



Lego Robot



av

[]
[]
[]

INNEHÅLLSFÖRTECKNING

LABB1 (INFORMATION)	#
UPPGIFT 1	#
UPPGIFT 2	#
UPPGIFT 3	#
LABB2 (OPERATÖRER)	#
UPPGIFT 1	#
UPPGIFT 2	#
UPPGIFT 3	#
LABB3 (VARIABLER)	#
UPPGIFT 1	#
UPPGIFT 2	#
LABB4 (FÖRGRENING (IF-SAT))	#
UPPGIFT 1	#
UPPGIFT 2	#
LABB5 (LOOP BLOCK)	#
UPPGIFT 1	#
UPPGIFT 2	#
LITEN CASE	#
LABORATION 1	#
LABORATION 2	#

LABORATION 2

Information

Ansluta sensorer och motorer Motors (motorerna) och Sensors (sensorerna) fungerar bara när de är anslutna till EV3 Brick (EV3-enheten).

Använd de platta, svarta Connector Cables (anslutningskablarna) för att koppla Sensors till EV3 Brick via indataportarna 1, 2, 3 och 4.

Skapar du program när EV3 Brick inte är ansluten till en dator tilldelas Sensors standardportar, vilket sker automatiskt i EV3 Software (EV3- programvaran). Följande standardportar används:

- Port 1: Touch Sensor (trycksensor)
- Port 2: Ingen sensor
- Port 3: Color Sensor (färgsensor) +
- Port 4: Infrared Sensor (IR-sensor)



Om EV3 Brick är ansluten till en dator medan du programmerar, identifieras varje sensors eller motors port automatiskt.

Använd de platta, svarta anslutningskablarna för att koppla motorerna till EV3 Brick via utdataportarna A, B, C och D. Om EV3 Brick inte är ansluten till en dator medan du programmerar, tilldelas varje motor en standardport. Det sker automatiskt, precis som med sensorerna. Följande portar används:

- Port A: Medium Motor (medelstor motor)
- Port B och C: Två Large Motors (stora motorer)
- Port D: Large motor



Om EV3 Brick är ansluten till en dator medan du programmerar, tilldelas rätt port automatiskt i programmet.



1 Port Selector
2 Mode Selector
3 Inputs

Rotations	Degrees
1	360
2	720
0.5	180
1.25	450
7.2	2592

1 Positive Power
2 Negative Power

Positivt tal kör framåt
Negativt tal kör bakåt.

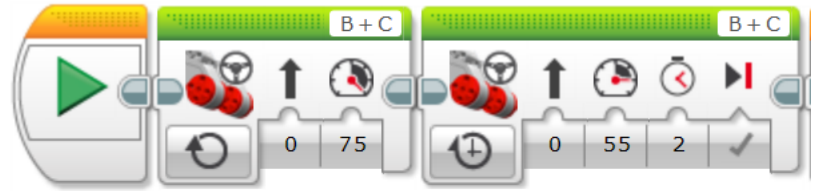
Input	Type	Allowed Values	Notes
Power	Numeric	-100 to 100	Motor power level. See Motor Power and Direction .
Steering	Numeric	-100 to 100	Steering direction. See Steering and Motor Speed .
Brake at End	Logic	True/False	Applies when the block finishes. If True, the motors are stopped immediately and held in position. If False, motor power stops and the motors are allowed to coast.
Seconds	Numeric	≥ 0	Movement time in seconds.
Degrees	Numeric	Any Number	Amount of movement amount in degrees. 360 degrees make a full rotation.
Rotations	Numeric	Any Number	Amount of movement in rotations.

