

Prøvetrykk 4.4

# Nils Kr. Rossing

# **MICRO:BIT**

# Forslag til undervisningsopplegg



Laserkuttet og 3D-printet robot med Micro:bit

Januar 2018

Denne siden er blank

# MICRO:BIT

# Forslag til undervisningsopplegg

Nils Kr. Rossing



#### Micro:bit - Forslag til undervisningsopplegg

Trondheim 2017

ISBN 978-82-92088-59-3

Bidragsytere: Nils Kr. Rossing, (nkr@vitensenteret.com) Vitensenteret i Trondheim

Layout og redigering: Nils Kr. Rossing, Vitensenteret i Trondheim

Tekst og bilder: Nils Kr. Rossing, Vitensenteret i Trondheim

Faglige spørsmål rettes til: Vitensenteret i Trondheim v/Nils Kr. Rossing, 73 59 77 23 nils.rossing@vitensenteret.com

Kongens gate 1 7013 Trondheim

Postboks 117 7400 Trondheim

Vitensenteret i Trondheim Telefon: 73 59 61 23 Telefaks: 73 59 61 20 http://www.vitensenteret.com/

Rev 4.4 - 26.01.18



#### Forord

Utgangspunktet for dette arbeidet var en henvendelse fra Jon Haavie i mars 2017 med forespørsel om det var mulig å lage en enkel robot basert på Micro:bit og laserkuttede deler. Den gang kom dette noe ubeleilig, men i løpet av sommeren 2017 har det vært tid til å gjøre noen enkle forsøk med et laserkuttet chassie og noen 3D-printete deler. Høsten 2017 ble det laget en prototyp som ble prøvd ut under Skaperfestivalen ved Teknisk Museum (Oslo) 9. og 10. november. I november laget Joachim Haagen Skeie et utlegg for et kretskort for tilkobling av servoer med mere til Micro:bit. Disse leveres både som byggesett og som ferdig oppkoblet og gjør byggingen av roboten enklere og mer fleksibel. I løpet av jula 2017 ble det laget en ny prototyp som prøves ut under formidlersamlingen i Sandnes jan./feb. 2018. I tillegg vil roboten bli brukt som gjennomgående tema i Makerlærerkurset ved Vitensenteret i Trondheim våren 2018.

Vitensenteret i Trondheim har også klassesett av Micro:bit og Inventors kit og vil være interessert i å kunne tilby disse overfor elevgrupper eller besøkende.

Utfordringen har primært vært å framskaffe et konsept som er enkelt nok og som samtidig gir muligheter for flinke elever til å utvikle roboten i de retningene de ønsker. En kan diskutere om et laserkuttet chassie gir den fleksibiliteten vi ønsker, siden det kreves at man må beherske det verktøyet som trengs for å lage underlaget og dessuten har tilgang til en laserkutter. Det samme gjelder de 3D-printede delene. Siden chassiet er relativt enkelt og billig å kutte ut i MDF, finer eller akryl, så kan en tenke seg å selge disse platene til en relativt lav pris. Videre kan en erstatte de 3D-printede delene med ferdig innkjøpt deler som heller ikke koster så mye.

Så det skal ligge en del muligheter i dette designet.

Trondheim Januar 2018

Nils Kr. Rossing Vitensenteret i Trondheim



## Innhold

Innledni	ng9
1.1 M	icro:bit og roboter9
Litt tekn	isk småplukk om Micro:bit11
2.1 Ka	nntkontakten11
2.2 Ba	tteripakke for Micro:bit13
Program	mering av Micro:bit – øvingsoppgaver14
3.1 Øv	vingsopplegg14
3.1.1	Øvingsoppgaver – grunnleggende14
3.1.2	Forslag til andre øvingsoppgaver16
3.1.3	Oppgaver knyttet til styring av servoer17
3.2 Ti	ps til blokkoding med Micro:bit18
3.2.1	Programvare, arbeidsflaten og lagring av prosjektfiler18
3.2.2	Noen blokker
3.3 Ju	stering av 360 servoer
Den mag	iske terningen - en innledende oppgave27
4.1 Pr	oblemstilling
4.2 Ur	ndervisningsopplegg28
4.3 Fr	amstilling av den tradisjonelle terningen
4.4 Pr	ogrammet
4.4.1	Program Micro:bit 1 (mottaker)
4.4.2	Program Micro:bit 2 (sender)
Micro:bi	t robot med tilhørende undervisningsopplegg32
5.1 In	nledende betraktninger om valg av løsning
5.1.1	Chassie
5.1.2	Sensorer og aktuatorer er aktuelle for robot
5.2 Fo	rslag til undervisningsopplegg
5.2.1	Hensikt:
5.2.2	Enkelt eller avansert oppdrag
5.2.3	Anvendelse
Byggebes	skrivelse av en robot36
6.1 In	troduksjon – Betraktninger om realisering ("Tinkering")
	C1 · · · · · · · · · · · · · · · · · · ·
6.1.1	Chassie
	Innlednin         1.1       M         Litt tekn         2.1       Ka         2.2       Ba         Program         3.1       Øx         3.1.1 $3.1.2$ $3.1.3$ 3.2         3.2       Ti $3.2.1$ $3.2.1$ $3.2.1$ $3.2.1$ $3.2.1$ $3.2.1$ $3.2.1$ $3.2.2$ $3.3$ Ju         Den mag $4.1$ $4.2$ Ur $4.3$ Fr $4.4$ Pr $4.4$ Pr $5.1$ Im $5.1$ Im $5.1$ Fr $5.2$ Fo $5.2$ Fo $5.2.3$ Byggebes $6.1$ Im

# <u>?)]</u>

6.1	3 Servoer	41
6.1	4 Tegnefunksjon	42
6.1	5 Batterier	45
6.1	6 Kantkontakt- og Micro:bit kort	46
6.1	7 Alternative sensorer	50
6.2	Detaljerte byggebeskrivelser	52
6.2	Byggebeskrivelse for Kodegenets Servo:bot-kort for Micro:bit	52
6.3	Γest av kortet og servoene	57
6.4	Byggebeskrivelse robot	50
6.4	l Byggebeskrivelse – Grunnenhet med chassie	51
6.4	2 Byggebeskrivelse – Tegneenhet	56
6.5	Programmering av Micro:bit roboten	58
6.5	1 Styring av roboten	59
6.5	2 Programmering av senderenheten	70
6.5	3 Programmering av mottakerenheten	71
6.5	4 Program for styring av penna	71
6.5	5 Kombinert program med kjørekontroll og penn	74
Vedlegg A	Komponentliste	76
A.1	Komponent og innkjøpsliste	76
Vedlegg I	Løsningsforslag øvingsoppgaver	77
B.1	Løsningsforslag øving 1 – Lag en animasjon	77
B.2	Løsningsforslag øving 2 – Bruk av variabler	78
B.3	Løsningsforslag øving 3 – Bruk av akselerometer	78
B.4	Øving 4 Overføring av informasjon via radio	79
B.5	Løsningsforslag skritteller	30
B.6	Løsningsforslag lysteremin	30
B.7	Løsningsforslag til oppgave 7 - Lag kompass som peker mot nord	30
B.8	Løsningsforslag til oppgave 8 - Når Micro:bit "forstår" hva som blir sagt	33
B.9	Styring av 360° servo	33
Vedlegg (	OpenSCAD kode	35
C.1	Holder for nesehjul	35
C.2	Deler for løfting av penn	35
Vedlegg I	Læreplaner	87



D.1	Teknologi og Forskningslære 1	87
D.2	Teknologi og forskningslære 2	88
D.3	Teknologi i Praksis (TiP) – ungdomsskolen	89
D.4	Programmering – ungdomsskolen	90
Vedlegg	E Filer for laserkutter	93
E.1	Chassie for robot	93
E.2	Magisk terning	94



# 1 Innledning

Micro:bit er blitt en svært populær krets for å komme raskt i gang med programmering og å realisere ideer. Årsaken til dette er flere, men de viktigste grunnene er at kretsen er utstyrt med sensorer og et enkelt diodedisplay, den kan kommunisere trådløst med andre micro:bits og den kan programmeres med ulike programmeringsverktøy fra blokkprogrammering til Java og Python. Den kan også monteres i en sokkel (kantkontakt) slik at man kan få tilgang til mange av Micro:bits porter. Den er derfor lett å komme i gang med, men gir samtidig store muligheter.



Kretsen har dessuten et 3 akse akselerometer og magnetometer slik at den kan detektere helningen til Micro:bit-kortet. Den kan dessuten fungere som kompass. Videre kan diode-displayet brukes som lysdetektorer.

#### 1.1 Micro:bit og roboter

På markedet finnes det en mengde roboter bygget omkring Micro:bit, ikke minst roboter laget av enkle materialer. Her er noen eksempler som er i salg.

Bildet under viser en roboter bygget opp av meget billige materialer.



Artbot

Roombot

Golfbot



Artbot kan holde en pen og tegne på et papir mens den beveger seg. Roombot har en kollisjonsbryter foran laget av aluminiumsfolie som gjør den i stand til å detektere sammenstøt og på den måten snu og styre unna. Golfbot detekterer en ball som ruller inn i skuffen foran. Når ballen legger seg i skuffen sluttes en krets gjennom aluminiumsfolien<sup>1</sup>. Det fine med disse er at barna selv kan utforme og videreutvikle roboten selv med hjelpemidler de finner hjemme. Det forutsettes imidlertid at de har en Micro:bit. Pris for settet: ca. 20£.

#### Kitronik Micro:bit buggy

Kitronik leverer også en linjefølgende robot: *Micro:bit buggy*<sup>2</sup>. Denne leveres om byggesett og krever litt lodding. Den er basert på et DC-motor driverkort med sokkel for Micro:bit-kortet. To DC-motorer med utveksling driver hjulene. Tre lyssensorer er festet foran på undersiden og fungerer som sensor for følging av en svart linje.

Pris € 32 Micro:bit-kort ikke inkludert.

#### 4:tronix Robo:Bit Buggy<sup>3</sup>

har laget en interessant robotvariant med Micro:bit. Denne anvender akselerometeret for å detektere kollisjoner. Dessuten kan den styres trådløst via en annen Micro:bit. Med tillegg i prisen kan den utstyres med ultralyd avstandssensor og sensorer for å følge en linje.



Pris: £ 22,- hos Rapid electronics

#### 4:tronix Bit:Bot<sup>4</sup>



Denne roboten er basert på standard DC-motorer med utveksling som forsynes med 3 x AA batterier. 12 lysdioder er montert på armer på hver side av roboten, og to digitale sensorer for å følge en linje er montert under foran. En lydgiver (buzzer) kan programmeres til å gi lyd. Det følger dessuten med en innretning for montering av en penn. Roboten kan dessuten utstyres med flere sensorer. En ulempe er at pennen er plassert så langt bak, dette gjør det utfordrende å lage programmer for å tegne spesifikke tegninger. Pris: £29.75 + MVA

Dette er et lite utvalg av roboter som finnes på markedet og som benytter Micro:bit. Hva kan så vår robot tilføre dette mangfoldet?

<sup>1.</sup> https://www.techwillsaveus.com/shop/microbit-microbot-pack/

<sup>2.</sup> https://www.stuff.tv/features/how-masterthe-bbc-microbit/master-1-microbit-crane?\_format=amp

<sup>3.</sup> https://www.rapidonline.com/4tronix-robo-bit-buggy-for-bbc-micro-bit-75-0123

<sup>4.</sup> https://www.rapidonline.com/pdf/75-0117\_v1.pdf

https://www.rapidonline.com/4tronix-robo-bit-buggy-for-bbc-micro-bit-75-0123



# 2 Litt teknisk småplukk om Micro:bit

I dette kapittelet skal vi se nærmere på Micro:bit-kretsen og løfte fram noen aktuelle programmeringsblokker. Det er ikke tanken at dette skal være noen komplett oversikt over blokkoding av Micro:bit.

#### 2.1 Kantkontakten

Ved tilkobling av kantkontakten til Micro:bit får man tilgang til et langt større utvalg av porter, både inn- og utganger og analoge og digitale porter. Figuren under viser en oversikt over kantkontakten.



Som det framgår av figuren så har kretsen 20 porter hvorav 6 er analoge inn- eller utganger (P0, 1, 2, 3, 4 og 10), hvorav P0, 1 og 2 også kan brukes som berøringssensorer. De resterende 15 digitale portene (P5, 6, 7, 8, 9, 11, 12, 14, 15, 16, 17, 18, 19, 20) kan settes opp som inn- eller utganger etter behov så fremt de ikke er reservert til spesielle oppgaver som f.eks. P12 som er reservert til annet formål. Dessuten er P3, 4, 6, 7, 9, 10 brukt til å styre displayet, men kan frigjøres til andre formål etter behov. Vi legger også merke til at P3, 4 og 10, kombineres med analoge innganger. Hvilket betyr at lysdiodene også kan fungere som lyssensorer. Knappene A og B er koblet til P5 og P11.

I tillegg ser vi at drivspenningen er spesifisert til 3,3 V, men Micro:bit kan også kjøres på 3 V (dvs. ett CR2032 eller 2 x AA eller 2 AAA). Forsyningsspenningen er tilknyttet tre pinner. Dessuten er tre pinner koblet til jord. Selv om mange sensorer og *break out kort*<sup>5</sup> anvender 3,3 V, så er det også



mange som trenger en arbeidsspenning på 5 V. Den roboten vi skal bygge har en 6 V kilde som kan senkes til 3,3 V om nødvendig.

Pinne P19 (SCL) og P20 (SDA) kan også brukes til kommunikasjon via seriebuss ( $I^2C$ ). Dette er en særdeles anvendelig seriekommunikasjonslinje for å kommunisere med andre digitale sensorer o.l. Det er også gitt mulighet for trelinje SPI-buss (P13, 14 og 15).

