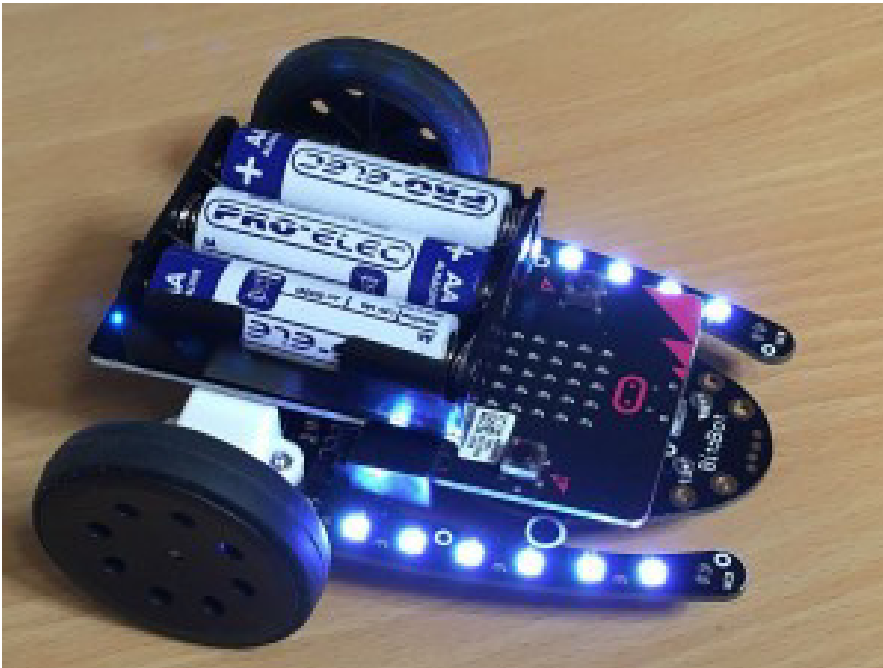


Bit:Bot – Introduktion og første opgave



INDLEDNING

Introduktion og program er hentet og oversat fra firmaet 4tronix hjemmeside:

<http://4tronix.co.uk/blog/?p=1490> samt fra Mark Atkinsons hjemmeside "Multiwingspan:"
<http://multiwingspan.co.uk/micro.php?page=bitbot>

Venlig hilsen og god fornøjelse!

Søren Westerholm, Coding Pirates Sorø.

Din Bit:Bot indeholder:

- 2 mikro- el-motorer med metal gear. Motorerne er 100 % software-kontrollerede, både med hensyn til hastighed og retning.
- Gummihjul, som holder godt fast i underlaget.
- Letløbende frontkugle
- 12 mini Neopixels (lysdioder) - 2 sæt af seks stk. på hver side. Du vælger selv farver og effekter.
- 2 digitale sensorer, der gør det muligt for Bit:Bot at følge en linje. Ideelt til Robotræs!
- 2 analoge lys-sensorer - 1 i venstre og 1 i højre side af fronten. Bit:Bot kan programmeres til at følge lyset fra en lommelygte - eller gemme sig det mørkeste sted i rummet!
- En lyd giver, så du kan trutte i hornet når du har lyst!
- Bit:Bot får strøm fra en 3xAA batteriholder med afbryder og blå indikator LED.
- Din BBC Micro:Bit passer perfekt i den monterede kantkonnektor.
- Udvidelsesstik til yderligere Neopixels (som f.eks. McRoboFace)
- Udvidelsesstik på fronten til påsætning af yderligere sensorer.

NB! Til at begynde med, er det en fordel at programmere din Bit:Bot ved hjælp af [Microsoft PXT-Editoren](#).