EXEMPEL: ROBOTPROGRAMMERING

Din Lego roboten ska köra framåt och bakåt. Roboten ska köra först framåt. Sedan roboten ska svänga 180 grader och stanna. Hunden skäller

Motorena B och C startar

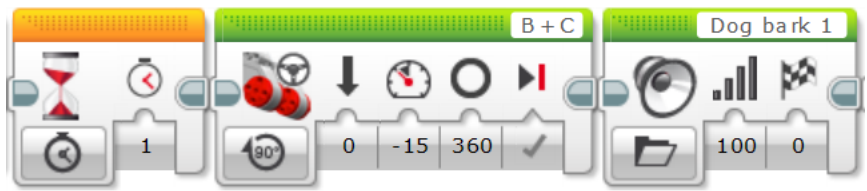
Roboten ska köra framåt (Pilen har värde 0), hastighet 55 och ska rotera 2 sekunder.



Roboten väntar en sekund.

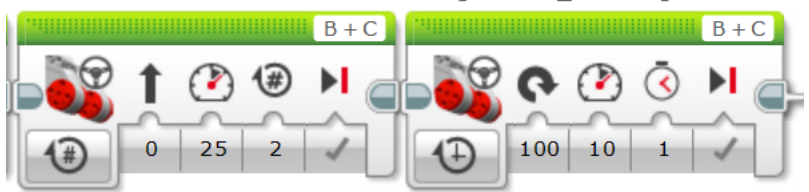
Roboten backar 360 grader. Med hastigheten -15

Hunden skällar



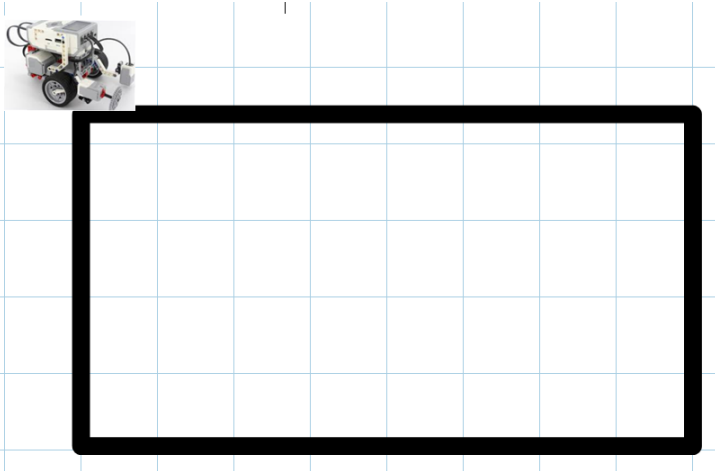
Roboten kör framåt 2 rotationer (360*2). Hastigheten 25.

Roboten vänder en rotation dvs 180 grader med låg hastighet



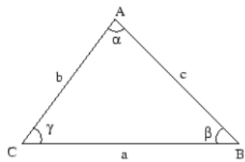
UPPGIFT 1

Skapa ett blockprogram med din Lego robot som kör runt i en fyrkant bana (se bilden). Din robot ska starta från en viss punkt och återvända i samma punkt. Din bana ska vara 75X20 cm.



UPPGIFT 2

Skapa ett blockprogram med din Lego robot som kör runt i en triangelbana. Din robot ska starta från en viss punkt och återvända i samma punkt. Triangelsidorna ska vara olika. Du kan bestämma själva banasmåttet



UPPGIFT 3

Skapa ett blockprogram med din Lego robot som kör runt i en bana. Sätt ett hinder var som hest på banan så att din Lego robot kör rund i det och fortsätter vidare till målet. Din robot ska starta från en viss punkt och återvända i samma punkt. Du kan bestämma själva banasmåttet och form.

LABORATION 2

Operatörer

En operatör är en symbol som berättar för kompilatorn att utföra specifika matematiska eller logiska funktioner. Java script är rikt på inbyggda operatörer och tillhandahåller följande operatörer.