Fem av tilkoblingspunktene er utvidet og gjennomhullet slik at det skal være mulig å koble til bananstikkere eller krokodilleklemmer. Dette gjelder batterispenningen 3V (+ og -)og portene P0, P1 og P2. Dette gjør at en kan komme igang svært fort med et minimum av tilleggsutstyr. Samtidig som det ligger et betydelig utviklingspotensial i kretsen.





Det finnes kantkontakter som passer til Micro:bit og som selges til en overkommelig pris (typ. £4 montert på kretskort og dobbel stiftlist). Figuren over til venstre viser en kantkontakt uten kretskort. Bildet til høyre viser en kantkort og kretskort som kan utstyret med dobbel stiftlist.

For den som ønsker å gå videre og utforske mulighetene til kretsen, kan en anskaffe et Inventors kit som gjør det lett å koble til eksterne komponenter på et koblingsbrett.

I tillegg til byggeplata inneholder settet jumpere og en rekke sensorer og aktuatorer, men selges uten Micro:bit så det må kjøpes separat. Pris i Norge ca. kr. 500,- inkl. mva. Mens Micro:bit koster ca. kr. 250,- inkl. mva.



5. Små kretskort med sensorer eller andre elektriske komponenter, påmontert kontakter for tilkobling til andre elektroniske kretser



Nylig er det blitt utviklet et kort med kantkontakt av Joachim Hagen Skeie som driver Kodegenet AS. Kortet har fått navnet Micro:bit Servo:bot, og er spesielt laget for å kunne brukes for å bygge små roboter. Kortet kan fås som byggesett eller ferdig bygget.

Det er forberedt for å kunne koble til 4 servoer, avstandssensor og to Neopixler (fargete LED). Vi skal bruke kortet til vår robot.



#### 2.2 Batteripakke for Micro:bit

Dersom Micro:bit skal brukes uten å være tilkoblet PC, er det praktisk å montere den på et batteridekk (Power Back) som inneholder batteri (CR2032 - 3V), lydgiver (piezo elektrisk) og bryter. Dekket monteres til Micro:bit med tre maskinskruer med mutter og avstandsstykker.





# **3** Programmering av Micro:bit – øvingsoppgaver<sup>6</sup>

Siden blokkprogrammering for Micro:bit er ganske intuitiv, spesielt for de som har litt erfaring med programmering fra før, er det kun nødvendig med en meget enkel introduksjon.

#### 3.1 Øvingsopplegg

Programmeringsverktøyet finnes på nett på følgende adresse:

Åpne gjerne programmet i Edge eller Internett Explorer<sup>7</sup>

#### https://makecode.microbit.org/#

#### 3.1.1 Øvingsoppgaver – grunnleggende

Følgende oppgaver gi en grunnleggende innføring i programmering av Micro:bit. Siktemålet er hele tiden å legge grunnlaget for å kunne forstå programmet som skal bruke til å styre roboten med, både på sender- og mottakersiden.

1. **Oppgave 1:** Lag en animasjon som beveger seg over displayet. Bruk minst tre bilder.

Tips:



Merk at fargen på blokkene viser hvilken meny de tilhører.

#### 2. Oppgave 2A: Bruk av variabler

Lag et program som øker et tall på displayet med 1 hver gang du trykker på knapp A



#### 3. Oppgave 2B

Lag et program som:

- ... viser økende antall på displayet for hvert trykk på knapp A og
- ... viser minkende antall på displayet for hvert trykk på knappB og
- ... nullstilles når begge knappene trykkes samtidig
- 6. Bygger på erfaringer som Jon Haavie fra Teknoteket ved Teknisk museum i Oslo

<sup>7.</sup> Chrome kan oppføre seg noe annerledes enn Internet Explorer



Tips:



4. **Oppgave 3A:** Lag et program som viser navnet til den ene deltageren på displayet når knapp A trykkes, og som viser navnet til den andre når knapp B trykkes.

Tips:



- 5. **Oppgave 3B:** Hva kan du gjøre for å vise mer enn to forskjellige navn?
- 6. **Oppgave 3C:** Bruk akselerometeret for å vise fire forskjellige navn avhengig av hvordan du holder Micro:biten

Tips:





7. **Oppgave 4A:** *Micro:bit kan overføre informasjon fra en Micro:bit til en annen. Lag et program som overfører informasjon om hvilken knapp som trykkes på den ene Micro:biten til den andre, og samtidig vises på displayet til den andre.* 

Tips sender:



#### 3.1.2 Forslag til andre øvingsoppgaver

# 8. Oppgave 5A: Lag en skritt-teller som teller hvor mange skritt som tas og viser summen av skritt på forespørsel.

TIPS: Forsøk å utnytt at akselerometeret kan brukes til å finne Micro:bits retning i rommet Hvem lager den mest pålitelige skrittelleren?

Tips:



9. **Oppgave 5B:** *La Micro:bit lage en trudelutt når den har talt 10 skritt.* Tips:





10. **Oppgave 6:** *Lag en lysteremin med en tone som varierer med lystyrken på lysdiodene* Tips:



11. Oppgave 7A: Lag et kompass som peker mot nord

Tips:



Lim neodym magneter på undersiden av bordet og bruk Micro:bit til å finne posisjonen til magnetene slik de framstår på oversiden av bordet.

12. **Oppgave 8:** Når Micro:bit "forstår" hva som blir sagt La en forsøksperson velge et tall fra 1 til 4 Legg Micro:bit med displayet ned Når Micro:biten snus rundt skal displayet vise det tallet som forsøkspersonen har sagt Det er du som snur Micro:bit'en

#### 3.1.3 Oppgaver knyttet til styring av servoer

I de etterfølgende oppgaver kreves to 360° servoer og kortet Servo:bot

#### 13. Oppgave 9A Programmering av 360° servo - Styring

Lag et program som kjører høyre servo forover når knapp A trykkes og venstre servo forover når knapp B trykkes. Når begge trykkes stopper begge motorene.

Tips:



P0 - høyre servo

P1 - venstre servo



Pulslengde 1525 – 1800, servo går framover, fart bestemmes av pulslengde Pulslengde 1475 – 1200, servo går bakover, fart bestemt av pulslengde Pulslengde 1475 – 1525, servo står i ro

#### 14. Oppgave 9B Programmering av 360° servo - Styring

Lag et program som kjører høyre servo framover når knapp A trykkes første gang og bakover når knapp A trykkes 2. gang, og venstre servo forover når knapp B trykkes første gang og bakover når knapp B trykkes 2. gang. Når både A og B trykkes samtidig, stopper begge motorene.

Tips:



Sett flaggene innledningsvis til 0



Velg er en av to aksjoner avhengig av hva som ble gjort sist gang

For å la samme bryter gi to forskjellige responser for annet hvert trykk kan man benytte "flagg". Et flagg er en merkelapp som kan slås av og på ("heises" eller "fires"), avhengig av flaggets tilstand velges en av to responser.

#### 15. Oppgave 9C Programmering av 360° servo - Fjernstyring

Lag en fjernstyringsprogram slik at en servoene kan styres som i oppgave A, men via en radioforbindelse med en Micro:bit nr. 2.

#### 3.2 Tips til blokkoding med Micro:bit

#### 3.2.1 Programvare, arbeidsflaten og lagring av prosjektfiler

Micro:bit kan kodes på ulike måter bl.a. ved hjelp av Python, Java eller blokkoding basert på Java. Vi skal ikke her gi noen fullstendig innføring i blokkoding, men bare peke på noen sentrale blokker som kan være nyttige for prosjektene i dette heftet. Generelle kommandoer i blokkoding er ganske like mellom de ulike blokkodingsverktøyene.



La oss imidlertid først se litt på programmeringsvinduet som er web-basert og finnes på følgende nettadresse: https://makecode.microbit.org/. Går vi til adressen kommer følgende vindu opp:

Last opp	o prosjekt		
😁 micro:bit 🛸 Projects < Sha		E Blocks () JavaScript	🤨 🂠 👫 Microsoft
Simulator Last fil til PC eller Micro:bit	III Basic III Basic Input Music Led Aradio C Loops X Logic Variables Math Advanced Advanced Arays T Text Game I Text Game	Velg mellom blokk eller Java rep 2. Blokkmeny 3. Arbe 1. Gi prosjektet et navn	presentasjon vidsflate Forstørr eller forminsk "Undo" og "Redo"
Download	Robot sender 3-0	a	

Her er en kort beskrivelse av gangen i arbeidet:

Vi anbefaler å bruke Edge eller Eksplorer, de fleste andre kan også brukes, men kan gi litt varierende prosedyrer for lagring.

#### 1. Opprett et nytt prosjekt

Start med å skriv inn et navn på prosjektet (	Magi	Magisk terning sender				
boksen (vist til høyre) og det kommer opp e	en menylinje:					

Velg "hake opp" til høyre for "Lagre" og få fram "Lagre som". Lagre fila i en ønsket katalog. Du får nå opp følgende:

microbit-Test-for-lagring.hex er lastet ned.	Åpne	Åpne mappe	Vis nedlastinger	×



Velg "Åpne mappe" og finn igjen den lagrede fila i mappa, Den har nå fått endingen hex

I Victoria						-	
Fest till Hurtigtilgang Ublipptavle	Kopier til * Slett Organiser	Gi nytt navn	Nyt element •	Egenskaper	ane - H Merk alt diger Merk ingent egg Merk ingent Werker utw Velg	ting	
← → → ↑ 📜 > Denne PCen > DATA (D:) >	Microbit > Test				~ 0	Søk i Test	,
Musikk	^ _	Navn	A		Endringsdato	Туре	Større
🐱 Nedlastinger		microbit-Getting-started.hex		09.07.2017 22.18	HEX-fil		
Skrivebord		microbit-LED-1.hex 10		10.07.2017 21.16	HEX-fil		
Videoer		mic	robit-Magisk-terning (5).h	ex	22.07.2017 22.02	HEX-fil	
😂 OS (C:)		mic	robit-Magisk-terning-mot	taker (7).hex	22.07.2017 23.16	HEX-fil	
_ DATA (D:)		📄 mic	robit-Magisk-terning-mot	taker-risting (5)	24.07.2017 21.01	HEX-fil	
MICROBIT (E:)		📄 mic	robit-Radio-1.hex		18.07.2017 22.23	HEX-fil	
see plu (\\luna syt ntnu no) (I.1		🗌 mic	robit-Radio-2.hex		19.07.2017 21.02	HEX-fil	
- nikra (lihoma socatt atau no) (M)		mic	robit-Servo (4).hex		17.07.2017 21.23	HEX-fil	
white (((nome.ansatt.ntnu.no) (w.)	<u> </u>	mic	robit-Servo (5).hex		08.08.2017 21.44	HEX-fil	
progaist (\\progaist.ntnu.no) (P:)		mie	robit-Servo0.hex	_	08.08.2017 21.45	HEX-fil	
🛫 ilu (\\felles.ansatt.ntnu.no\ntnu\su) (T:)		C 🗋 mic	robit-Test-for-lagring.hex	$\rightarrow$	18.08.2017 11.34	HEX-fil	
Felles (\\trigger.vitensenteret.com) (Y:)							
See filer (\\129.241.149.61) (Z:)	~ <						>
11 elementer							111 -

Bruk musa og dra hex-fila over til katalogen "MICROBIT" i lista til venstre, dermed programmeres Micro:biten som er tilkoblet PCen. Det nye programmet skriver automatisk over ev. gamle programmer.

Du kan også benytte "Lagre som" kommandoen og lagre fila rett i Micro:bit, det kan være praktisk ved uttesting. Du må da være klar over at programmet ikke lagres og at det ikke uten videre er mulig å hente programmet fra Micro:bit og legge det i en katalog.

#### 2. Skriv programmet

Ved å velge blant fanene i blokkmenyen får du opp lister over blokkommandoer. Disse dras fram av menyen og settes sammen på Arbeidsflata til de blir det ønskede programmet som i eksempelet vist på figuren under.

💿 micro:bit 🖕 Projects 🔫 s	Share		📩 Blocks	{} JavaScrip				0	٠	<b>*</b>	licrosoft
	Saint. Q		1.1	2.5	5. 5. 5		5.0		12	Getting	Started
	III Basic										
	⊙ Input		on start		_						
	O Music		_d radio	set group 🛙	1						
·u	C Led		III forever								
	al Radio		a radi	send value	acc_y 11	-( 0 =	celeration (m	c) (VIII)			1.1
0 $0$ $0$ $0$ $0$	C Loops		III paus	(=s) 0 S0	1						1.1
0 1 2 3V GND	🔀 Logic		al rada	send value	··· acc_x ···	- 1 💿 #	celeration (m	e) 🖽			
	Variables		III paus	e (es) ( 50							10.00
	Math										10.04
₽ <b>*</b>	Advanced										10.04
·0	i≡ Arrays										
4	T Text										
0 0 0 0 0	co Game										
	Images								_	_	
🗉 📥 Download	Robot sender 3-0	8								9.0	• •



Legg spesielt merke til at simulatoren viser to Micro:bits til venstre. Dette skyldes at akkurat dette programmet spesifiserer radiokontakt med en annen Micro:bit, dermed trengs det minst to for å simulere funksjonen til programmet. Det kan være praktisk å bruke simulatoren for å teste ut programmer før man overfører til den fysiske Micro:biten.

3. *Lagre og overfør programmet* som vist foran under punkt 1. Det kan være verd å merke seg at ved kun å velge "Lagre" og ikke "Lagre som" *så vil filene automatisk havne i mappa "Nedlasting*". Her vil filene få endingen (2), (3), (4) osv. etter som nye versjoner skrives og lagres. Dette kan være en fordel dersom man vil lagre tidligere versjoner.