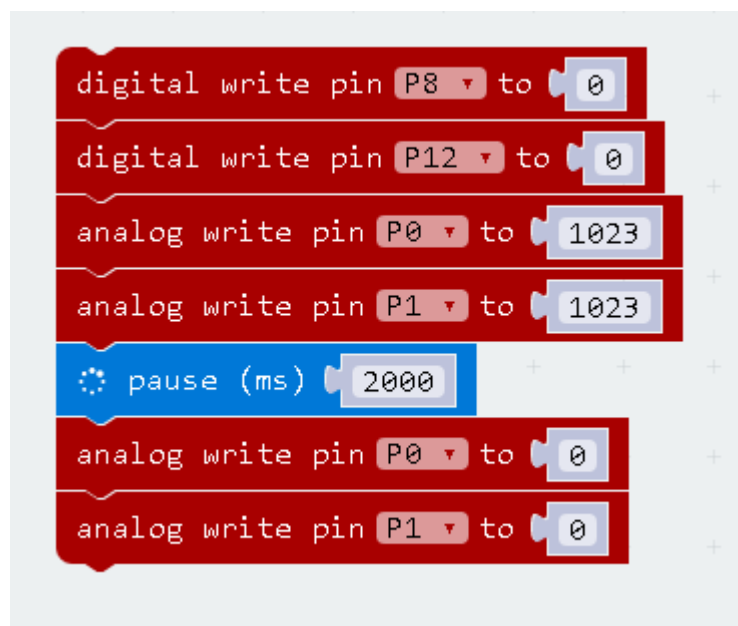
PROGRAM 1: KØR MED MOTORERNE

Motorerne er forbundet til følgende PINS på din Micro:Bit:

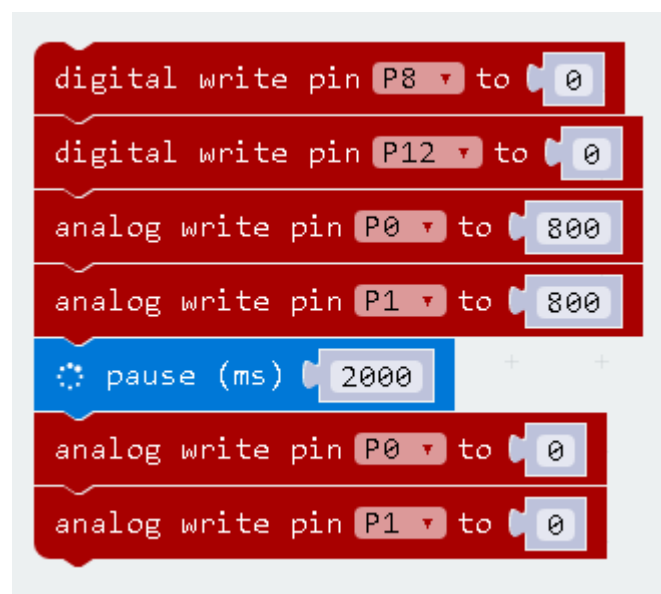
- **VENSTRE Motor:** PWM pin 0, DIR pin 8
- **HØJRE Motor:** PWM pin 1, DIR pin 12

Du skal bruge kommandoen **digital write** på **IDR** (retningspindene), hvor 0 er frem og 1 er tilbage. Du skal bruge **analog write** på **PWM** (hastighedspin) til at styre motorens fart.

Hvis du eksempelvis vil køre fremad med fuld fart i 2 sekunder og derefter stoppe, ser blokkene ud som vist nedenunder:

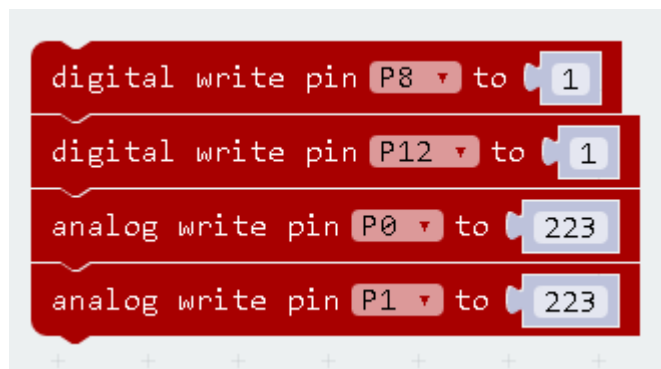


I eksemplet herunder er hastigheden sat til 80 %:



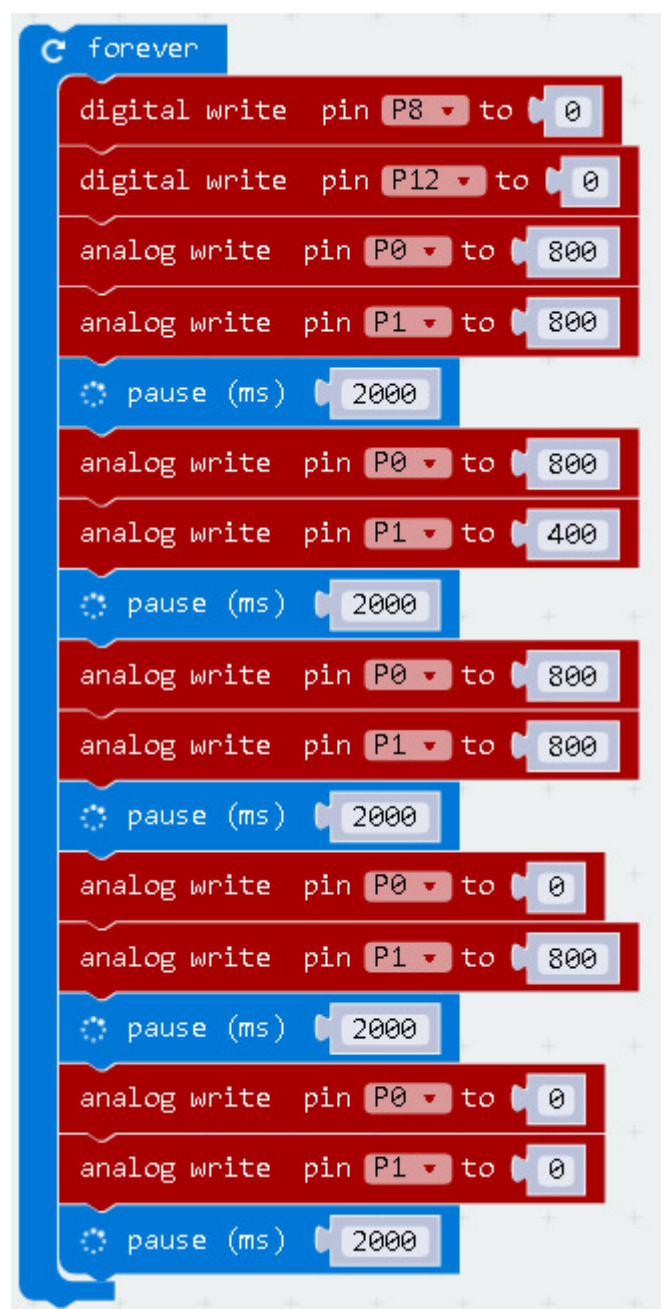
Hvad nu, hvis du gerne vil have din Bit:Bot til at køre langsomt og baglæns? Hvad vil du så ændre i programmet?

Skulle det drille, kan du altid få hjælp herunder:



```
digital write pin P8 to 1
digital write pin P12 to 1
analog write pin P0 to 223
analog write pin P1 to 223
```

Prøv nu at sætte programstumperne sammen til et helt program. Har du brug for hjælp, så kig herunder:



```
forever
  digital write pin P8 to 0
  digital write pin P12 to 0
  analog write pin P0 to 800
  analog write pin P1 to 800
  pause (ms) 2000
  analog write pin P0 to 800
  analog write pin P1 to 400
  pause (ms) 2000
  analog write pin P0 to 800
  analog write pin P1 to 800
  pause (ms) 2000
  analog write pin P0 to 0
  analog write pin P1 to 800
  pause (ms) 2000
  analog write pin P0 to 0
  analog write pin P1 to 0
  pause (ms) 2000
```

I JavaScript delen af PXT Editoren ser programmerne ud som vist herunder.

Er du en doven hund, behøver du blot at klippe / klistre JavaScriptene ind i PXT-Editoren for derefter at kigge nærmere på blokkene ☺

```
basic.forever(() => {
  pins.digitalWritePin(DigitalPin.P8, 0)
  pins.digitalWritePin(DigitalPin.P12, 0)
  pins.analogWritePin(AnalogPin.P0, 800)
  pins.analogWritePin(AnalogPin.P1, 800)
  basic.pause(2000)
  pins.analogWritePin(AnalogPin.P0, 800)
  pins.analogWritePin(AnalogPin.P1, 400)
  basic.pause(2000)
  pins.analogWritePin(AnalogPin.P0, 800)
  pins.analogWritePin(AnalogPin.P1, 800)
  basic.pause(2000)
  pins.analogWritePin(AnalogPin.P0, 0)
  pins.analogWritePin(AnalogPin.P1, 800)
  basic.pause(2000)
  pins.analogWritePin(AnalogPin.P0, 0)
  pins.analogWritePin(AnalogPin.P1, 0)
  basic.pause(2000)
})
```

For at gøre det nemmere at læse og forstå et JavaScript, er der mulighed for at tilføje kommentarer. Kommentarerne starter altid med to skråstreger og et mellemrum (//)

