



Micro:bit

PXT: Ballong Battle

Skrevet av: Sigurd Schaathun

Kurs: Microbit

Tema: Elektronikk, Blokkbasert, Spill

Fag: Matematikk, Naturfag, Programmering, Teknologi

Klassetrinn: 5.-7. klasse, 8.-10. klasse

Introduksjon

I super:bitpakken er Bit:Bot, en bil som vi kan programmere med micro:bit. I denne oppgaven skal vi lage en fjernstyring til den slik at vi kan ha battle.

Steg 1: Radiokommunikasjon

I de første oppgavene med Bit:Bot, er alt forhåndsprogrammert på en micro:bit som festes på bit:bot. Vi vil fjernstyre bit:bot og da trenger vi to micro:bit som snakker sammen.



Sjekkliste



I seksjonen Radio finner du alle kodeblokkene for å bruke radiofunksjonene.

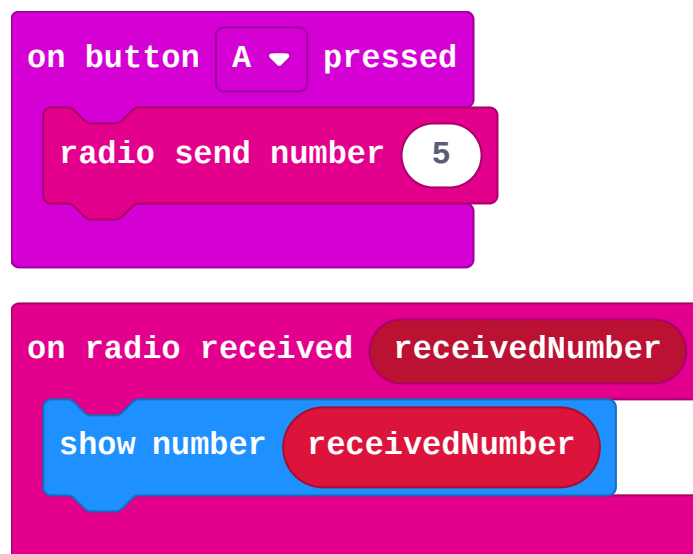


I kodeblokken ved start må du fortelle micro:biten hvilken radiokanal som skal brukes. Sett inn radio sett gruppe og velg f. eks. 42. Alle micro:bit som skal snakke sammen må ha samme kanal. Er du i kodeklubben eller på skolen vil instruktøren eller læreren gi deg en kode.



Når du trykker på A skal micro:bit sende et tall.

- ☐ Når micro:biten mottar et tall, skal tallet vises.



Test prosjektet

Last inn programmet på 2 micro:bit-er.

- ☐ Mottar den andre micro:bit-en tallet når du sender ved å trykke på A?
- ☐ Hvor langt unna kan man være og likevel motta?

Steg 2: Fjernkontrollprogrammet

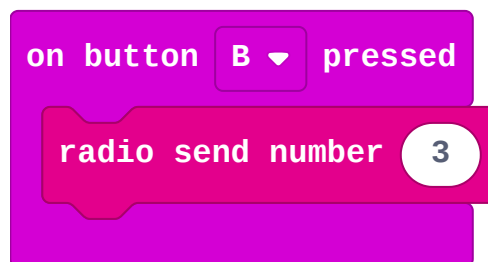
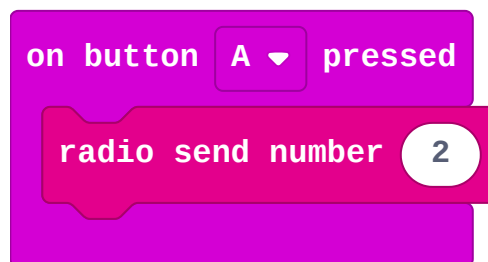
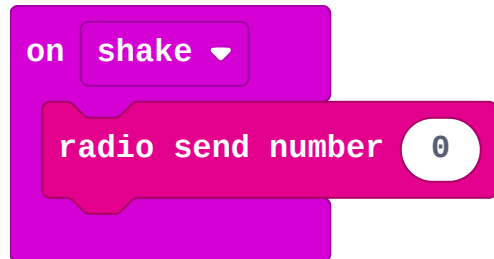
Nå skal vi sette opp et enkelt fjernstyringsprogram. Micro:bit-en skal sende tall for å få bit:boten til å gjøre ulike ting. Foreksempel kan 0 være stopp, 1 være å kjøre fremover, 2 sving venstre og 3 sving høyre.

Sjekkliste

- ☐ Når micro:bit ristes, skal den sende 0.
- ☐ Når vi trykker på A og B, skal den sende 1.

☐ Når vi trykker på A, skal den sende 2.

☐ Når vi trykker på B, skal den sende 3.



Test prosjektet

Last inn programmet på den ene micro:biten.

☐ Prøv å trykke på knappene og riste. Mottar den andre micro:biten kommandoene?