Ønsker man opplæring går man til den brune fanen øverst i høyre hjørne: "Getting started" så får man tilbud om et øvingsopplegg.

#### 3.2.2 Noen blokker

Her skal vi se på noen blokker i et par undermenyer som er litt spesielle for Micro:bit og nyttige for de formål vi skal bruke dem til.

#### Bruk av variabler

Når man programmerer med blokkode så har man to typer variable, det er *heltall* og *tekst*. Hvilken som blir hva bestemmes av om man fyller en variabel med tall eller tekststrenger. Det er ikke nødvendig å deklarere variable innledningsvis. Men trenger heller ikke innledningsvis å sette en variabel til 0 eller en tekst-variabel til "ingen tekst", det vil programmet gjøre automatisk ved oppstart.



Tilordning av variabler er vist i figuren over til høyre. Her settes variabelen "variabel" til verdien 0, dvs. at variabelen "variabel" vil være en tall-variabel hvor det ikke er mulig å skrive inn en bokstav eller tekst. Variabelen "name" deri-

mot settes imidlertid til en tekst her lik "HEI" og vil dermed bli en tekstvariabel (tekststreng), og kan da ikke senere i programmet tilordnes et tall. Gjør man det, får man en advarsel som vist nederst på figuren over.

#### Endre variabelnavn

Det anbefales ikke å bruke de navnene som automatisk kommer opp ved når man trykker på pilen til høyre for "variabel".

Velger man menyen "Variabler" får man opp listen av "default" variabler (variabel, name, item osv.) i tillegg til de man har definert selv (berøring). Ved å velge "Lag en variabel"

C Skjerm	Lag en variabel
I Radio	W variabel •
C Løkker	berøning •
🔀 Logikk	list v
Variabler	name •

(øverst i lista, kan man definere egne variabler. Dette anbefales sterkt. Bruk navn som er forståelige og umiddelbart gir mening for den som leser programmet. Disse vil da legges til lista.



#### Akselerometer - Under katalogen Input

Denne variabelen inneholder løpende avlest verdi av akselerometeret i Micro:bit. Akselerometeret måler akselerasjon i alle tre akseretninger x-, y- og z-retningen og leverer en verdi lik  $\pm$  2048 med benevningen milli-g (mg), hvor g = 9.81 m/s<sup>2</sup>. Dvs. at med denne innstillingen (område satt lik 2g) kan Micro:bit måle verdier fra – 2 g til + 2 g. Området for akselerometeret kan sette til ett av fire områder, se under for nærmere beskrivelse.

Et akselerometer som ligger i ro eller beveger seg i en konstant rettlinjet bevegelse, vil her på jorda kun påvirkes av gravitasjonen. Når Micro:bit ligger flatt på bordet vil vi ideelt sett kun få en verdi i z-retning  $\operatorname{acc} z = -$ 1000 milli-g = -1000 mg = -1 g. Verdien er negativ fordi den positive retningen til z-aksen er rettet oppover fra Micro:bit, mens gravitasjonen vir-



ker nedover. Figuren over til høyre<sup>8</sup> viser hvordan aksene til akselerometeret er definert.

Følgende bevegelser vil gi ulike verdier for x-, y- og z-retningene:

Handling	x-retning	y-retning	z-retning
Flatt på bordet, display opp	0 mg	0 mg	- 1000 mg
Flatt på bordet, display ned	0 mg	0 mg	+ 1000 mg
Tilting mot høyre om y-akse, display opp	økende positiv verdi	0 mg	minkende negativ verdi
Tilting mot venstre om y-akse, display opp	økende negativ verdi	0 mg	minkende negativ verdi
Tilting framover om x-aksen, display opp	0 mg	økende positiv verdi	minkende negativ verdi
Tilting bakover om x-aksen, display opp	0 mg	økende negativ verdi	minkende negativ verdi

Med andre ord så kan akselerometeret fortelle oss orienteringen av Micro:bit i rommet. Følgende tre blokkvariabler gjør det mulig å avlese de tre verdiene.

Disse er avleste verdier som kan tilordnes variabler som vist i eksempelet under.



Her tilordnes akselerometerverdien i x-retning til variabelen "x".

Men akselerasjon kan også skje på andre måter enn ved tyngdeakselerasjonen. For eksempel ved at Micro:bit gis en akselererende bevegelse, dvs. at farten øker eller minker.

<sup>8.</sup> http://microbit-challenges.readthedocs.io/en/latest/tutorials/accelerometer.html



Kommandoen "Set accelerometer range" kan måleområdet for akselerometeret endres til verdiene: 1, 2, 4 eller 8 g. Avlesningen vil alltid være i milli-g (mg) slik at verdien kan gå opp til  $\pm$  8000 mg.

⊙ set accelerometer range 1g 🔽

#### Radio - Under katalogen Radio

Denne skal vi bruke for å opprette radioforbindelse mellom to eller flere Micro:bits. La oss anta at det er ønskelig å opprette en radioforbindelse mellom to Micro:bits. Det skal overføres et tallverdi, og en tekst som kan knyttes til dette tallet. Dermed er det mulig å overføre forskjellige parametere (verdier) og



det er fortsatt mulig å skille verdiene fra hverandre ved hjelp av den "medsendte" tekststrengen.

- 1. I "on start" funksjonen bestemmes hvilken kanal vi ønsker å opprette radioforbindelsen i ("radio set group ch"). I alt kan man velge mellom 256 kanaler fra 0 255.
- 2. Dernest må man velge et navn og en parameter man ønsker å sende:



Figuren under viser et eksempel der vi sender to parametere knyttet til hver av parametrene "acc\_x" og "acc\_y".

forever forever		4 4		4	1		
radio send v	alue 🕻 🐇	acc_y >>	= ( 🖸	accel	leration	(mg) y	
🏭 pause (ms) 🖡	50	8 4		+	* +	÷	Ŧ
radio send v	alue 🕻 "	acc_x "	= (	accel	leration	(mg) 🗙	
🎞 pause (ms) 🕻	50						

Her ser vi at vi avleser verdien til akselerometeret i x-retning med blokken "acceleration (mg) z) og sender verdien sammen med navnet acc x. Til-

⊙ acceleration (mg) Z ▼

svarende sender vi akselerometerverdien i y-retning under navnet acc\_y. La oss se hvordan disse kan skilles igjen på mottakersiden.

 Også mottakeren må settes opp med riktig kanal. Dette gjøres på samme måte som på sendersiden i "on start" blokken..





4. Vi har ulike kommandoer som kan brukes til å motta og skille ut de ulike parametrene. Under ser vi alle de som ligger i menyen. I virkeligheten er alle disse den samme funksjonen, man har bare valgt ut litt forskjellige parametere.





Dersom vi trykker på det vesle hjulet i venstre hjørne får vi opp samtlige valgmuligheter som vist i figuren under.

Her er en komplett oversikt<sup>9</sup> over parametere som kan velges inn i datapakken som sendes over radiokanalen (Blue tooth):

*receivedNumber:* Er en variabel som inneholder den *verdien* som er knyttet til datapakken (akselerometerverdien i eksempelet foran)

*receivedString:* Er teksten som er knyttet til datapakken. Det kan også være navnet til variabelen viss verdi ønskes overført.

*time:* Er systemtidspunktet til Micro:bit som sendte meldingen, dvs. tid fra påslag. *serial:* Serienummeret til den Micro:bit som sendte meldingen, om den er merket med et serienummer fra fabrikken.

*signal:* Angir signalstyrken til radiosignalet, fra en svakeste verdi på -128 til en sterkeste verdi på -42.

Vi velger å ta med variablene "receivedNumber" og "receivedString" og får følgende argumenter i funksjonen. Her knytter vi "receivedString" ("default" navn) til parameterens navn f.eks. acc\_y og receivedNumber" til en verdi mellom +/-0 - 1000 som angir akselerometerverdien.



<sup>9.</sup> https://makecode.microbit.org/reference/radio/on-data-packet-received



5. I figuren under ser vi hvordan vi anvender innholdet i receivedString for å skille de to variablene x\_acc og y\_acc.



Ved hjelp av betingelsen til if-setningen sjekkes hvilke av de to medfølgende tekststrengene sendingen gjelder slik at de to tilhørende verdiene kan tilordnes de riktige variablene.

6. Dernest kan vi la det etterfølgende programmet bruke de overførte verdiene som nå er tilordnet variabler.

I stedet for å bruke "receivedString" og "receivedNumber" kan definere sine egene variabler som gir mer mening.

#### Servo - Under katalogen Pins

Denne skal vi bruke for å styre ulike servoer, både 180° og 360° servoer.

Hos en 180° servo kan akslingen dreies 0 - 180°. Slike egner seg for å dreie en arm eller dra i en sleide. En 360° servo roterer kontinuerlig 360° rundt og vil fungere som en motor. Slike egner seg til å kjøre en robot bortover gulvet. Figuren til høyre viser oppkoblingen for en 180° servo. Siden vi bruker P0 som styreutgang kan vi bruke en bananstikker eller en krokodilleklemme.

Vi skal her se nærmere både på blokker som kan styre ut 180° og 360° servoer. Disse blokkene finner vi under blokkfamilien "Pins".

Blokken vist under er primært for styring av en 180° servo.





Her velger vi hvilken pinne som vi anvender til styring og hvor mange grader vi vil at akslingen skal dreie. I eksempelet over vil servoen som er koblet til port P0, dreie 180°.



Når man trykker den vesle pila ved siden av pinnenummeret, vil man få opp en komplett oversikt over de portene som er valgbare.

P0	P1	P2	P3
P4	P10	P5 (write only)	P6 (write only)
P7 (write only)	P8 (write only)	P9 (write only)	P11 (write only)
P12 (write only)	P13 (write only)	P14 (write only)	P15 (write only)
P16 (write only)	P19 (write only)	P20 (write only)	

Skal vi styre en 360° servo så bruker vi kommandoen:



Her velger vi hvilken pinne som vi ønsker å bruke til styring og hvor fort servoen skal rotere. Blokken setter opp den valgte porten (f.eks. P0) som en analog utgang som sender et pulstog med pulser som har en lengde gitt i mikrosekunder (f.eks. 1500µS). Ved å trykke pila ved siden av P0, så vil man få opp alle de mulig valgbare utgangene.

360° servoer er ofte konstruert slik at de står i ro for pulslengder på omkring 1500 μs, går med full fart framover ved en pulslengde på 1800 μs og full fart bakover med en pulslengde på ca. 1200 μs.

#### 3.3 Justering av 360 servoer<sup>10</sup>

Det finnes en rekke ulike 360° servoer på markedet. Vi har valgt å bruke *FS90R Continuos servo*. Denne har en liten trimmeskrue på baksiden som gjør det mulig å justere nullpunktet for servoen. Dvs. justere den slik at servoen står stille i et 45µs vindu omkring 1500 µs pulslengden.



<sup>10.</sup>https://www.pololu.com/product/2820



## 4 Den magiske terningen - en innledende oppgave

I denne oppgaven skal vi trene på programmering av noen sentrale egenskaper med Micro:bit:

- 1. Bruk av akselerometeret
- 2. Bruk av radiokommunikasjon
- 3. Bruk if-setninger og betingelser

I tillegg skal vi bruke:

4. Tilfeldighetsfunksjonen

#### 4.1 Problemstilling

Tenk deg følgende: Du har en tradisjonell sekskantet terning med øyne og en elektronisk terning laget ved hjelp av en Micro:bit.



Når man rister på den elektroniske terningen vil den gi tilfeldige antall øyne fra 1 til 6 på displayet.

Så blir man bedt om å trykke inn trykknapp B og holde denne inne et par sekunder. Så rister man på nytt. Det som nå skjer er at resultatet av ristingen gir samme antall øyne hele tiden, nemlig det antall øyne som den tradisjonelle terningen viser. Helt konsekvent.





Dersom man snur den tradisjonelle terningen og rister den elektroniske, så vil den vise den nye verdien. Slik er det hver gang, den elektroniske terningen ser ut til å kopiere den tradisjonelle terningen.

Forsøk å analyser problemstillingen og kom opp med ulike forslag til hvordan dette er mulig.

#### 4.2 Undervisningsopplegg

Vis frem terningen å la elevene utføre eksperimentet som beskrevet foran:

La dem eksperimentere med terningene og se hva som skjer:

- La dem diskutere ulike løsninger for hvordan dette kan være mulig. Det vil selvfølgelig være stor forskjell for elever som er ukjent med, og de som er kjente med muligheten i Micro:bit. Oppgaven vil sannsynligvis passe best for de som har litt kjennskap til Micro:bit på forhånd.
- Når de har avslørt hemmelighet, la dem beskrive funksjonen til programmene med ord. Utfordre dem så til å lage en grov skisse av et flytdiagram for programmene. Diskuter løsningene med dem på grunnlag av flytdiagrammene.
- Be dem sette opp et forslag til hvordan de vil gå fram for å designe terningene. Planen bør inneholde en gradvis oppbygging av programmet med milepæler som inkluderer testing av programmet med hårdvare.
- Har de tilgang til en laserkutter kan de jo også designe den tradisjonelle terningen.

#### Forslag til plan for programmering

Test programmet i Micro:bit mellom hvert av punktene. Oppgaven krever at man har to stykk Micro:bit: Micro:bit 1 (mottaker) og Micro:bit 2 (sender) og at hver av dem er utstyrt med batteri.

Det kan være lurt å lage flere testprogrammer for å teste ut ulike funksjoner om man er ukjent med hvordan funksjonene fungerer.