Herunder er det samme JavaScript vist igen, men denne gang med kommentarer.

```
basic.forever(() => {
  // Set up direction pins for forward motion
  pins.digitalWritePin(DigitalPin.P8, 0)
  pins.digitalWritePin(DigitalPin.P12, 0)
  // forwards
  pins.analogWritePin(AnalogPin.P0, 800)
  pins.analogWritePin(AnalogPin.P1, 800)
  basic.pause(2000)
  // slow turn right
  pins.analogWritePin(AnalogPin.P0, 800)
  pins.analogWritePin(AnalogPin.P1, 400)
  basic.pause(2000)
  // forwards
  pins.analogWritePin(AnalogPin.P0, 800)
  pins.analogWritePin(AnalogPin.P1, 800)
  basic.pause(2000)
  // sharp left turn
  pins.analogWritePin(AnalogPin.P0, 0)
  pins.analogWritePin(AnalogPin.P1, 800)
  basic.pause(2000)
  // coast/stop
  pins.analogWritePin(AnalogPin.P0, 0)
  pins.analogWritePin(AnalogPin.P1, 0)
  basic.pause(2000)
})
```

UDFORDRING:

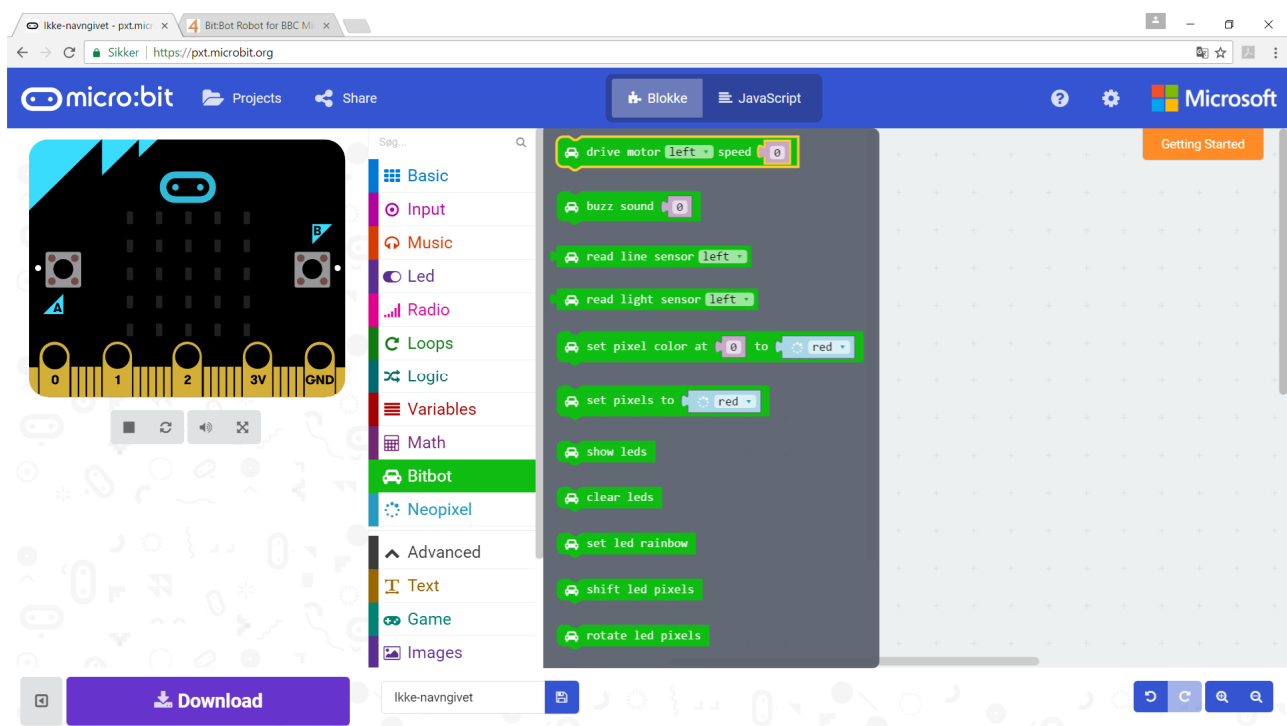
1. Lav et program, der fører din Bit:Bot tilbage til startstedet!
2. Lav en forhindringsbane som din Bit:Bot herefter skal køre fejlfrit igennem på kortest tid!

TILFØJELSESPAKKE TIL PXT_EDITOREN

[Microsoft PXT-Editoren](https://pxt.microbit.org). Er blevet udvidet med en "Bit:Bot" pakke.

- Du installerer pakken, når du har åbnet PXT-Editoren.
- Klik på "Advanced" folderen.
- Klik på "Add Package" og søg efter "Bit:Bot."

Din PXT-Editor har nu fået tilføjet nogle nye "Byggeklodser," som passer perfekt til programmeringen af din Bit:Bot!



UDFORDRING:

3. Løs OPGAVE 1, men gør programmeringen nemmere og simplere ved at bruge dine nye "Byggeklodser."