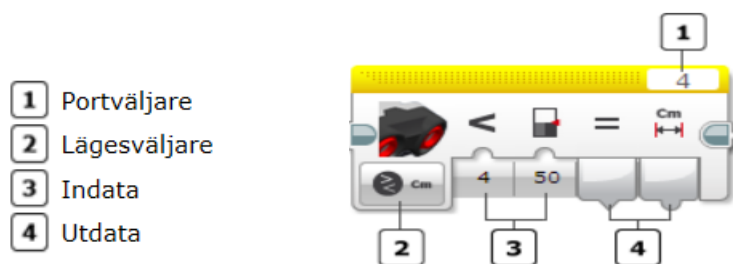
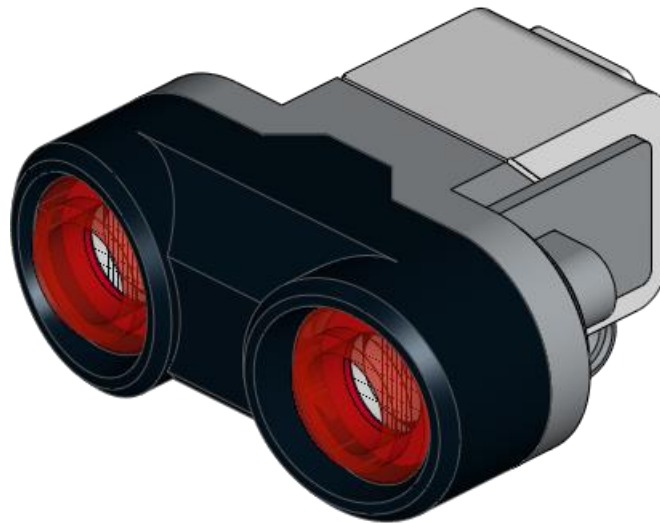
Jämförelse Operatörer

Antag att variabel A =6 och variabel B =12 då

Operatörsnamn	Operatör enkel	Beskrivning	exempel
Lika med	==	Kontrollerar om värdet av två operander är lika eller inte, om ja då blir förhållandet sant.	(A == B) är inte sant
Mindre än	<	Kontrollerar om värdet av vänster operand är mindre än värdet på höger operand, om ja då blir förhållandet sant.	(A<B) är sant
Mer än	>	Kontrollerar om värdet på vänster operand är större än värdet på höger operand, om ja då blir förhållandet sant	(A<B) är inte sant
Mindre än eller lika med	<=	Kontrollerar om värdet av vänster operand är mindre än eller lika med värdet på höger operand, om ja då blir förhållandet sant.	(A<B) är sant
Mer än eller lika med	>=	Kontrollerar om värdet på vänster operand är större än eller lika med värdet på höger operand, om ja då blir förhållandet sant.	(A<B) är inte sant
Inte lika med	!=	Kontrollerar om värdet på två operander är lika eller inte, om värdena inte är lika då villkoret blir sant	(A<B) är sant

Ultraljudsensorns

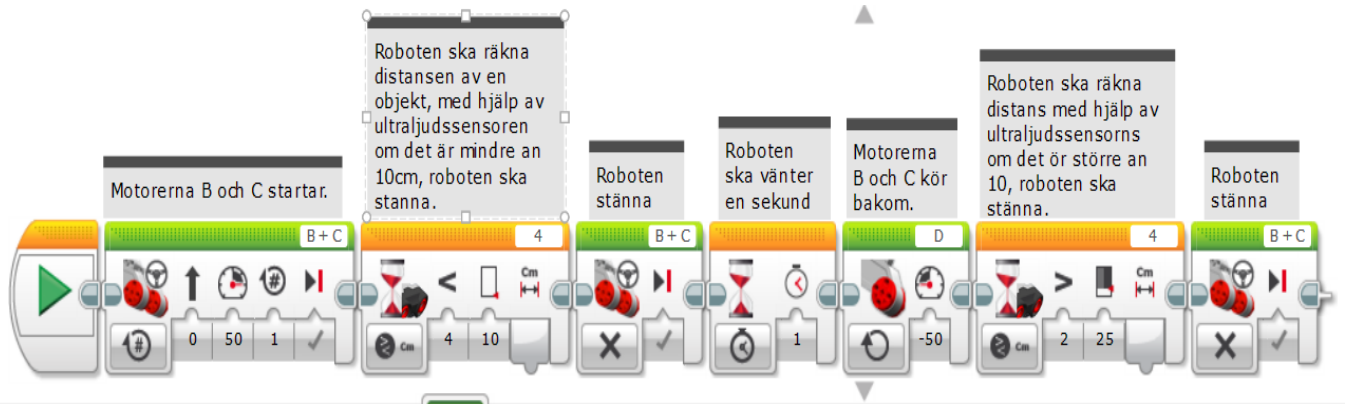
Ultraljudsensorns avger högfrekventa ljudvågor (dessa vågor kan inte detekteras av mänskliga öron eftersom de är för höga) och väntar på att dessa ljudvågor återspeglar ett objekt och beräknar hur lång tid det tar för ljudet att återgå till sensorn. Sensorn jag använde kommer med sändnings- och mottagningsstiften märkta "trigger" respektive "echo". Dessa stift gör det mesta av arbetet för oss.