#### Lag en riste-terning

- Lag en elektronisk terning (Micro:bit 1) som skriver ut tilfeldige antall øyne når man trykker på A-knappen Tips 1: Bruk "on button A pressed" under *Input* Tips 2: Bruk "Pick a random ..." under *Math*
- 2. La terningen bli styrt av risting. Terningen gir en ny verdi hver gang man rister Micro:bit Tips: Bruk "on shake …" under *Input*

#### Lag en terning som avhenger av orienteringen i rommet

 Lag en terning (Micro:bit 2) som viser ulike antall øyne avhengig av hvilken side som peker opp. Bruk en tradisjonell terning til å bestemme hvordan øynene skal plasseres. Tips: Bruk "acceleration (mg)" under *Input*



#### Opprett radioforbindelse

- 4. Sett opp radiokanalen mellom Micro:bit 2 og Micro:bit 1 Tips: Bruk "radio set group" under *Radio*
- 5. Legg inn en send-kommando i Micro:bit 2 som sender terningverdien Tips: Bruk "radio send <name> value" under *Radio*
- 6. Lag et nytt program for Micro:bit 1 som mottar og viser terning-verdien fra Micro:bit 2 Tips: Bruk "on radio received <mottatt verdi>" under *Radio*

#### Lag fjernstyrt terning

 Kombiner programmet som mottar terning-verdier fra Micro:bit 2 med den tilfeldige terningfunksjonen. Bruk A-knappen og B-knappen til å gå mellom vanlig terning til radiostyrt terning (trykk B-knappen) og tilbake (trykk A-knappen). Tips 1: Bruk "if than" og "if than else" under *Logic* Tips 2: Bruk flagg-variabler for å huske siste trykk på bryterne



#### 4.3 Framstilling av den tradisjonelle terningen

Terningen kan lett framstilles ved å bruke et av mange boksprogrammer. Det kan være praktisk å bruk et program som gir mulighet til å angi innvendige mål da disse må passe til bredden til en Micro:bit.



Til dette designet har vi brukt: Maker Case: http://www.makercase.com/ og øyne er skrevet på senere i programmet FlexiDesign. Fire små pappbiter er benyttet for å lage spor til Micro:bit inne i terningen slik at den skal ligge i ro.



#### 4.4 Programmet

Programmet består av to programmer, ett for hver av de to Micro:bit. Det ene er for Micro:bit 1 som ligger inne i terningen og som skal registrere terningens posisjon og sende ut verdier. Det andre programmet skal dels kunne genere tilfeldige tallverdier fra 1-6 eller motta og vise de mottatte verdiene fra Micro:bit 2.

#### 4.4.1 **Program Micro:bit 1 (mottaker)**

Figuren under viser det blokkprogrammerte programmet.



Micro:bit - Forslag til et undervisningsopplegg 31



#### 4.4.2 Program Micro:bit 2 (sender)

Figuren under viser det blokkprogrammerte programmet for senderen. Denne detekterer retningen til terningen og sender denne til mottakeren som viser riktig terningverdi.



## 5 Micro:bit robot med tilhørende undervisningsopplegg

I dette kapittelet skal vi realisere en robot med Micro:bit. Dette er ikke ment som et endelig design, men grunnlag for videre utvikling. Vi har derfor tatt med et innledende avsnitt som diskuterer de ulike løsningene og begrunner de valg vi har gjort. I tillegg vil vi her peke på muligheter som ennå ikke er realisert i vår prototyp.

#### 5.1 Innledende betraktninger om valg av løsning

Innledningsvis skal vi diskutere ulike løsninger og gi korte begrunnelser for de valg vi har gjort uten at dette skal være noen fasit.



#### 5.1.1 Chassie

Selve ramma til roboten bør være så enkel som mulig og kunne laserkuttes i MDF, finer eller akryl.

På oversiden bør den ha plass til:

- To små servomotorer med hjul, enten innkjøpte eller laserkuttede hjul
- Kantkontakt for Micro:bit
- Et lite koblingsbrett (for oppkobling av servomotorer, ev. ultralyd avstandsmåler, lysdioder)
- Plass til avstandssensor og lysdioder

På undersiden bør den ha plass til:

- To batteripakker for to AA batterier, min. 6V
- Et nesehjul eller støtte, ev. et kulehjul, må kunne monteres foran
- Mulighet for montering av to reflektanssensorer foran for å kunne følge en sort linje

Styring og kontroll:

• Roboten bør kunne fjernstyres med en Micro:bit

#### 5.1.2 Sensorer og aktuatorer er aktuelle for robot

Følgende aktuatorer er tilgjengelig ev. må være mulig å montere:

- To 360° servomotorer, hvor fart og retning kan styres
- Et display med 5 x 5 lysdioder (på Micro:bit)
- Ev. eksterne lysdioder
- En eller to 180° servoer påmontert armer

Følgende sensorer er tilgjengelig internt på Micro:bit:

- Akselerometer, for styring fra håndholdt Micro:bit
- Magnetometer (anvendelse ikke bestemt)
- Radiokommunikasjon, kanal for fjernstyring

Følgende eksterne sensorer kan være aktuelle på roboten:

- Kollisjonssensor
- Reflektanssensorer
- Avstandssensor



#### 5.2 Forslag til undervisningsopplegg

#### 5.2.1 Hensikt:

Hensikten med undervisningsopplegget er å:

- ... tilby et litt større prosjekt knyttet til programmering av Micro:bit
- ... lage en robot med Micro:bit som kan fjernstyres gjennom en løype på kortest mulig tid.

De skal lære:

- ... hvordan en enkel robot fungerer
- ... hvordan de enkelte delene av en robot fungerer
- ... grunnleggende blokkprogrammering
- ... hvordan programmere servoer
- ... hvordan programmere fjernstyring med sving, forover og bakover med ulike hastighet
- ... å beskrive hvilke utfordringer byggingen, programmeringen og bruken av roboten ga
- ... å dokumentere hva de har gjort slik at andre kan bygge på det de har gjort

#### 5.2.2 Enkelt eller avansert oppdrag

En kan tenke seg en enkel løsning eller en krevende løsning:

Enkelt oppdrag:	Krevende oppdrag
Sett sammen roboten med delene til	Konstruer en robot med bestemte egenskaper
byggesettet.	
Installere programvare og sjekke at roboten fungerer som ønsket	Skriv kode slik at roboten er istand til å utføre et bestemt oppdrag

Det krevende oppdraget kan deles opp i mindre oppgaver med økende vanskelighetsgrad:

#### Hovedoppdrag:

Det skal konstrueres en robot som har følgende egenskaper:

- Konstruer en robot som styres av ett Micro:bit kort. Roboten skal kunne gå framover og bakover og svinge til høyre og venstre
- Robotens bevegelser skal kunne fjernstyres av et Micro:bit kortet
- Roboten skal utstyres med en mekanisme som er i stand til å ødelegg en ballong montert på en annen robot. Mekanismen skal kunne fjernstyres.

Eller ev.

• Roboten skal kunne tegne en strek mens den går. Senking og heving av pennen skal kunne fjernstyres.



Hovedoppdraget (avansert) kan deles opp i mange påfølgende oppdrag som hjelper deltagerne til å nå det endelige målet:

- 1. Finn ut hvordan en kobler sammen en Micro:bit og en 360° servo og lag et blokkdiagram
- 2. Koble en Micro:bit til en 360° servo og batterier (2x2xAA) ved hjelp av en kantkontakt
- 3. Skriv en så enkel kode som mulig for å sjekke at servoen kan gå framover og bakover, ev. at det er mulig å styre hastigheten
- 4. Konstruer roboten (chassiet) med to servoer med drivhjul og nesehjul, Micro:bit kort i kantkontakt og nødvendig batterikassetter.
- 5. Skriv en enkel kode for Micro:bit og sjekk at roboten kan gå rett fram og rett bakover. Finn ut hva som må gjøres for å kalibrere den slik at den ikke går skjevt.
- 6. Skriv ny kode slik at du kan bruke en Micro:bit nr. to som fjernstyringsenhet for den som er montert på roboten. Med fjernstyringsenheten skal du kunne starte, stoppe og kjøre forover og bakover ved hjelp av de to knappene på fjernstyrings Micro:biten
- 7. Skriv ny kode slik at du ved å helle fjernstyrings Micro:biten til høyre eller til venstre klarer å svinge roboten mot høyre og venstre. Knappene skal fortsatt kunne brukes til å kjøre fram og tilbake og til å stoppe.
- Skriv ny kode slik at du frigjør knappene og blir istand å kjøre framover, bakover, svinge til høyre og venstre ved å bikke fjernstyrings Micro:biten framover, bakover, bikke til høyre og venstre.
- 9. Skriv ny kode slik at du blir istand til å bruke knappene til å bevege armer med nåler framover og til sidene eller styre en penn.
- 10. Test ut og dokumenter det du har gjort

#### 5.2.3 Anvendelse

Deltagerne konkurrerer om å komme gjennom en løype med hinder på kortest mulig tid uten og berøre hindringene.

Eller de skal kjempe om å slå istykker ballonger på konkurrentenes roboter. En robot med ødelagt ballong er ute av konkurransen.



### 6 Byggebeskrivelse av en robot

Roboten skal være enkel og det skal brukes billige deler som skoler også har råd til å anskaffe.

#### 6.1 Introduksjon – Betraktninger om realisering ("Tinkering")

#### 6.1.1 Chassie

Figuren under viser omrisset av selve chassiet som kan skjæres ut av en laserkutter.



De to kvadratlignende klossene til høyre er ment som holder for en servo foran over nesehjulet eller som holder for en servo som kan løfte og senke en penn. Det vil framgå hvordan dette er tenkt i de neste bildene.

Chassiet kan skjæres ut i forskjellig typer materiale. Det kan være 3 eller 6 mm finer, MDF eller også akryl.

Bildet til høyre viser det utskårne chassiet i 3 mm sort MDF.




Bildet under viser en tegning av hvordan roboten kan monteres. Denne varianten er forberedt for montering av koblingsbrett og servo i fronten.



De to batteripakkene er montert på undersiden av chassiet, på innsiden av drivhjulene på undersiden av servoene. Batteriledningene kommer opp gjennom et hull i plata ved kretskortet med kantkontakt og kobles til Micro:bit og servomotorene via kontakter på kretskortet. Alternativt kan ledningene fra batteriet loddes direkte til kretskortet på undersiden.

Figuren til høyre viser et bilde av den monterte prototypen. Batterikontakten er loddet direkte til batterikontakten på undersiden. Batteriholderne er montert på undersiden under hver av servoene.





# 6.1.2 Hjul

Denne modellen har standard drivhjul som vist under.



Hjulene festes med en skrue i navet.

Vi har også prøvd å skrive ut hjul, hvilket er fult mulig. Vår erfaring er imidlertid at de lett blir noe vinglete. Bildet under viser et hjemmelaget hjul med ett hvitt bånd av ballonggummi tredd over felgen for å gi økt friksjon mot underlaget. Til høyre er vist en forminsket utgave av skjæremalen til hjulene. De to små hjulene monteres innerst ved navet og passer til akslingen på servomotoren.







Vi har forsøkt å benytte et lite leddhjul fra Clas Ohlson, hvilket synes å fungere.

Leddhjulet (25/36 mm) er ført gjennom chassie-plata og festet mellom to plater med fire treskruer, som vist på figuren under. Dette hjulet vil kreve en noe annen utforming av chassiet enn det vi har beskrevet foran.





Dersom man ønsker å lage nesehjulet selv kan man f.eks. skrive det ut på en 3D-printer. Man tar da utgangspunkt i passende klinkekule og konstruerer en holder rundt denne. I vårt tilfelle valgte vi en klinkekule med en



diameter på 15,5-16,0 mm. Vi brukte så OpenSCAD for å lage selve holderen som vist på figuren under. Bildet over viser holderen med og uten kule slik den blir seende ut i OpenSCAD. En tommelfingerregel er at diameteren til hullet bær være ca. 0,6 mm større enn kula.

OpenSCAD-koden er vedlagt i vedlegg C.1 på side 85.

Holderen til nesehjulet skrus fast med fire maskinskruer som er plassert i hjørnene i et kvadrat. Bildet til høyre viser hvordan det 3D-printede hjulet ser ut når det er montert.

Man kan også kjøpe nesehjul med kule fra f.eks. Pololu (https://www.pololu.com/category/45/pololu-ball-casters). Chassiet er også forberedt for en slik løsning. Som vist til høyre på figuren under så er det laget hull og gjort plass for montering av muttere i braketten for servoen.







En liten spalt foran nesehjulet gjør det mulig å montere en *nesestøtte* som er en langt enklere løsning enn et hjul. Denne skjæres ut av chassiet og kan presses på plass i spalten som vist på bildet under.



En slik støtte vi kunne gli bort over et jevnt underlag, men egne seg dårlig på ujevnt underlag og tepper.

#### 6.1.3 Servoer

Feste for nesehjulet kombineres på oversiden med festeanordningen for en 180° servo. Siden festeanordningen er kvadratisk, kan den settes i en av fire posisjoner 90° på hverandre, slik at en ev. servo kan slå i ulike retninger.



Åpningen i kvadratet passer eksakt til en standard 180° servo som derfor bare kan presses ned i plata. Hensikten med en 180° servo i fronten kan f.eks. være å kunne slå med en nål for å sprenge andre roboters ballonger, men kan selvfølgelig anvendes til langt fredeligere formål. De to 360° miniatyr servomotorene er presset ned i plata slik at man slipper annen festeanordning. Det er skåret ut to kvadratiske åpninger bak servoene. Dette er for at man skal komme til med trimmeverktøy for å justere "stoppområdet" for servoene dersom man velger å bruke et noe tykkere chassie som f.eks. 6 mm. For nærmere beskrivelse av justering av servoene se under avsnitt 3.3 på side 26.



