



Del 2 – Grundläggande kretsar

Grundläggande elektronik och programmering

I den här delen kommer du att lära dig att använda ett kopplingsdäck för att bygga en enkel krets med en lysdiod och ett motstånd. Du kommer att skriva kod för att tända, släcka och blinka lysdioden med hjälp av CircuitPython.

Material

- Adafruit Feather RP2040
- Kopplingsdäck
- USB-sladd för att koppla Feather till datorn
- Dator
- Motstånd
- Lysdiod
- Sladdar för koppling

Pins

Kopplingspunkterna runt om mikrokontrollern kallas för pins. Dessa används för att komma åt olika funktioner. Pinsen kan delas in i några kategorier:

- Ström: Här ingår batteri kontakten, positiv spänning (3.3 volt), samt jord (GND).
- Digitala ingångar och utgångar: Dessa benämns med endast siffror och kan användas antingen för att driva en utgångsström eller för att läsa av en ingångssignal.
- Analoga ingångar: Dessa benämns med ett A följt av en siffra. Deras uppgift är att läsa analoga signaler
- Kommunikation: Feathern har inbyggt stöd för ett antal olika kommunikationsprotokoll såsom I2C, SPI och UART.

Blinka en lampa

På Feather-kortet finns det tre stycken inbyggda lampor.

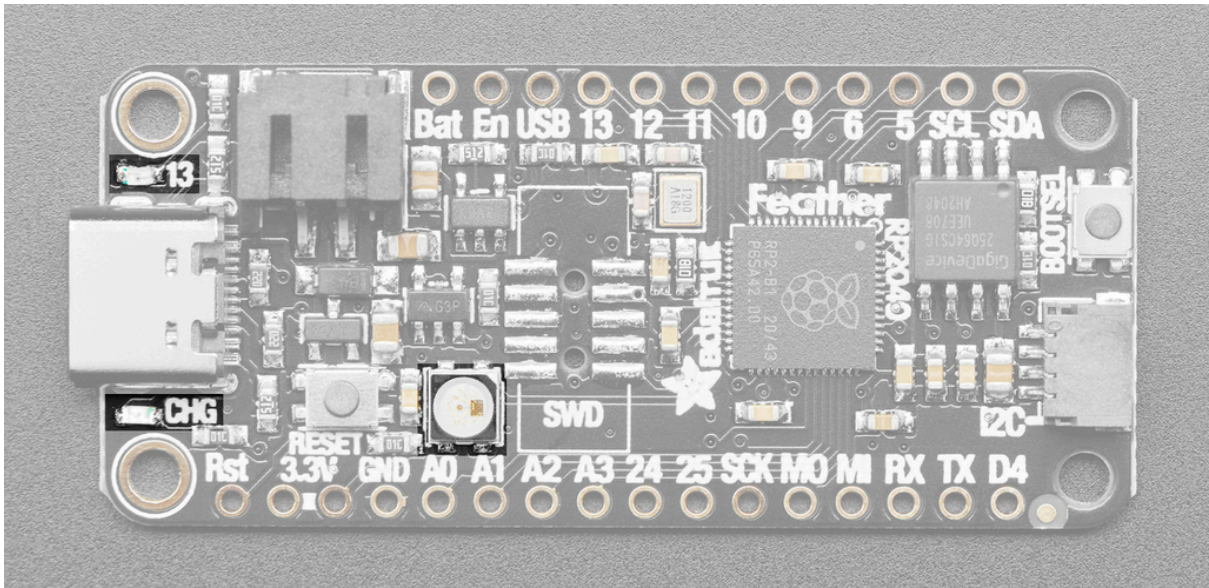
E-post: cansat@au.se

Telefon: 070-000 90 56

Senast uppdaterad: 2024-11-19



**Astronomisk
Ungdom**



1

- Lampan under USB-porten med texten "CHG" indikerar om ett eventuellt inkopplat batteri laddas.
- Den fyrkantiga lampan ovanför A0 och A1 är en flerfärgslampa som kan användas till mer avancerad signalering.
- Lampan ovanför USB-porten med texten "13" är den vi kommer använda till en början med i denna laboration. Den är kopplad direkt till den digitala pin 13.

Styra digitala utgångar med kod

För att styra en digital ingång behöver vi först ställa in om det ska vara en ingång eller utgång, sedan kan vi använda `.value` för att läsa eller styra den.