Utdata	Typ	Anteckningar
Avstånd i centimeter	Numerisk	Avstånd i centimeter (0–255 cm).
Avstånd i tum	Numerisk	Avstånd i tum (0–100 tum).
Ultraljud detekterat	Logik	Sant om en ultraljudssignal detekteras, annars Falskt.
Jämför resultat	Logik	Sant/falskt-resultat för ett Jämför-läge.

EXEMPEL: ROBOTPROGRAMMERING

Använd ultraljudssensorns läge för att vänta på en ändring, för att upptäcka att ett objekt är nära.



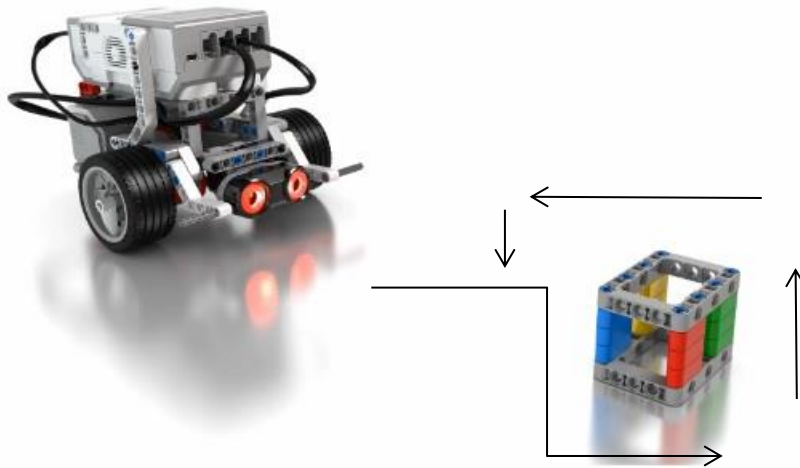
UPPGIFT 1

Skapa ett block program med din Lego robot som används ultraljudssensor för att upptäcka ett objekt. Roboten ska baka 10cm. Sedan vrider roboten 90 gradar och fortsatte sin resa.