## 6.1.4 Tegnefunksjon

En artig funksjon hos noen roboter er at de kan tegne en strek der de beveger seg. Dette gjøres ved hjelp av en tusj som sklir langs papiret mens roboten beveger seg. Bit:bot har løst dette med en slags "tilhenger" bak på roboten <sup>11</sup>. Denne er utstyrt med en myk gummimalje som slutter omkring tusjen. Utfordringen blir lett at pennen slipper papiret med mindre holderen fjærer slik at tusjen presses ned mot papiret. En annen ulempe med en slik løsning er at pennen ikke er sentrert om aksen til roboten. Heller ikke er det mulig å heve pennen fra papiret mens den kjører.



Det finnes et alternativ til dette hvor tusjen sklir i et rør og på den måten kan følge papiret uavhengig av bevegelsene. En slik løsning vil neppe slippe papiret.

I denne Arduino-baserte roboten glir tusjen opp og ned i et 3D-printet rør. En flens omkring tusjen gjør det mulig å heve og senke den med en 180° servo.



<sup>11.</sup> Bildene er hentet fra: https://shop.4tronix.co.uk/products/pen-holder-for-bit-bot-bitbot-or-crumblebot



Figuren under viser flensen som festes til penna og som servo-armen støter mot når tusjen skal løftes opp fra papiret.



En tusj med filtspiss er nok det beste, alternativt kan en benytte gel-tusj som inneholder en refill som kan gli opp og ned i holderen. Vi har foreslått Pilot G-TEC-C4. Ulempen med denne er at den tegner en svært tynn strek (0,4 mm), fordelen er at selve tusj-refillen lett glir opp og ned i holderen som da også kan brukes som holder ombord på roboten. Dersom metallkjeglen foran skrus av, vil man se at selve plasthylsa ender i en innsnevring av røret med gjenger som ev. kan skrus fast i chassiet.



Bildet under viser hvordan pennen kan monteres. Dette er en temmelig primitiv måte å gjøre det på, men kan være utgangspunkt for et mer hensiktsmessig design, gjerne med heving av pennen ved hjelp av en servo.





I vår neste løsning har vi valgt å 3D-printe et rør med en flens som trykkes gjennom chassiet fra undersiden. Flensen er utformet slik at ved å vri den 90° så vil flensen kiles fast under batteriholderne og holdes på plass.



Røret vil stikke ca. 20 mm opp over chassiet og er tilpasset en filttusj av typen Edding 141 F (fine). Det er viktig at røret er så romt at penna kan gli lett opp og ned i røret. En tilsvarende muffe med flens er laget for å passe rundt penna (se figuren under). Denne må være så trang at den passer akkurat rundt penna, men uten å skli opp og ned med mindre at man bruker litt kraft.

Figuren under viser hvordan dette fungerer.



En 180° servo påmontert en liten arm stikkes inn under flensen slik at den kan løfte pennen som vist til høyre på figuren over. Som feste for servoen benyttes den samme anordningen som beskrevet for montering av servo over nesehjulet. Denne snus og flyttes til settet med hull som er nærmere sentrum av roboten, som vist på bildet til høyre. For å få til dette, må koblingsbrettet fjernes.





En 180 servo presses ned i åpningen i braketten som vist på bildet under.



En kan se for seg at pennen styres opp og ned ved hjelp av knapp A og B på Micro:biten (senderen).

#### 6.1.5 Batterier

Vi har valgt å montere to batteriholdere, hver for 2 stk. 1,5 V AA batterier. En smal spalte i chassiet indikerer kanten på batteriholderen. Det er viktig at batteriholderne monteres på rett sted både for å unngå konflikt med andre komponenter, men ikke minst for å sørge for at tyngdepunktet kommer tilstrekkelig langt foran slik at en unngår at roboten tipper bakover.







Figuren under viser hvordan de to batteriholderne er monter på undersiden av chassieplaten.

Vi foretrekker batteriholdere med innlegg fra oversiden da denne gjør det lettere å legge inn batteriene. Batteriholderne er festet til chassiet med dobbeltsidig tape. Vi har valgt å bruke 6V siden servoene får bedre ytelse med 6 enn med 3.3 V. Regulatoren på kortet senker spenningen til 3,3 V tilpasset Micro:biten i tillegg til at spenningen filtreres slik at støy fra servoene ikke når Micro:biten.

## 6.1.6 Kantkontakt- og Micro:bit kort

Her vil vi vise et par alternativer for tilkobling av Micro:bit-kortet til roboten.

#### Kitroniks kantkontaktkort



Et relativt rimelig alternativ er å bygge om kantkontakten som leveres fra Kitronik slik at de to 360° servoene og en ev. 180° servo for å styre en bevegelig arm, kan plugges rett inn i kretskortet til kontakten. Figuren under viser hvordan Kitroniks kantkontakt med kort kan modifiseres.



Fordelen med en slik løsning er at servoene kan kobles direkte til kantkontakten uten å gå om koblingsbrettet som kan brukes til andre ting. Dessuten kan seriekoblingen av de to batteripakkene kobles direkte inn i hylsekontaktene som vist under. Det er en fordel at uttaket fra batteripakkene legges nært opp til kantkontakten, det samme gjelder hullet for gjennomføring av ledningene fra



#### batteripakken.



Som vi ser av figuren over så forsyner batteripakke 1 Micro:bit samtidig som den seriekobles med batteripakke 2 for å gi spenning til servoene. Det en bør være klar over er et en slik løsning ufiltrert kan gi støy fra servoene til Micro:bit-kortet. Vi har sett antydninger til det når vi bruker akselerometeret på Micro:bit-kortet på roboten. Dette kan løses ved å filtrere spenningen til Micro:bit eller ved å skille spenningskilden for Micro:bit fra spenningskilden til servoene.

Bildet under viser oppkoblingen av de to 360° servoene.





Bildet under viser hvordan den tredje servoen tilkobles.



Figuren under viser et enkelt koblingsskjema for oppkobling av de to servomotorene og batteriene.





#### Kodegenets Servo:bot-kort for Micro:bit

Senhøsten 2017 ble det utviklet et nytt kretskort for Micro:bit beregnet for bruk i roboter. Bildet under viser kortet.



#### Servoer

Som vi ser så har kortet tilkobling for fire servoer, i praksis kun tre som kan styres med biblioteksfunksjoner. Den fjerde må ev. styres direkte med pulser. Følgende beskriver koblingen mellom servoutganger og porter:

Servo 1 -> P0 (port 0) Servo 2 -> P1 (port 1) Servo 3 -> P2 (port 2) Servo 4 -> P8 (port 8)

#### Ultrasonisk avstandsmåler

De fire hylsekontaktene er tilpasset en standard avstandsmåler som enten kan plugges rett i kontakten eller flyttes fram til koblingsbrettet lengre fram på roboten ved hjelp av jumpere.

I tillegg er alle portene (kontaktene) på kantkontakten tilgjengelig via den lange hylselisten nærmest kantkontakten. Følgende beskriver koblingen mellom trigger-puls og echo-puls og porter:

Echo -> P14 (port 14) Trig -> P15 (port 15)



## Neopixler

Neopixler er produktnavnet på flerfargede lysdioder koblet i rekke. Disse kan styres fra Micro:bit ved å kobles til hylselistene øverst til venstre og høyre på kortet, som vist på bildet over. Følgende beskriver styringen av Neopxlene:

Data -> P12 (port 12)

#### Strømforsyning

Kortet kan tilføres batterispenning fra 4,5 – 15 V. Spenningen senkes til 3.3 V av spenningsregulatoren og som er spenningen Micro:bit skal ha. Siden servoene tilføres batterispenningen uregulert vil vi her begrense batterispenningen til 6 V som er den optimale spenningen for servoene. Spenningen kan brytes med en Av/På bryter. En diode beskytter mot feilkobling.



## Montering

Kantkontaktkortet monteres med to 3 mm maskinskruer. Utsparinger i chassiet under kretskortet gjør at det ligger støtt. Grunnen til dette er at loddepunktene under sokkelen stikker litt ut og vil uten utsparingene komme i veien under monteringen.

## 6.1.7 Alternative sensorer

#### Reflektanssensorer<sup>12</sup>

I tillegg er roboten klargjort for to reflektanssensorer i snuten på roboten. Disse kobles med tre ledninger til koblingsbrettet og videre til kantkontakten, ev. direkte til kantkontakten. Tegningen til høyre og bildene under viser hvordan

reflektanssensorene (QTR-1A) er festet til chassiet. Disse sensorene kan brukes til å registrere forandringer i reflektert lys fra



12.Kan kjøpes hos Polulu



underlaget og kan for eksempel detektere en kant på et bord eller en sort stripe. En 3-polt hylsekontakt stikkes gjennom plata. Reflektanssensoren som har påmontert en trepolt stiftlist, kan da skyves rett opp i hylsekontaktene.



Denne delen av roboten er ennå ikke testet og vi er i skrivende stund ikke kjent med at det er skrevet programvare for disse sensorene for Micro:bit.

# Avstandssensor (Ultrasonic)<sup>13</sup>

Det er gjort plass på chassiet slik at koblingsbrettet kan monteres på tvers av kjøreretningen. På den måten er det mulig å montere sensorer som beker framover, f.eks. en ultralyd avstandssensor som vist på bildene under. Som det framgår av bildet er denne ennå ikke koblet til Micro:bit.



Denne sensoren registrerer hindringer og avstanden til hindringer. Den vil derfor være et godt hjelpemiddel til å unnvike hindringer som rager opp foran roboten.

13.Kan kjøpes hos www.kultogbilligno



# 6.2 Detaljerte byggebeskrivelser

Her vil beskrive oppbyggingen av vårt prototyp-robot slik den er pr. januar 2018.

#### 6.2.1 Byggebeskrivelse for Kodegenets Servo:bot-kort for Micro:bit

Bildet under gir en oversikt over komponentene som trengs for å bestykke kortet.



Her har vi brukt fire topolte hylsekontakter. Disse vil i noen tilfeller bli byttet ut med en batterikontakt og to trepolte stiftlister som normalt følger med byggesettet.

Figuren under viser komponentplasseringen på kortets komponentside.





## 1. Mulig feilkilde

Da vi først koblet opp kortet oppdaget vi at Av/På bryteren ikke virket. Uansett i hvilken stilling den sto var den på, unntatt i et lite område midt mellom av og på. Etter litt utforsking oppdaget vi at fire klammer (bildet under til venstre) som holder bryteren sammen, dannet en kortslutning mellom de to ytterste loddelandene slik at når bryteren skulle ha vært avslått, så ledet metallet som omsluttet bryteren, strømmen forbi bryteren.



Problemet løses lett ved å bore bort loddelandet (paden) på komponentsiden av kortet som vist på bildene over. En trenger bare å vri en kvass bor på 2-3 mm rundt i hullet med hånda slik den fortinnede kobberringen forsvinner så skulle problemet være løst.

#### 2. Monter lang hylsekontakt

Stikk kontakten inn fra komponentsiden. Lodd pinnene til *loddelandene*<sup>14</sup> på *loddesiden* av kortet. Det kan være lurt å lodde fast ett bein først for deretter å se om kontakten står rett. Dersom den står litt skjevt så er det lett å varme opp loddingen for så å rette opp kortet. Med alle beina fastloddet er dette nesten umulig.



## 3. Mini loddekurs

VIKTIG: Alle komponenter skal loddes til banene på loddesiden selv om dette ikke står i teksten. Bein lengre enn 2 mm klippes av med avbiter etter at loddingen er utført.

<sup>14.</sup> Loddeland er de blanke ringene på loddesiden (baksiden) av kortet. Komponentbeina skal loddes til disse ringene.





- A) Tørk av den varme loddebolten på en våt klut eller svamp slik at spissen blir ren og blank
- B) Smelt litt loddetinn på loddespissen slik at den blir blank å fin
- *C) Press loddespissen hardt ned mot loddestedet slik at både ringen på kortet og beinet til komponenten blir varmet opp. Hold den slik under hele loddingen.*
- D) Etter 4 5 sekunder er loddestedet varmet slik at loddetinn kan tilføres. Tilfør loddetinn der loddespissen møter beinet og loddestedet.
- *E)* Hos en god lodding skal loddetinnet flyte oppover beinet og ut over loddestedet. Dannes det en kule rundt komponentbeinet har ikke loddestedet vært varmt nok og loddingen må smeltes på nytt

#### 4. Monter hylselister for Neopixler

Stikk fire trepolte hylsekontakter ned i hver sine hull som vist på figuren til høyre og lodd beina til loddlandene på loddesiden. Her kan det være spesielt viktig å lodde fast ett bein først for så ev. å gjøre justeringer.



Hylsekontakter for montasje av Neopixler

5. **Montering av hylsekontakt for avstandsmåling** Stikk den firepolte hylsekontakten ned i hullene markert med "Ultrasonic" og lodd fast beina til loddelandene på loddesiden.





#### 6. Montering av fire trepolte stiftlister

Stikk den korte enden av de fire trepolete stiftlistene ned i hullene der det står SERVO1, 2, 3 og 4. Lodd fast *en* av pinnene på hver av siftlistene. Unngå trykk på enkelt pinner under oppvarmingen da de lett forskyver seg i den sorte plastholderen. Sjekk at stiftene står rett, ev. rett opp. Lodd fast samtlige stifter når de står rett.

7. Montering av Av/På-bryter

Stikk de tre beina til glidebryteren ned i de tre hullene som vist på bildet til høyre. Det er det samme hvilken vei bryteren monteres.

8. **Montering av spenningsregulatoren** Spenningsregulatoren er en trebent komponent med en metallflens på den ene siden.

LD33V

9. Klipp av beina

Etter lodding vil beina til regulatoren stikke ut på baksiden. Bruk en s*ideavbiter* og klipp av beina tett inntil loddingene.