Den inbyggda lampan kan benämnas med antingen `board.LED` eller `board.D13`

¹ RP_pinouts_LEDs, By Kattni Rembor, CC BY-SA 3.0, <https://learn.adafruit.com/assets/100375>

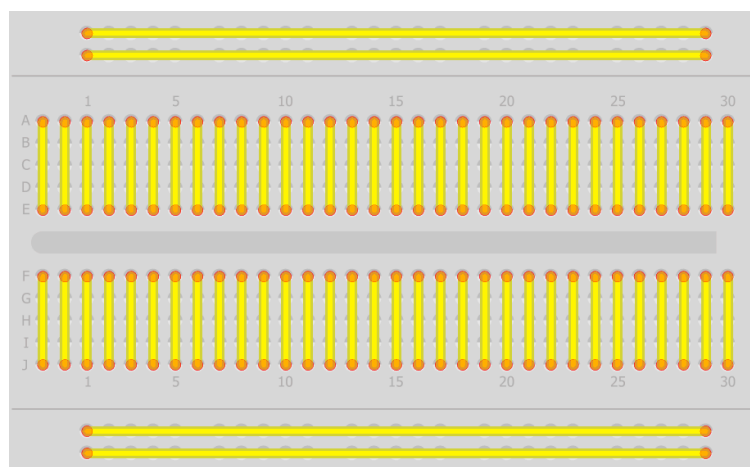
code.py

```
1 import board
2 import digitalio
3 import time
4
5 # Ställ in LED-pinnen
6 led = digitalio.DigitalInOut(board.LED)
7 led.direction = digitalio.Direction.OUTPUT
8
9 # Tända och släcka LED i en loop
10 while True:
11     led.value = True # Tänder LED
12     time.sleep(1)   # Väntar 1 sekund
13     led.value = False # Släcker LED
14     time.sleep(1)   # Väntar 1 sekund
15
```

Spara koden och observera att den inbyggda lampan blinkar.

Kopplingsdäck och Kretsar

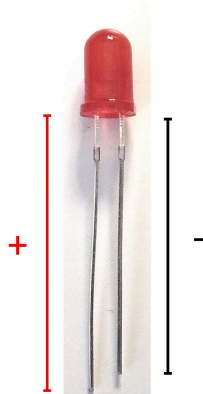
Vi kan enkelt utöka vår installation till att styra flera dioder med hjälp av en *kopplingsplatta*. En kopplingsplatta fungerar genom att hålen på samma rad är kopplade till varandra så att man kan sticka in de elektroniska komponenternas "ben" och göra en krets utan att behöva löda något. Bilden nedan visar hur hålen på en kopplingsplatta är sammankopplade. De långa skenorna längst ut på sidorna är användbara för att koppla spänning och jord till flera komponenter.



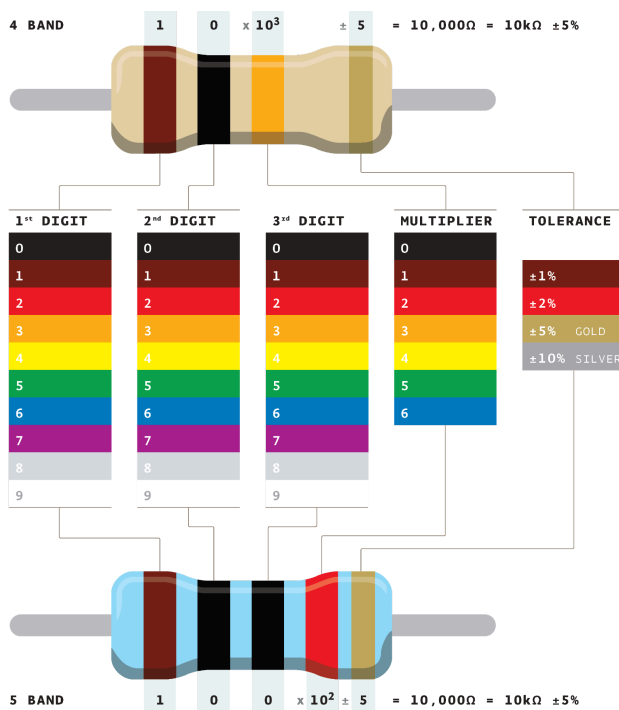
fritzing

Bygga en lysdiodkrets på kopplingsdäcket

En *lysdiod* är en enkel komponent som lyser när ström går igenom den. Lysdioder är "enkelriktade", d.v.s. de kommer bara låta strömmen gå genom dem åt ett håll. Man kan se vilket håll dioden ska vara kopplad åt genom att jämföra längden på benen. Kort ben mot minus (jord) och långt ben mot plus. Lysdioden kommer även sakna en bit av kanten som går runt den längst ner på jord-benet.