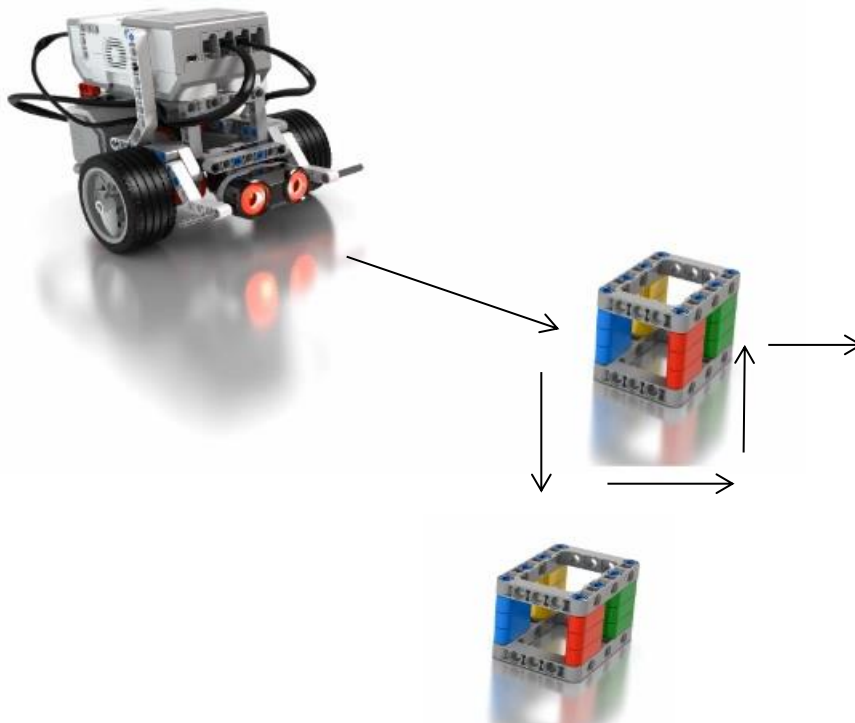
UPPGIFT 2

Skapa ett blockprogram med din Lego robot som används ultraljudssensor för att upptäcka att ett objekt och kör runt kring det.



UPPGIFT 3

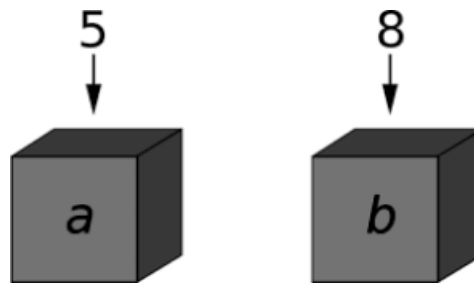
Skapa ett blockprogram med din Lego robot som används ultraljudssensor för att upptäcka att ett objekt och han passerar objekt från höger och tillbaka till samma linje.



LABORATION 3

Variabler

Inom programmering är en variabel ett namngivet objekt som har och kan ändra värde. Det finns flera olika typer av variabler som används i olika sammanhang. Så en variabel ses kanske enklast som en låda, med en etikett på. På etiketten står det ett namn och i lådan ligger det något.



Med hjälp av variabler kan din Lego-robot lagra nummer eller text baserat på inmatning.

Färgsensor

Den digitala EV3-färgsensorn skiljer mellan sju olika färger och kan också upptäcka frånvaron av färg.



Blocket Färgsensor hämtar data från färgsensorn. Du kan mäta färgen eller intensiteten i belysning och få ett numeriskt utvärde. Du kan också jämföra sensordata med ett indatavärde och få ett logiskt (sant eller falskt) utvärde.