Spenningsregulator

 $Micro: bit-Forslag\ til\ et\ undervisningsopplegg\ 55$ 





Flens



#### 10. Montering av beskyttelsesdiode

Bøy beina på dioden i rett vinkel tett inn til diodehuset som vist til venstre på bildet til høyre. Stikk beina til dioden gjennom hullene der det står *Diode* på kretskortet. Pass på at dioden plasseres rett vei med den grå ringen i nederkant som vist på bildet.

Lodd beina på baksiden og klipp av tett inntil loddingen.

 Montering av avkoblingskondensator 1µF Monter avkoblingskondensatoren som vist på bildet til høyre. Merk at kondensatoren må plasseres rett vei. – til venstre som vist på bildet til høyre.

 Montering av avkoblingskondensator 0,33μF Monter avkoblingskondensatoren som vist på bildet til høyre. Merk at kondensatoren må plasseres rett vei. – bak som vist på bildet til høyre.







)]

 Montering av avkoblingskondensator 0,1μF Monter avkoblingskondensatoren som vist på bildet til høyre. Merk at kondensatoren må plasseres rett vei. – bak som vist på bildet til høyre.

14. Montering av 2 x 3 stiftlister

Monter 2 x 3 siftlister. Merk at stiftlistene skal stå tett sammen. Disse er ment å være en tilkoblingsressurs for spenning (3.3 V) og jord (GND). Merk at man må bruke jumpere med hunn-kontakter.

# 15. Montering av batterikontakt

Monter en JST<sup>15</sup>-plugg (batterikontakt) for tilkobling av batterispenning (6 V eller 9 V). MERK: Legg spesielt merke til hvilken vei pluggen står, det er avgjørende om polariteten blir riktig.

Kortet er nå klart for å tilkobles batterispenning og servoer.

# 6.3 Test av kortet og servoene

15.JST - Japan Solderless Terminal

Vi skal nå teste at kortet fungerer som det skal for at vi skal få til dette må vi koble på servomotorene. For at vi skal

være istand til å se at de beveger seg kan det være lurt å sette på hjulene.









## 1. Sett på hjulene

Før servomotorene monteres på chassiet kan vi sette på hjulene. Disse presses inn på akslingen og holdes på plass av en skrue som skrus inn i navet. Skrøen ligger i posen som følger med servoene.



## 2. Kontroller Servo:bot kortet - NB!

Tiden er nå inne for å kontrollere at kortet er koblet opp riktig og at det ikke er kortslutninger på kortet. Ta gjerne kontakt med en av kurslederne slik at dere sammen kan kontrollere kortet.

 Koble servoene til Servo:bot kortet Plugg ledningene fra servoene ned i de to stiftkontaktene "Servo 1 og 2". PASS PÅ at den sorte ledningen er på rett side av kontakten (se bildet til høyre).



## 4. Sett på/av-bryter i "Off"

Før batteriet kobles til, sett på/av-bryteren i "Off" posisjon. Dvs. skyv den mot venstre sett fra kantkontakten som vist på bildet til høyre.



#### 5. Koble til batteripakken og slå på strømmen

Siden vi ikke har montert roboten, velger vi å bruke en 6V batteripakke med JST-plugg (hvit). Sett JST-pluggen inn i den hvite sokkelen. Pass på at denne står riktig vei. Sett på/av-bryteren i "on" posisjon.

## 6. Legg inn testprogrammet i Micro:biten

Vi skal nå legge inn et enkelt testprogram i Micro:biten som skal styre roboten.



Når knapp A trykkes starter høyre motor, når knapp B trykkes starter venstre motor. Når knapp A og B trykkes skal begge motorene stoppe. Om de ikke står helt stille må de trimmes slik at de står i ro.

#### 7. Trimming av motorene

Trykk A+B slik at motorene står stille. Om de likevel går bruk et lite stjerne-skrutrekker og juster skruen bak på motoren som ikke står stille. Juster skruen til motoren står i ro. Det skal svært lite til.







#### 8. Modifiser programmet

Skriv i følgende program og forklar for hverandre hvordan programkoden virker. Undersøk om beskrivelsen ble riktig.



# 6.4 Byggebeskrivelse robot

Bygge beskrivelsen av roboten er delt i to:

- 1. Oppbygging av grunnenhet med chassie
- 2. Oppbygging av tegneenhet



## 6.4.1 Byggebeskrivelse – Grunnenhet med chassie

Dette vedlegget gir en detaljert byggebeskrivelse for en enkel standardutgave av roboten. Med utgangspunkt i denne kan man lage sine egen versjoner.



Bildet viser en oversikt over komponentene som inngår i byggesettet (se også komponentliste vedlegg A.1 på side 76). I enkelte sammenhenger vil det være aktuelt at deltagerne selv skriver ut chassie og nesehjul ev. deler til pennholder.

1. Montering av servoer

Servoene presses ned i de profilerte åpningene på hver side av chassiet. Det kan være nødvendig å file litt i spaltene der flensene skal inn. Servoene presses ned i åpningen til de ligger kant i kant med plata på undersiden.

 Montering av kantkontakt-kort Vi forutsetter at kantkontakt-kortet er ferdig oppkoblet (se byggebeskrivelse avsnitt 6.2.1 på side 52). Kantkontakt-kortet monteres med to skruer (M3, 12 mm) med mutter (M3). Utsparingene i chassiet gjør det unødvendig å løfte plata fra kortet.







3. *Klargjøring av batteriholdere* Sett en dobbeltsidig tape på undersiden av hver av batteriholderne. Dersom batteriholderne krever batterikontakter, klips disse på kontaktene.



4. *Monter batterikassettene* De to batterikassettene monteres under hver av de to servomotorene med dobbeltsidig tape. Framenden av batteriholderne skal gå kan t i kant med to spalter i chassiet. Ledningene skal vende mot kantkontakten. Dersom ledningene skal tilkobles batterikontakten på kortet, føres ledningene gjennom hullet og opp på oversiden.



5. Tilkobling av batteriledninger

De to batterikassettene skal seriekobles slik at vi får totalt 6 V. Dette gjøres ved å lodde den røde ledningen (+) fra den ene til den sorte (-) fra den andre, som vist på bildene under. Husk å tre på krympestrømpa før lodding. Dra deretter strømpa over loddestedet og bruk loddebolten til å varme opp strømpa slik at den krymper omkring skjøten.





Den mest elegante løsningen er å lodde de to gjenværende ledningene til undersiden av batterikontakten som vist på bildet under. En må da tilpasse lengden til ledningene og sørge for at polariteten blir riktig som vist.



Alternativt kan bruke ledningene i full lengde, føre dem gjennom hullet i plata og stikke dem ned i JST-kontakt eller hylsekontakten som vist på figuren under.



6. Montering av nesehjul

Nesehjulet monteres med fire skruer med mutter (M3, 20 mm). Sammen med nesehjulet kan man samtidig, på oversiden, montere en brakett (plater) for montasje av en 180° servo.





7. *Alternativ 1. Montering av gummihjul* Gummiringen legges på felgen slik at forhøyningen langs felgen fyller slissen til dekkene. Hjulene skrus så fast til akslingen med en liten skrue som følger med servoene.



8. *Alternativ 2 - Bruk av laserkuttede hjul* I chassie-fila medfølger to hjul. Disse kan alternativt brukes i stedet for standard plast- og gummihjul som kan kjøpes.

Sammen med hjulene følger to navringer som limes (ev. skrues) fast i navet til hvert av hjulene. Tanken er at navringene skal skyves inn på akslingen til servomotoren. Deretter skrus en skrue gjennom hjulet og inn i akslingen til servoen.





Bildene under viser montering av navringene på hjulene.





En kan legge en ballonggummi rundt felgen for å øke friksjonen.



## 9. Tilkobling av servoer

De to servoene kobles til de to stiftlistene som vist på bildet under (innringet). Pass på at hylsekontakten blir stående rett vei, med den hvite ledningen til venstre sett bakfra som vist på bildet.



10. Den ferdige roboten

Merk at Micro:bit skal monteres slik at lysdiodedisplayet er vendt opp. Dersom man har behov for å utstyre roboten med ekstra elektronikk, så kan det gjøres på et koblingsbrett som monteres foran servoene som vist på bildet under. Dersom man ønsker å montere en penn, bør man kanskje vente med å montere koblingsbrettet.







Vi har erfart at roboten kan bli litt baktung til tross for at plasseringen av servoer og batterier skal ta hensyn til dette. I tilfelle at roboten skulle bikke bakover er det laget to hull, et på hver side av servobraketten, hvor det er mulig å montere skruer med muttere som kan fungere som vekter.



#### 6.4.2 Byggebeskrivelse – Tegneenhet

I dette avsnittet skal vi beskrive hvordan vi kan utstyre roboten med en tusjpenn slik at roboten kan tegne streker mens den kjører. Vi ønsker også å kunne fjernstyre pennen ved hjelp av knappene på Micro:bit.



#### 1. Fjern koblingbrettet



Dersom koblingbrettet er montert med den medfølgende dobbeltsidige tapen, kan det være vanskelig å ta av dette uten ødelegge brettet. Det som kan skje er at tapen tar med seg hele baksiden av koblingsbrettet slik at koblingsskinnene blir med ut.

2. Snu servobraketten

Chassiet har flere sett med hull som gjør det mulig å flytte og endre retning på braketten slik at 180° servoen kan snus mot hullet som er ment for pennen. En spalt i plata gir plass til flensen til servoen når den skyves inn gapet til brakken.



## 3. Monter røret som styrer pennen

Et 3D-printet rør med flens stikkes opp fra undersiden av chassiet og låses ved å dreie røret 90 slik at flensen smetter under batteriholderne.





4. *Monter muffe med flens på pennen* En 3D-printet muffe tilpasset den valgte pennen skyves inn på pennen. Denne er så trang at muffa blir sittende i den ønskede posisjonen. Muffa skal plasseres slik at når servoens arm heves, så skal pennen forlate papiret.



5. Montering av servo og penn

En 180° servo presses inn i braketten og pennen slippes ned i røret. Armen på servoen monteres slik at den er istand til å løfte pennen opp fra papiret.



 Tilkobling av servo Servoen kobles til servo 3 på kretskortet (innringet på bildet til høyre).





# 6.5 Programmering av Micro:bit roboten

I dette avsnittet skal vi se hvordan vi kan bygge opp programvaren som skal styre roboten etter intensjonen. I innledningen har vi valgt meget enkle programmer som har stort potensial for forbedring. I tillegg ønsker vi å løfte og senke pennen fra papiret, til denne funksjonen ønsker vi å bruke de to bryterne A og B.

For å være istand til å programmere de tre servoene må vi vite hvilke porter som styrer de ulike servoene. Disse opplysningene står også på kretskortet.



Servo 1 -> P0 (port 0) Servo 2 -> P1 (port 1) Servo 3 -> P2 (port 2) Servo 4 -> P8 (port 8)

I tillegg er kretskortet forberedt for en avstandssensor (

## 6.5.1 Styring av roboten

Siden målet er at roboten skal kunne styres framover, bakover og svinge til høyre og venstre fra en håndholdt Micro:bit, så må vi lage program både for senderen (Micro:bit'en i hånda) og mottakeren (Micro:bit'en på roboten). Både sender- og mottakerenheten er Micro:bits. Senderenheten må ha eget batteri, mens mottakerenheten kan forsynes fra robotens strømkilde.



**Oppdraget:** Siden en Micro:bit kun har to knapper velger vi å bruke akselerometeret for styring. Vi ønsker at roboten skal kjøre forover når vi heller senderen framover og bakover når vi heller den bakover. Tilsvarende vil vi at den skal svinge til venstre når vi heller den mot venstre og tilsvarende mot høyre. I tillegg vil vi at farten og hvor brått den svinger skal være avhengig av helningsvinkelen.



#### 6.5.2 Programmering av senderenheten

Figuren under gir et forslag til programmet i senderen.



Figuren over er nesten selvforklarende. Variasjoner i y-aksen er knyttet til tilting forover og bakover, mens x-aksen er knyttet til tilting sideveis.



Kjør bakover

Å holde orden på retningene er mottakerens oppgave. En kan imidlertid med fordel legge inn piler på displayet slik at fjernstyreren ser at senderen fungerer korrekt.



## 6.5.3 Programmering av mottakerenheten

Programmet for mottakeren er noe mer komplisert da dette programmet må tolke signalene fra senderen. Figuren under viser programmet for mottakeren som er så kompakt det er mulig å lage det.



Fordelen med dette programmet er at det er enkelt og det fungerer faktisk ganske godt når man legger senderen i hånda. Ev. avvik vil automatisk korrigeres i bevegelsen av hånda på bakgrunn av observasjoner av roboten. Likevel er det stort potensial for å forbedre koden.

## 6.5.4 Program for styring av penna

Vi ønsker å kunne løfte og senke penna ved å trykke knappene A og B på Micro:biten. Vi ønsker at også denne funksjonen skal være fjernstyrt og på sikt integrert med styring av robotens framdrift.

Figuren til høyre et enkelt testprogram som installeres på Micro:biten som er montert på roboten for å teste servoen som kobles til Servo 3 som igjen styres av port P2.







#### Penn kontroll – sender

Vi bruker samme teknikk her som for styre roboten. Siden vi ikke trenger å overføre noen parameter velger vi å bruke en enklere kommunikasjonskommando som kun overfører ordene "Lower" for å senke pennen, og "Lift" for å heve pennen. I tillegg vises et symbol på skjermen som indikerer om pennen heves eller senkes.




#### Penn kontroll – mottaker

Også i mottakeren benyttes samme metode som vi brukte for å styre roboten.

Kjør rutine ved mottatt melding



Avhengig av hvilket kommandoord som overføres, løftes eller senkes armen til servoen. Legg merke til at symbolene som viser stillingen til pennen er tegnet opp-ned slik at det skal se fornuftig ut når roboten betraktes bakfra.