☐ Ser du noe som mangler i programmet? Hvordan kan vi gi beskjed om å kjøre bakover?

Tips

Micro:bit har en gyro som kan registrere om den holdes vannrett eller tipper oppover eller nedover ved å bruke blokken `helningsvinkel tonehøyde` under `Inndata-mer`. Her er oversettelsen litt dårlig pitch er oversatt med tonehøyde. Det kan vi bruke for å bestemme om den skal kjøre fremover eller bakover. I tillegg kan vi bruke en `hvis`-løkke og sjekke hvordan knappene er trykket inn.

Sjekkliste

- ☐ Legg inn en `hvis`-løkke i for alltid kodeblokken.
- ☐ Når `helningsvinkel tonehøyde` er mindre enn 0, skal vi ha en ny `hvis`-løkke.
- ☐ Hvis knapp `A+B` trykkes, skal den sende tall 1,
- ☐ Hvis ingenting trykkes, skal den sende tall 0.
- ☐ Legg til de øvrige kommandoene den skal sende kommandoer for å kjøre bakover.

forever

if rotation (°) pitch < 0 then

if button A+B is pressed then

radio send number 1

else if button A is pressed then -

radio send number 2

else if button B is pressed then -

radio send number 3

else -

radio send number 0

+

else -

if button A+B is pressed then

radio send number 4

else if button A is pressed then -

radio send number 5

else if button B is pressed then -

radio send number 6

else -

radio send number 0

+



Test prosjektet

Last inn programmet på den ene micro:biten.

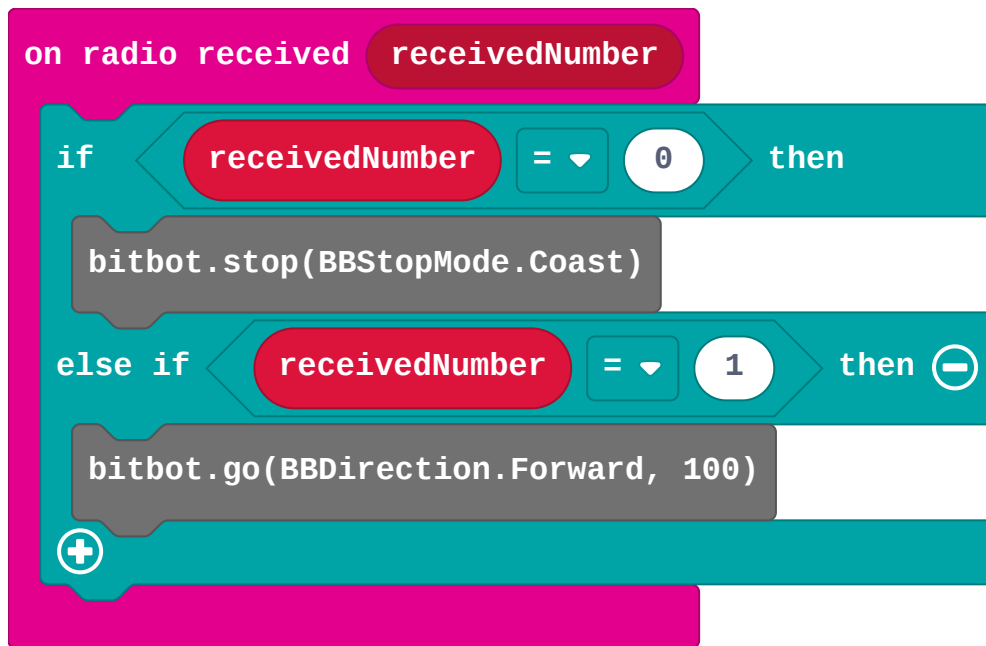
- ☐ Prøv å sende kommandoene. Mottar den andre micro:biten kommandoene riktig?
- ☐ Hvorfor sender den 0 når ingen knapper er trykket?

