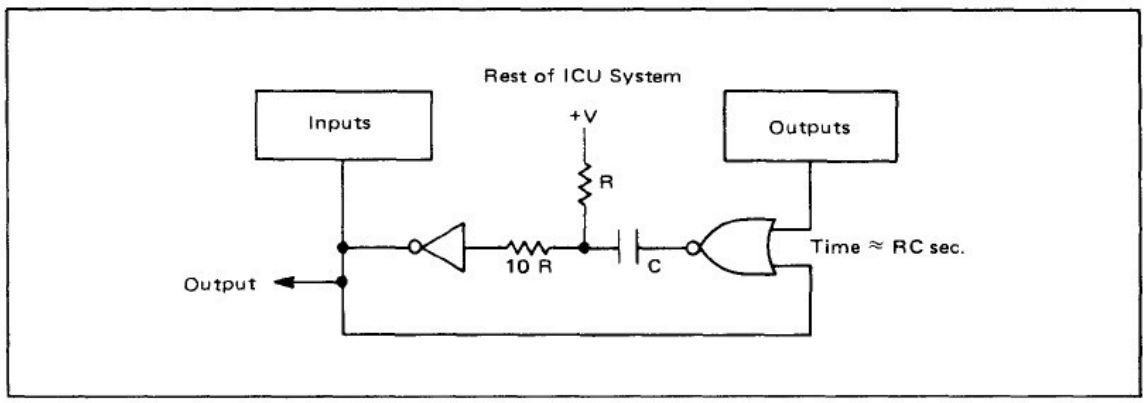
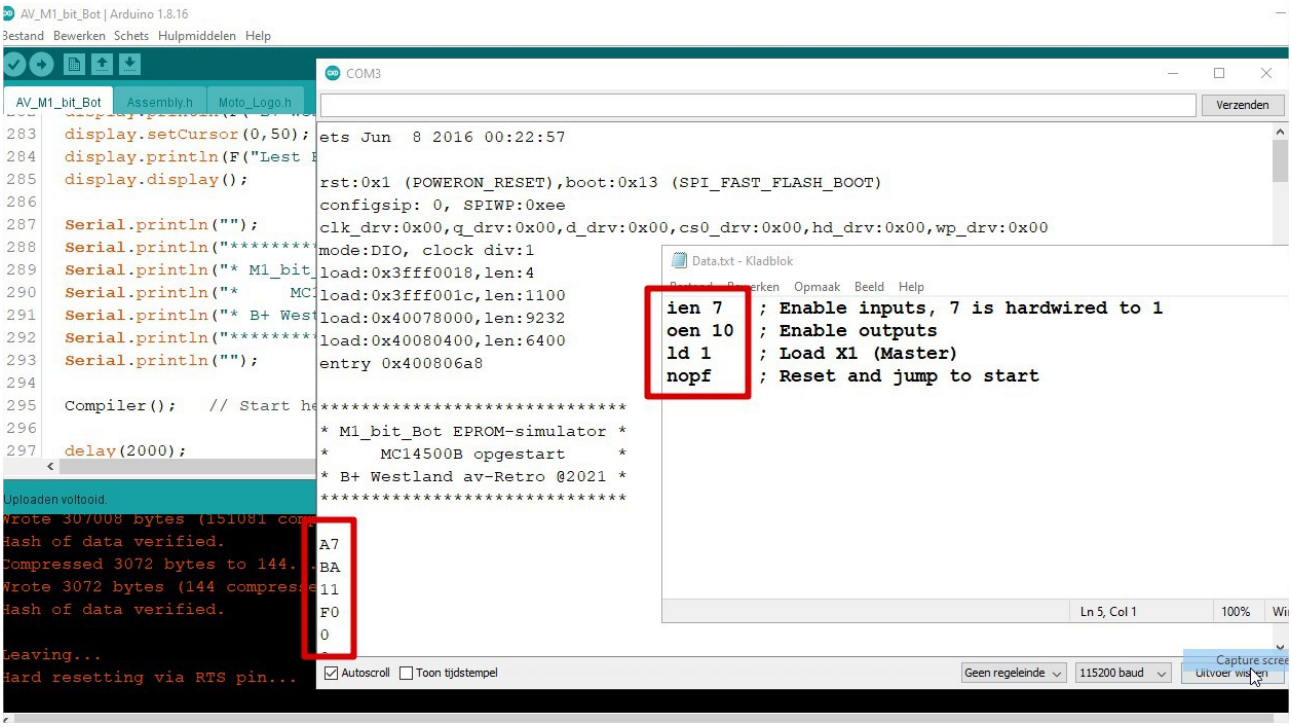