Bildene under viser hvordan servoen oppfører seg og hvordan displayet illustrerer det som skjer.





#### 6.5.5 Kombinert program med kjørekontroll og penn

Vi skal nå kombinere styring av robotens bevegelser og løfting og senking av pennen. Dette er ikke vanskeligere enn at vi kombinerer de to programmene i ett.

Hill     Hill					÷
	al radio send tekst ( 46 Lower 22	radio se	end teks	st 🕻 🧐 🗋	ift D
	₩ pause (ms) 0 500	pause (m	ns) ( 5	00	
gjenta for alltid	## pause (ms) 0 500	pause (m	ns) ( 5	00	
gjenta for alltid ""  radio send verdi   44 a	cc_y 22 = [ ⊙ akselerasjon (mG) y x	pause (m	ns) 0 5	00	
gjenta for alltid "al radio send verdi 0 44 a III pause (ms) 0 50	cc_y <sup>22</sup> = ( o akselerasjon (mG) ()	pause (m	ns) 0 51	00	

Kombinert robotprogram med kjørekontroll og styring av penn – sender

Kombinert robotprogram med kjørekontroll og styring av penn – mottaker







# Vedlegg A Komponentliste

## A.1 Komponent og innkjøpsliste

Komponentene kan kjøpes fra mange forskjellige forhandlere. Liste under gir en mulig løsning:

2 stk. Hjul for FS90R 60 x 8 mm		Kitronik (pris 2.30 £ pr. par)
1 stk. Servo:bot-kretskort for Micro:bit (må mo	nteres)	Kodegenet (pris kr. 75,- pr. stk)
1 stk. Stiftlist header		Kultogbillig.no (pris ca. 1 kr.)
1 stk. Hylselist socket		Kultogbillig.no (pris ca. 1 kr.)
2. stk. 360° servo FS90R		Kitronik (pris 7.70 £ pr. par)
1. stk. Mini prototyp bread board		Kitronik (pris 1.89 £ pr. stk.)
2. stk. PP3 batteri klips		Kitronik (pris 0.52 £ pr. par)
2. stk. 2xAA Batteriholder		Kitronik (pris 0,66 £ pr. par)
6 stk. Sporskruer M3x20/16		Clas Ohlson 11-1124-320
6 stk. muttere M3		Clas Ohlson 11-503-3
4 stk. AA batterier		Biltema (pris 14,00 kr. pr. bil)
1 stk. Glidelås poser for pakking		Biltema (pris 26.90 kr. 10 stk)
1 stk. Klinkekule		BR-Leker
1 stk. chassie 3 - 6 mm MDF/bjørkefiner		ViT i Trondheim
1 stk. Nesehjulholder - PLA		ViT i Trondheim
Alternativer:		
1 stk. Kretskort m/kantkontakt for Micro:bit (m	å monteres)	Kitronik (pris 2.10 £ pr. stk)
1 stk. Kretskort m/kantkontakt for Micro:bit (m https://www.kitronik.co.uk/5601b-edge-conne	ontert) ctor-breakout-boa	Kitronik (pris 4.16 £ pr. stk) rd-for-bbc-microbit-pre-built.html
1 stk. Servo:bot, kretskort for Micro:bit (ferdig	loddet) I	Kodegenet (pris kr. 139,- pr. stk)
Leverandører:		
Kodegenet (Norsk)	https://kodegen	et.no/shop/category/microbit
Kult og Billig (Norsk)	http://kultogbil	lig.no/Elektronikk
Kitronik (Engelsk)	https://www.ki	tronik.co.uk/microbit.html
Polulu Robotics and Electronics (Amerikansk)	https://www.po	lolu.com/

# Vedlegg B Løsningsforslag øvingsoppgaver

## B.1 Løsningsforslag øving 1 – Lag en animasjon

Hensikten med øvingen er å erfare hvor lav terskelen for å få til et enkelt program.





## B.2 Løsningsforslag øving 2 – Bruk av variabler

Hensikten med øvingen er å bli kjent med bruken av variable



## B.3 Løsningsforslag øving 3 – Bruk av akselerometer

## B.3.1 Løsningsforslag til øving 3A

Hensikten med øvingen er å vise hvor enkelt det er å bruke knappene som inngangsparametere for aksjoner.



## B.3.2 Løsningsforslag til øving 3B

Her finnes det sannsynligvis flere løsninger, men bruk av akselerometeret er en.



## B.3.3 Løsningsforslag til øving 3C

Hensikten med øvingen er å å lære å bruke if-setning, operatorene > og <, og å hente in data fra akselerometeret. Dette krever også at en forstår hvordan akselerometeret operer.

III gjenta for	alltid	
🔯 hvis	( 🕜 akselerasjon (mG) 🗙 🕬 🗸 50	0
	vis tekst 🕻 🥰 🖁 🥬	
ellers hvis	· ( o akselerasjon (mG) 💽 💽 🚺 -5	00
	vis tekst 🕻 🥰 🗴 🤉	
ellers hvis	: ( 🚺 💿 akselerasjon (mG) 👽 🔊 🖬 50	9
	vis tekst ( 6 B 🥬	
ellers hvis	: ( / ⊙ akselerasjon (mG) 👽 💶 🗸 -5	99
	vis tekst 🕻 🤐 🖡 🨕	
ellers 📻	vis bilde	

## B.4 Øving 4 Overføring av informasjon via radio

## B.4.1 Løsningsforslag øving 4 – Overføring av en aksjon

Hensikten med øvingen er å vise hvordan informasjon kan overføres via en trådløs radioforbindelse.

Senderdel



Mottakerdel

 $Micro: bit-Forslag\ til\ et\ undervisningsopplegg\ \ 79$ 



## B.5 Løsningsforslag skritteller

Hensikten med denne øvelsen er å kunne forstå hvordan akselerometeret fungerer og kan brukes til å bestemme helningsvinkelen til Micro:biten, videre å lære å bruke tellevariabler.

Det er ikke vedlagt noe løsningsforslag.

## B.6 Løsningsforslag lysteremin

Hensikten med øvelsen er å lære hvordan man kan lese av lysstyrken og konvertere denne verdien til en tone.



## B.7 Løsningsforslag til oppgave 7 – Lag kompass som peker mot nord

Hensikten er å lære å benytte magnetometeret som kompass, og å få erfaring med hvor pålitelig kompassfunksjonen er.

III gjenta f	or alltid												1.10	
D hvis	(	348 ≤ 🔹	[ <sup>1</sup> ⊙ komp	assretr	ing (°	)	eller		13		0	kompas	sretni	ng (°)
D hvi	is ( <b>500</b>		⊙ kompass	retning	(°)		12	24	10	<i>.</i>	1	1	0.01	10
	vis tekst (	G CALI	BRATE 22	10										
III ton	n skjermen	1.1												
	<b> </b>	-												
ellers hy	vis ( / /	13 💷	∫	ssretni	.ng (°)		og 🔹	C 35	<b>) 2</b>	⊚ ]1	kom	passret	ning (	•)
III ton	n skjermen	1.5	$\lambda = \lambda$	- 0		2	- 21		1		÷		1	
III vis	bilde													
		-												











lers hvis (	0 30	5 💷	0	komp	assretr	ning (	(°)	og •	9	26		⊙ ko	mpassr	etning	(
🎹 tøm skj	ermen			14	1.1						14		- 0		
III vis bil	de														
									_		_		_	_	
lers hvis (	( <sup>1</sup> 0 320	6 50	0	komp	assreti	ning (	(°)	og 🔹	1	48		⊙ ko	mpassr	etning	; (
lers hvis (	f 0 320 ermen	6 30	0	komp	assretr	ning (	(°)	og 🔹		48		⊙ ko	mpassr	etning	: (
lers hvis ( III tøm skj III vis bil	f 0 320 ermen de		0	komp	assretr	ning (	(°)	og		48		⊙ ko	mpassr	etning	; (
lers hvis ( III tøm skj III vis bil	ermen de		ſ <mark>.</mark> 0	komp	assretr	ning (	(*)	og 🖸		148		⊙ ko	mpassr	etning	: (
lers hvis ( III tøm skj III vis bil	f t 320 ermen de		0	komp	assretr	ning (	•	og		148		⊚ ko	mpassr	etning	: (
lers hvis ( III tom skj III vis bil	f t 32( ermen de		0	komp	assretr	ning (	•)	(og 💽		448		⊘ ko	mpassr	etning	: (
lers hvis (	de		0	komp	assretr	ning (	•)	og ••		448		⊘ ko	mpassr	etning	: (

## B.8 Løsningsforslag til oppgave 8 – Når Micro:bit "forstår" hva som blir sagt

Hensikten med denne oppgaven er å tenke kreativt for å kunne løse oppgaven. Det finnes sannsynligvis flere løsninger på oppgaven. Foreløpig er det derfor ikke foreslått noen løsning.

#### **B.9** Styring av 360° servo

### B.9.1 Løsningsforslag oppgave 9 A

Hensikten med oppgaven er dels å få erfaring med styring av 360° servoer ved hjelp av knapp A og B.





## B.9.2 Løsningsforslag oppgave 9 B 360° Servo – Styring med knapper

Hensikten med oppgaven er dels å få erfaring med styring av 360° servoer og dels med å bruke samme bryter til flere funksjoner eller i dette tilfellet "toggling".





# Vedlegg C OpenSCAD kode

## C.1 Holder for nesehjul

Koden under viser koden som skriver ut holderen for nesehjulet.

```
difference ()
{
cylinder($fn = 4, $fa = 3, $fs = 2, h = 20, r1 = 10, r2 = 10, center =
false);
translate ([0,0,17]) sphere ($fn = 100, 7.9);
translate ([0,0,18]) cylinder($fn = 40, $fs = 2, h = 4, r1 = 7.5, r2 =
7.5, center = false);
}
difference ()
{
translate ([-17.0,-14.5,0]) cube(size = [34, 29, 2], center = false);
translate ([-14,-11.5,-1]) cylinder($fn = 10, h = 7, r1 = 2.0, r2 = 2.0,
center = false);
translate ([-14,11.5,-1]) cylinder($fn = 10, h = 7, r1 = 2.0, r2 = 2.0,
center = false);
translate ([14,-11.5,-1]) cylinder($fn = 10, h = 7, r1 = 2.0, r2 = 2.0,
center = false);
translate ([14,11.5,-1]) cylinder (\$fn = 10, h = 7, r1 = 2.0, r2 = 2.0,
center = false);
}
translate ([0,10,0]) cylinder($fn = 10, h = 20, r1 = 1.5, r2 = 1.5, center
= false);
```

```
translate ([10,0,0]) cylinder($fn = 10, h = 20, r1 = 1.5, r2 = 1.5, center
= false);
```

```
translate ([0,-10,0]) cylinder($fn = 10, h = 20, r1 = 1.5, r2 = 1.5, cen-
ter = false);
translate ([-10,0,0]) cylinder($fn = 10, h = 20, r1 = 1.5, r2 = 1.5, cen-
ter = false);
```

// Ønsker man å inkludere kula legger man til følgende setning

```
color("white"1.0) translate ([0,0,17]) sphere ($fn = 200, 7.8);
```

## C.2 Deler for løfting av penn

I dette vedlegget presenteres deler til pennholderen.



## C.2.1 Pennholder ytre rør med flens

Program for 3D-printing av ytre rør med flens:

```
difference()
{
    union()
    {
      cylinder($fn = 100, h = 25, r1 = 6.6, r2 = 6.6, center = false);
      cylinder($fn = 100, h = 1, r1 = 10, r2 = 10, center = false);
    }
    translate([0, 0, -1]) cylinder($fn = 100, h = 27, r1 = 5.5, r2 = 5.5,
    center = false);
    translate([21, 0, -1]) rotate ([0, 0, 45]) cylinder($fn = 4, h = 3, r1
      = 20, r2 = 20, center = false);
    translate([-21, 0, -1]) rotate ([0, 0, 45]) cylinder($fn = 4, h = 3, r1
      = 20, r2 = 20, center = false);
}
```

## C.2.2 Pennholder muffe med flens

Program for 3D-printing av muffe med flens som tres ned over pennen:

```
difference()
{
    union()
    {
      cylinder($fn = 100, h = 7, r1 = 6.6, r2 = 6.6, center = false);
      cylinder($fn = 100, h = 1, r1 = 17, r2 = 17, center = false);
    }
    translate([0, 0, -1]) cylinder($fn = 100, h = 9, r1 = 5.5, r2 = 5.5,
    center = false);
}
```



# Vedlegg D Læreplaner

## D.1 Teknologi og Forskningslære 1

Fargekodene forsøker å antyde hva som *lett kan* knyttes til bygging av en robot med Micro:bit og programmering (rødt/kursiv), hva som er litt på siden (blått) og det som vanskelig lar seg oppfylle innen prosjektet (sort).