Steg 3: Kjøre bilen

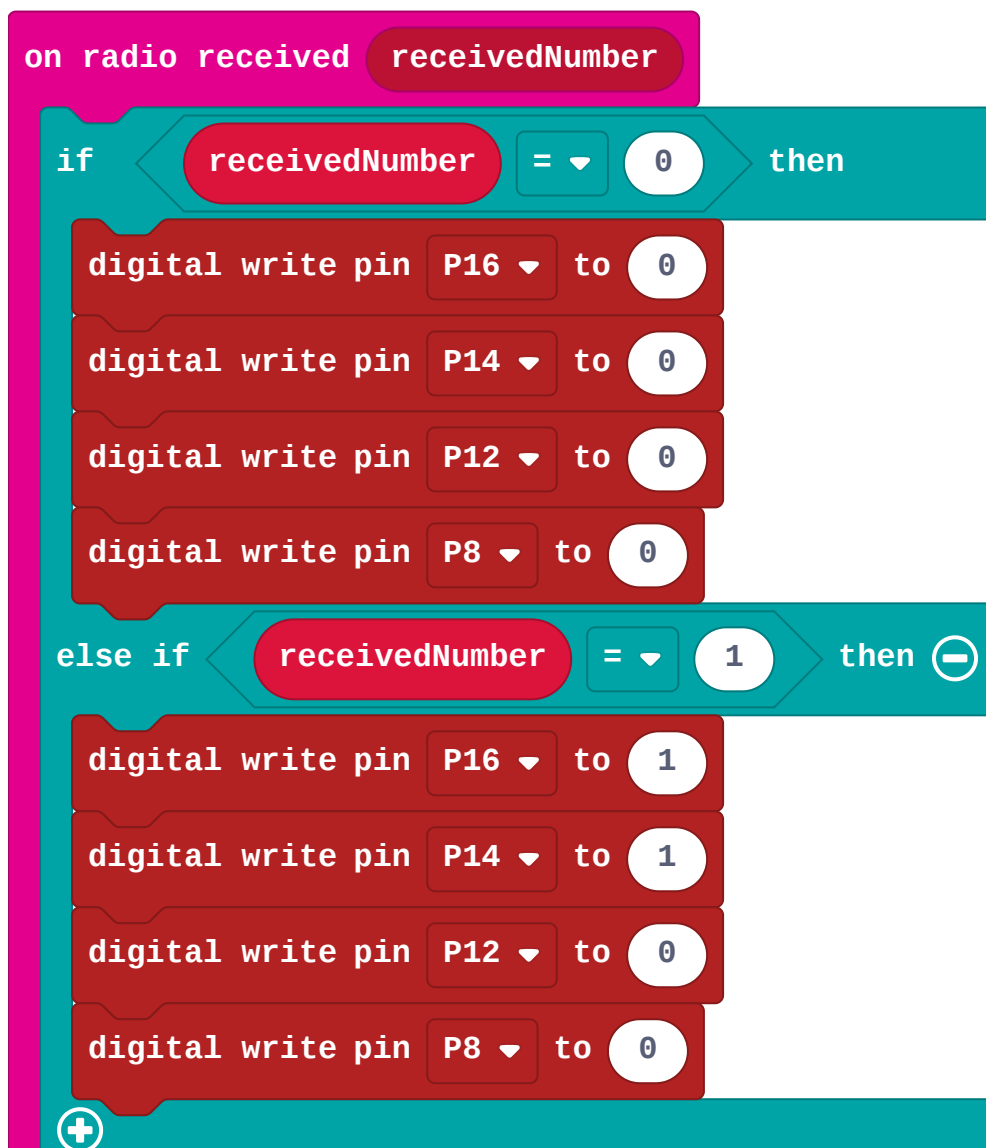
Nå må vi lage programmet på bit:boten. Den skal oversette tallene til bevegelse. Her er det to måter å gjøre det på. Du kan legge til egne bit:bot-koder. Da går du nederst under avansert og velger utvidelser. Der velger du bit:bot og får opp en egen fane som heter bit:bot. Den andre måten er å hardkode - med å bruke `skriv digital` eller `skriv analog`. Da må du finne ut hvilken pinne som gjør hva.

Sjekkliste

- ☐ Her trenger du Når radio mottar -blokken. Inni den legger du en hvis - løkke. Den skal sjekke hvilket kommandotall du mottar.
- ☐ For hvert tall, legg inn den funksjonen du ønsker, enten med bit:bot-kommandoer:



- ☐ Eller bruk hardkoding. Da må du finne ut hvilke pinner som gjør hva, og legge inn:



Tips

`skriv digital` er enten 0 eller 1. Hvis du vil kjøre med redusert fart, kan du bruke `skriv analog`. Da kan du justere kraften mellom 0 og 1023. Dette må vi bruke for å kunne svinge mens vi rygger.

På undersiden av `bit:bot`, står det hvilke pinner motoren er koblet til. Når det står P14/P8, er det en pinne for å kjøre fremover og en for å kjøre bakover. Hvis du setter strøm på begge samtidig, vil ikke motoren gå.

Test prosjektet

Last inn programmet på `micro:bit`en du setter i `bit:boten`.

- ☐ Får du bilen til å bevege seg?
- ☐ Lystrer den fjernstyringen?
- ☐ Hvordan er den å styre?

Utfordring

- ☐ På `Bit:boten` er det masse ledlys. Kan du få det til å blinke eller lyse når du kjører?
- ☐ Kan `bit:boten` få ansikt (bruk `vis bilde`)?
- ☐ Hva skjer hvis vi dekker til en `micro:bit` med aluminiumsfolie? Hvorfor?

BallongBattle

Da er det å blåse opp en ballong og dytte den nedi pennefestet på bit:boten, og feste en knappenål med tape foran på festet. Da kan dere prøve å sprekke ballongene til de andre, mens du beskytter din egen ballong.

Lisens: CC BY-SA 4.0 (<http://creativecommons.org/licenses/by-sa/4.0/deed>)