1.1

Tijd Onderdeel: 10:29

Embedded systems

1. Adapter telefoon
2. Rekenmachine
3. Zelfrijdende auto
4. Weerballon
5. Philips Hue
6. Router
7. Atlas & Spot
8. CT-Scanner
9. Oscilloscoop
10. Bluetooth speaker
11. Satelliet
12. Bloeddrukmeter
13. DAC
14. Printer
15. 3D-printer
16. Reclamemasten
17. Game boy
18. Digitale Camera
19. Snelheids
20. WG 54 snelheidsmeter

1.2

Tijd Onderdeel: 9:59

|  |  |
| --- | --- |
| 3.3V  1 kOhm | Ledje brand zacht.  Spanning Led: 1.8V  Spanning Weerstand: 1.5V  Stroom:3.3mA |
| 5V  1 kOhm | Het Ledje brand feller  Spanning LED: 1.8V  Spanning Weerstand:3.2V  Stroom:5mA |
| 3.3V  100 Ohm | Het Ledje brand nog feller  Spanning LED: 1.8V  Spanning Weerstand: 1.5V  Stroom: 33 mA |

1.3

Tijd onderdeel: 1:02:00

|  |  |  |
| --- | --- | --- |
| **Naam** |  |  |
| Pulse | **import** RPi.GPIO **as** GPIO **import** time GPIO.setmode( GPIO.BCM ) GPIO.setwarnings( 0 )  print( "GPIO pulse" )  **def** pulse( pin\_nr, high\_time, low\_time ):  GPIO.output(pin\_nr,GPIO.HIGH)  time.sleep(high\_time)  GPIO.output(pin\_nr, GPIO.LOW)  time.sleep(low\_time)  """  Geef een puls op de pin:  Maak de pin pin\_nr hoog, wacht high\_time,  maak de pin laag, en wacht nog low\_time  """  *# implementeer dez<e functie*  led = 18 GPIO.setup( led, GPIO.OUT ) **while True**:  pulse( led, 0.2, 0.2 ) |  |
|  |  |  |
| Tatatataaa | **def** kort():  GPIO.output(18, GPIO.HIGH)  time.sleep(0.5)  GPIO.output(18, GPIO.LOW)  time.sleep(0.5)   **def** lang():  GPIO.output(18, GPIO.HIGH)  time.sleep(3)  GPIO.output(18, GPIO.LOW)  time.sleep(0.5)   **while True**:  kort()  kort()  kort()  lang() |  |
| Morse code | **ef** morse(pin\_nr, dot\_length, text):  **for** i **in** text:  **if** i == ' ':  GPIO.output(pin\_nr, GPIO.LOW)  time.sleep(dot\_length)  **else**:  GPIO.output(pin\_nr, GPIO.HIGH)  **if** i == '-':  time.sleep(dot\_length \* 3)  **elif** i == '.':  time.sleep(dot\_length)  GPIO.output(pin\_nr, GPIO.LOW)  time.sleep(dot\_length) |  |

1.4

Tijd onderdeel: 30:43

|  |  |
| --- | --- |
| Pulse | **def** servo\_pulse(pin\_nr, position):  x = (position \* 0.000025) + 0.0005  print(x)  GPIO.output(pin\_nr, GPIO.HIGH)  time.sleep(x)  GPIO.output(pin\_nr, GPIO.LOW)  time.sleep(0.002) |
| Blink | **while True**:  **for** i **in** range(0, 100, 1):  servo\_pulse(servo, i)  time.sleep(0.5)  **for** i **in** range(100, 0, -1):  servo\_pulse(servo, i)  time.sleep(0.5) |