Figure 6.2 CMOS Monostable Timer

Motorola Logo, 128 * 64 pixel:

```
static const unsigned char PROGMEM logo_bmp[] ={
0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x03, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x80, 0x00, 0x00, 0x1e, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x80, 0x00, 0x00, 0x1e, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x80, 0x00, 0x00, 0x1e, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x0f, 0xc0, 0x00, 0x00, 0x3f, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x0f, 0xc0, 0x00, 0x00, 0x3f, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x0f, 0xc0, 0x00, 0x00, 0x3f, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x1f, 0xc0, 0x00, 0x00, 0x3f, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x1f, 0xe0, 0x00, 0x00, 0x7f, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x1f, 0xe0, 0x00, 0x00, 0x7f, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x1f, 0xe0, 0x00, 0x00, 0x7f, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x3f, 0xe0, 0x00, 0x00, 0xff, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x3f, 0xf0, 0x00, 0x00, 0xff, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x3f, 0xf0, 0x00, 0x00, 0xff, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x7f, 0xf0, 0x00, 0x01, 0xff, 0xe0, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x7f, 0xf8, 0x00, 0x01, 0xff, 0xe0, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x7f, 0xf8, 0x00, 0x01, 0xff, 0xe0, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x7f, 0xf8, 0x00, 0x01, 0xff, 0xf0, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0xff, 0xfc, 0x00, 0x03, 0xff, 0xf0, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0xff, 0xfc, 0x00, 0x03, 0xff, 0xf0, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0xff, 0xfc, 0x00, 0x03, 0xff, 0xf0, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x01, 0xff, 0xfe, 0x00, 0x07, 0xff, 0xf8, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x01, 0xff, 0xfe, 0x00, 0x07, 0xff, 0xf8, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x01, 0xff, 0xfe, 0x00, 0x07, 0xff, 0xf8, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x03, 0xff, 0xff, 0x00, 0x0f, 0xff, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x03, 0xff, 0xff, 0x00, 0x0f, 0xff, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x03, 0xff, 0xff, 0x00, 0x0f, 0xff, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x03, 0xff, 0xff, 0x80, 0x0f, 0xff, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x07, 0xff, 0xff, 0x80, 0x1f, 0xff, 0xfe, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x07, 0xff, 0xff, 0x80, 0x1f, 0xff, 0xfe, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x07, 0xff, 0xff, 0x80, 0x1f, 0xff, 0xfe, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x0f, 0xff, 0xff, 0xc0, 0x3f, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x0f, 0xff, 0xff, 0xc0, 0x3f, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x0f, 0xff, 0xff, 0xc0, 0x3f, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x1f, 0xff, 0xff, 0xe0, 0x3f, 0xff, 0xff, 0x80, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x1f, 0xff, 0xff, 0xe0, 0x7f, 0xff, 0xff, 0x80, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x1f, 0xff, 0xff, 0xe0, 0x7f, 0xff, 0xff, 0x80, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x3f, 0xf8, 0x7f, 0xe0, 0x7f, 0xc3, 0xff, 0x80, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x3f, 0xe0, 0x0f, 0xf0, 0xff, 0x00, 0x3f, 0xc0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x3f, 0x80, 0x07, 0xf0, 0xfe, 0x00, 0x1f, 0xc0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x3f, 0x80, 0x01, 0xf0, 0xfc, 0x00, 0x0f, 0xe0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x7e, 0x00, 0x01, 0xf9, 0xf8, 0x00, 0x07, 0xe0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x7c, 0x00, 0x00, 0xf9, 0xf0, 0x00, 0x03, 0xe0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x7c, 0x00, 0x00, 0x79, 0xe0, 0x00, 0x03, 0xe0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0xfc, 0x00, 0x00, 0x7f, 0xe0, 0x00, 0x01, 0xf0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0xf8, 0x00, 0x00, 0x3f, 0xc0, 0x00, 0x01, 0xf0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0xf0, 0x00, 0x00, 0x3f, 0xc0, 0x00, 0x00, 0xf0, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0xf0, 0x00, 0x00, 0x1f, 0x80, 0x00, 0x00, 0xf8, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x01, 0xe0, 0x00, 0x00, 0x0f, 0x80, 0x00, 0x00, 0x78, 0x00, 0x00, 0x00, 0x00,
```

```
0x00, 0x00, 0x00, 0x01, 0xe0, 0x00, 0x00, 0x0f, 0x00, 0x00, 0x00, 0x78, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x01, 0xe0, 0x00, 0x00, 0x0f, 0x00, 0x00, 0x00, 0x78, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x03, 0xc0, 0x00, 0x00, 0x06, 0x00, 0x00, 0x00, 0x3c, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x03, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3c, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x03, 0xc0, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3c, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x07, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1e, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x07, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1e, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x07, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x1e, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x0f, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0f, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x0f, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x0f, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x0f, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x0e, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x07, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
 };
```



MC14500B-Assembler:

This looks quite easy / simple, we will also make it in the ESP32. In the second instance we will upload that IN the ESP32 using the web interface.

We have a few rules:

1 # Capital OR Lowercase letters only.
2 # A space is skipped/is redundant.
3 # After the ; is the comment.
4 # No address, than use 0.
5 # nopf/NOPF must -always- the last command.
6 # Maximum 64 characters on a line.
7 # Maximum 256 lines of code.

Error codes:

-301- Not a valid instruction
-302- No instruction but an address
-303- Letter/Instruction -after- address
-304- Program longer than 256 bytes