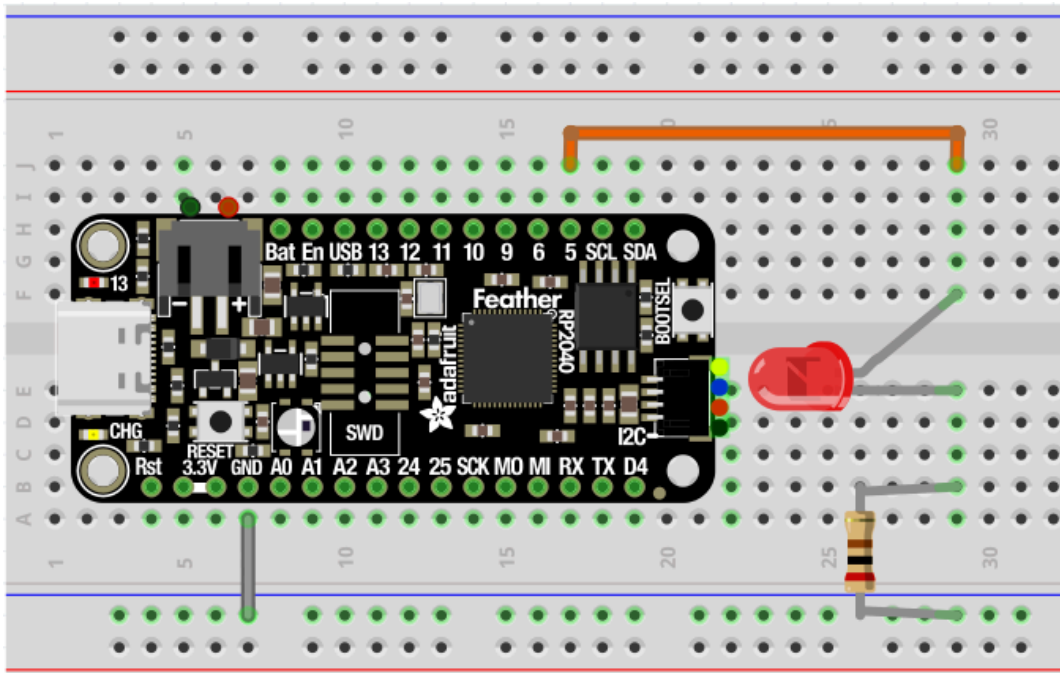


För att styra en lysdiod med Feather-kortet behöver vi en enkel krets med en lysdiod och ett motstånd. Motståndet hjälper till att begränsa strömmen och skydda lysdioden. För att ta reda på värdet på ett motstånd kan man använda följande tabell:



Vi behöver tala om vilken pin vi ska koppla dioden på. Den pin som heter 5 på Feathern kommer vi att använda och bli den som matar spänning ut till en diod.

Att göra: Koppla pin 5 till det positiva benet av lysdioden, koppla diodens negativa ben till ena benet av ett motstånd på 200 Ω . Motståndets andra ben kopplas till GND (jord).



Programmera styrning av lampan

Använd samma kod som tidigare, fast byt nu ut board.LED mot den digitala utgång du valt att använda.



code.py

```
1 import board
2 import digitalio
3 import time
4
5 # Ställ in LED-pinnen (kopplad till D5 i detta exempel)
6 led = digitalio.DigitalInOut(board.D5)
7 led.direction = digitalio.Direction.OUTPUT
8
9 # Tända och släcka LED i en loop
10 while True:
11     led.value = True # Tänder LED
12     time.sleep(1)   # Väntar 1 sekund
13     led.value = False # Släcker LED
14     time.sleep(1)   # Väntar 1 sekund
15
```