#### Den unge ingeniøren - Beskrivelse av hovedområde

Mål for opplæringen er at eleven skal kunne:

- planlegge og bygge en konstruksjon som er fast eller bevegelig, og som har en definert funksjon
- bruke tredimensjonale tegninger eller skisser i utvikling av konstruksjoner
- bruke forskjellige materialer og former for sammenføyninger og begrunne valg av materialer og byggemåte ut fra materialenes egenskaper og konstruksjonens funksjon
- bruke sensorer og styringssystemer i forbindelse med forsøk og konstruksjoner
- dokumentere og vurdere konstruksjoners fysiske egenskaper og funksjonalitet ved hjelp av målinger og enkle beregninger

#### Den unge forskeren - Beskrivelse av hovedområde

Mål for opplæringen er at eleven skal kunne:

- gjøre rede for hvordan et naturvitenskapelig prosjekt planlegges, gjennomføres og etterarbeides før det blir publisert
- planlegge, gjennomføre, analysere og dokumentere systematiske målinger
- om støy, luftforurensning, inneklima og vannkvalitet, og drøfte virkninger på helse og miljø

#### Teknologi, naturvitenskap og samfunn - Beskrivelse av hovedområde

Mål for opplæringen er at eleven skal kunne

- drøfte etiske, miljømessige, kulturelle og politiske sider ved teknologisk utvikling (bruk av robotteknologi)
- beskrive den historiske utviklingen av en teknologisk innretning, forklare virkemåten og drøfte anvendelser i samfunnet
- gjøre rede for utvikling og produksjon av et teknologisk produkt og vurdere produktets brukervennlighet, utviklingsmuligheter og miljøpåvirkning
- beskrive prinsipper og virkemåte for noen moderne instrumenter i industri, helsevesen eller forskning, og gjøre rede for nytten og eventuelle skadevirkninger
- kartlegge og presentere praktisk bruk av realfag i en lokal bedrift eller institusjon



#### Design og produktutvikling - Beskrivelse av hovedområde

Mål for opplæringen er at eleven skal kunne

- gjøre rede for funksjonen til vanlige komponenter i elektroniske kretser, og gjenkjenne komponentene i en krets
- lage elektroniske kretser ved å lodde komponenter og simulere og teste kretsene
- forme og utvikle produkter som har en definert funksjon og inneholder elektronikk
- dokumentere og presentere designprosesser fra idé til ferdig produkt
- begrunne valg av materialer i produkter og vurdere produktenes form og funksjon, miljømessige konsekvenser, estetikk og forbedringsmuligheter
- utføre målinger med eller teste et eget produkt, og vurdere kvaliteten på produktet med tanke på funksjonalitet

### D.2 Teknologi og forskningslære 2

#### Den unge forskeren - Beskrivelse av hovedområde

Mål for opplæringen er at eleven skal kunne

- gjøre rede for et forskningsprosjekt i en bedrift eller institusjon, og beskrive problemstillinger, organisering, måleutstyr, resultater og finansiering
- planlegge og gjennomføre naturvitenskapelige undersøkelser basert på egne ideer, og presentere arbeidet i en vitenskapelig form
- drøfte resultater fra egne undersøkelser i forhold til relevant kunnskap på området, og vurdere hvordan kontroll av variabler og reproduserbarhet er ivaretatt

#### Naturvitenskapelige arbeidsmetoder - Beskrivelse av hovedområde

Mål for opplæringen er at eleven skal kunne

- forklare hva som menes med modell, teori og hypotese, og gjøre rede for hvordan de brukes og utvikles i forskning
- drøfte ved å bruke eksempler hvordan empiriske data kan styrke eller forkaste en hypotese
- gjøre rede for hvordan forskning utvikles og kvalitetssikres gjennom samarbeid, kritisk vurdering og argumentasjon
- gjøre rede for strukturen i en vitenskapelig publikasjon eller presentasjon

#### Forskning, teknologi og samfunn - Beskrivelse av hovedområde

Mål for opplæringen er at eleven skal kunne

- beskrive kjennetegn ved grunnforskning, anvendt forskning og utviklingsarbeid og gjøre rede for hovedtrekk ved finansiering og styring
- 88 Micro:bit Forslag til et undervisningsopplegg



- gjøre rede for betydningen av naturvitenskapelig forskning og teknologiutvikling for næringsliv og samfunn
- drøfte økonomiske, miljømessige og etiske spørsmål i forbindelse med naturvitenskapelig forskning og teknologiutvikling
- drøfte og gi eksempler på hvordan forskningsresultater og ny teknologi formidles og brukes av forskningsinstitusjoner, medier, bedrifter, interessegrupper og myndigheter

### Vitenskapsfilosofi og vitenskapsteori - Beskrivelse av hovedområde

Mål for opplæringen er at eleven skal kunne

- beskrive hovedtrekk i den historiske utviklingen av vitenskapelige tenkemåter og drøfte teknologiens rolle i denne utviklingen
- gjøre rede for hovedideene til noen sentrale vitenskapsteoretikere og vitenskapsfilosofer vurdere hvordan argumentasjon i aktuelle naturvitenskapelige debatter bygger på empiriske resultater, teoretisk kunnskap og ideologisk ståsted

## D.3 Teknologi i Praksis (TiP) – ungdomsskolen

#### Formål

Valfaga skal medverke til at elevane, kvar for seg og i fellesskap, styrker lysta til å lære og opplever meistring gjennom praktisk og variert arbeid. Valfaga er tverrfaglege og skal medverke til heilskap og samanheng i opplæringa.

Teknologi handlar om den menneskeskapte verda og om innretningar og system som kan gjere kvardagen betre. Opp gjennom tidene har menneska brukt kreativitet og skaparevner til å utvikle reiskapar, maskinar og andre teknologiske produkt og løysingar. Teknologien grip inn på mange område, og har gjeve og kan gje både moglegheiter og utfordringar, både for den einskilde og for samfunnet. Innanfor teknologien finn vi dei enklaste verktøy og produkt og den mest avanserte elektronikken. Erfaring med og innsikt i teknologi kan fremje personleg utvikling, demokratisk deltaking og medverke til eit aktivt forhold til ein teknologisk kvardag.

Valfaget teknologi i praksis skal motivere elevane til å utvikle teknologiske produkt med utgangspunkt i lokale behov og problemstillingar. Prosessen frå idé til eit ferdig produkt kan medverke til skaparglede og meistringsoppleving. Gjennom eige arbeid og i samarbeid med andre kan elevane utvikle ferdigheiter og innsikt. Det inneber å prøve ut eigne talent og moglegheiter på ulike steg i prosessen, vurdere prosessar og produkt og få tilbakemeldingar frå andre.

Valfaget handlar om å planleggje, konstruere og framstille gjenstandar og produkt med varierte materiale og teknologiske løysingar. Kunnskap om teknologiske produkt som blir brukte i dagleglivet, gjev eit godt grunnlag for å forbetre produkt og utvikle nye produkt.

Valfaget hentar hovudelement frå matematikk, naturfag og kunst og handverk/duodji. Element frå norsk/samisk, RLE og samfunnsfag kan også inngå.



#### Hovedområder

#### Undersøkingar

Hovedområdet handlar om korleis teknologiske produkt er konstruerte og verkar, kva for prosessar som inngår i utvikling og bruk, og kva for behov produkta dekkjer. Utvikling, konstruksjon og produksjon av teknologi inngår i hovudområdet, i tillegg til helse, miljø og sikkerheit (HMS). Kunnskap om korleis teknologien byggjer på nokre grunnleggjande prinsipp, og korleis ny teknologi byggjer på tidlegare erfaringar, høyrer også med til hovudområdet.

#### Idéutvikling og produksjon

Hovudområdet omfattar planlegging, framstilling og utprøving av eigne produkt og konstruksjonar. Planar for framstilling og utprøving av eigne produkt og konstruksjonar byggjer på kravspesifikasjon.

I utviklingsfasen er kjennskap til design og verkemåte til andre produkt viktig. Diskusjon omkring ulike sider ved produkta er viktig i alle fasar av produktutviklinga og kan også medverke til å forbetre prosessar og produkt.

#### Kompetansemål

#### Undersøkingar

- undersøkje teknologiske produkt og dei vala som er gjorde med omsyn til bruk, tekniske løysingar, funksjonalitet og design
- demonstrere riktig bruk av utvalde verktøy
- vurdere teknologiske produkt ut frå brukartilpassing, HMS-krav og miljøtilpassing

Idéutvikling og produksjon

- utvikle ein realistisk kravspesifikasjon for eit teknologisk produkt og beskrive kva behov produktet skal dekkje
- framstille produktet med eigna materiale, komponentar, og funksjonelle teknologiske løysingar
- bruke kunnskap om andre produkt i arbeidet med eige produkt
- teste eigne produkt og foreslå moglege forbetringar

#### D.4 Programmering – ungdomsskolen

#### Formål

Valgfagene skal bidra til at elevene, hver for seg og i fellesskap, styrker lysten til å lære og opplever mestring gjennom praktisk og variert arbeid. Valgfagene er tverrfaglige og skal bidra til helhet og sammenheng i opplæringen.

Vi lever i en teknologirik verden hvor de fleste forholder seg til en rekke digitale enheter daglig. Bruken av digital teknologi øker i alle samfunnsområder, innen alt fra samferdsel til helse, og stiller nye krav til digitale ferdigheter. Utviklingen av det teknologirike samfunnet vi har i dag og i



fremtiden, utfordrer både måten vi lærer på og hvilke kompetanser som blir viktige. Uavhengig av yrkesvalg vil en grunnleggende forståelse av teknologiens oppbygning og virkemåter være en nødvendig del av fremtidens kompetanse.

Programmering er en viktig ferdighet i dagens samfunn, som inngår i de fleste fagområder, fra digital musikk til naturvitenskap og matematikk. Programmering åpner for å utforske komplekse og realistiske modeller av virkeligheten. Det gir også utvidede muligheter til å behandle store datamengder. Å gi elever grunnleggende ferdigheter i programmering er med på å forberede dem for fremtidige realfaglige jobber, i tillegg til å øke forståelsen for naturvitenskapelige og matematiske problemer.

Valgfaget programmering handler om å lage programkode, det vil si et sett med regler og uttrykk for å styre digitale enheter. I dette inngår prosessen fra å identifisere problemer og utforme mulige løsninger, til å lage kode som kan forstås av en datamaskin, systematisk feilsøke og forbedre denne koden, og dokumentere løsningen på en forståelig måte. Det omfatter alle nivåer fra å forutse og analysere hva et program skal gjøre, til å kjenne igjen mønstre, eksperimentere og evaluere mulige løsninger, og samarbeide med andre. Summen av disse ferdighetene kalles algoritmisk tankegang.

Opplæringen skal legge til rette for at elevene lærer å løse problemer på nye måter. Elevene skal få muligheten til å utvikle sin kreativitet og skape produkter ved hjelp av programmering. Gjennom å lage programmer oppøver elevene også ferdigheter i å vurdere eget og andres arbeid, gi konstruktive tilbakemeldinger og samarbeide med andre. Å skape og produsere digitalt, krever forståelse og kompetanse i programmering.

#### Hovedområder

Valgfaget er strukturert i to hovedområder. Hovedområdene utfyller hverandre og må ses i sammenheng.

Oversikt over hovedområder:

#### Modellering

Hovedområdet tar for seg stegene som kreves for å løse problemer ved hjelp av programmering, også kjent som algoritmisk tankegang. Til hovedområdet hører kunnskap om hva slags problemer som egner seg for å løses av en datamaskin, hvordan disse kan brytes ned i delproblemer, og hvordan løsninger kan utformes. Modellering av matematiske og naturvitenskapelige fenomener er en sentral del av dette. I hovedområdet inngår hvordan datamaskiner og programmer er konstruert og virker, ulike programmeringsspråk, og styrker og svakheter ved de ulike språkene. Prinsipper som ligger til grunn for god programmeringspraksis inngår også i hovedområdet, deriblant forklaring og dokumentasjon av løsninger og programkode.

#### Koding

Hovedområdet handler om å utvikle egne programmer ved hjelp av ulike programmeringsspråk. I dette inngår å bruke og forstå grunnleggende prinsipper i programmering, slik som løkker, tester, variabler, funksjoner og enkel brukerinteraksjon. Hovedområdet omfatter også kontrollering eller simulering av fysiske objekter, så som roboter, sensorer, baller som spretter og molekylers beve-



gelse. Videre dreier det seg om simuleringer og beregninger basert på matematiske og naturfaglige problemstillinger. Hovedområdet omfatter også feilsøking, generalisering og gjenbruk av løsninger, inkludert vurdering og analyse av egen og andres programkode.

#### Kompetansemål

#### Modellering

Mål for opplæringen er at eleven skal kunne

- gjøre rede for hvordan datamaskiner og programmer fungerer, inkludert et utvalg utbredte programmeringsspråk og deres bruksområder
- omgjøre problemer til konkrete delproblemer, vurdere hvilke delproblemer som lar seg løse digitalt, og utforme løsninger for disse
- dokumentere og forklare programkode gjennom å skrive hensiktsmessige kommentarer og ved å presentere egen og andres kode

#### Koding

Mål for opplæringen er at eleven skal kunne

- bruke flere programmeringsspråk der minst ett er tekstbasert
- bruke grunnleggende prinsipper i programmering, slik som løkker, tester, variabler, funksjoner og enkel brukerinteraksjon
- utvikle og feilsøke programmer som løser definerte problemer, inkludert realfaglige problemstillinger og kontrollering eller simulering av fysiske objekter
- overføre løsninger til nye problemer ved å generalisere og tilpasse eksisterende programkode og algoritmer.



# Vedlegg E Filer for laserkutter

## E.1 Chassie for robot



Viktig!!! Juster størrelsen slik at målet til venstre blir akkurat 100 mm



## E.2 Magisk terning

Denne tagningen er forminsket pass på at størrelsen blir juster til riktig størrelse.





Heftet er ment som et forslag til prosjekter hvor BBCs Micro:bit er benyttet som programmerbar styringsenhet. Det beskrives to prosjekter, Den magiske terningen og en fjernstyrt robot. Heftet ikke en lærebok i blokkprogrammering av Micro:bit, men heller et idehefte til et par prosjekter som kan brukes til å lære seg programmering av Micro:bit. Tanken er at heftet ikke skal være en fasit, men skape ideer for videre arbeid.

#### Nils Kr. Rossing

Prosjektleder ved Vitensenteret i Trondheim Dosent ved Skolelaboratoriet ved NTNU e-post: nkr@vitensenteret.com