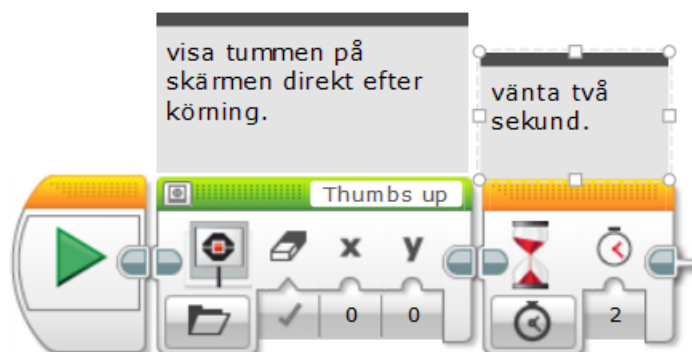
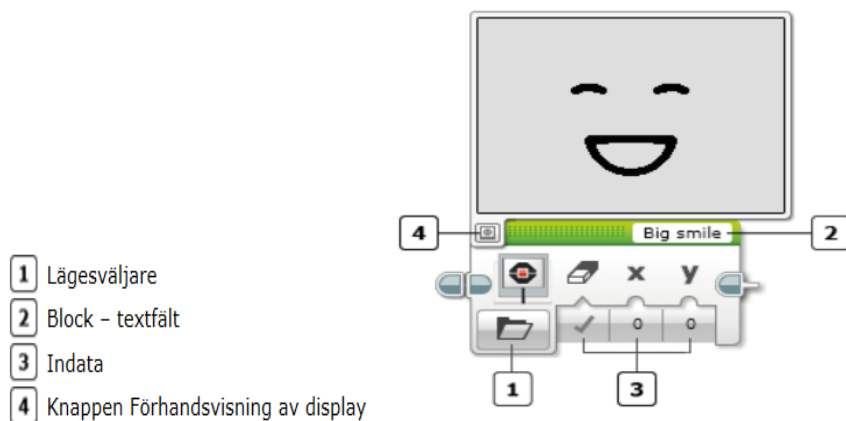
Det fungerar också som en ljussensor genom att detektera ljusintensiteter. Studenter kan bygga färgsortering och radföljande robotar.



Indata	Typ	Tillåtna värden	Anteckningar
Uppsättning färger	Numerisk vektor	Varje element: 0-7	Vald(a) färg(er) att testa för laget Jämför - färg: 0 = ingen färg 1 = svart 2 = blå 3 = grön 4 = gul 5 = röd 6 = vit 7 = brun
Jämför typ	Numerisk	0-5	0: = (lika med) 1: ≠ (inte lika med) 2: > (större än) 3: ≥ (större än eller lika med) 4: < (mindre än) 5: ≤ (mindre än eller lika med)
Tröskelvärde	Numerisk	Valfri siffra	Värde som sensordata ska jämföras med
Värde	Numerisk	0-100	Ljusstyrka för Kalibrera-lagen

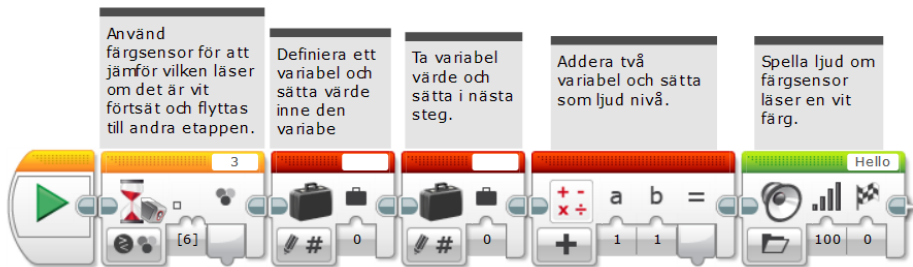
Enhetsdisplay

Använd lego robot enhetens display för att visa bilder och text eller värde från roboten sensor på skärmen.



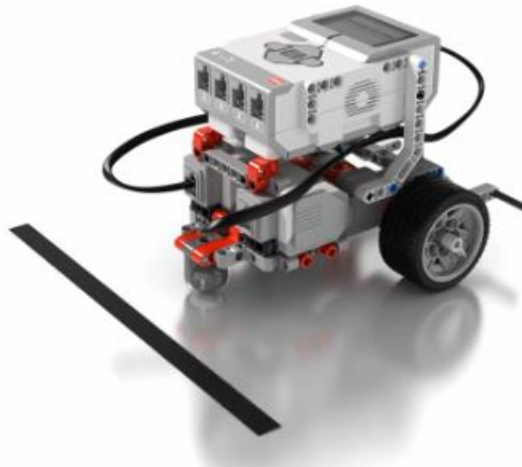
EXEMPEL: ROBOTPROGRAMMERING

Skapa en kod för att göra lego robot läs färg med färgsensor och kontrollera om dess vita färg så att roboten kommer att spela ljud.



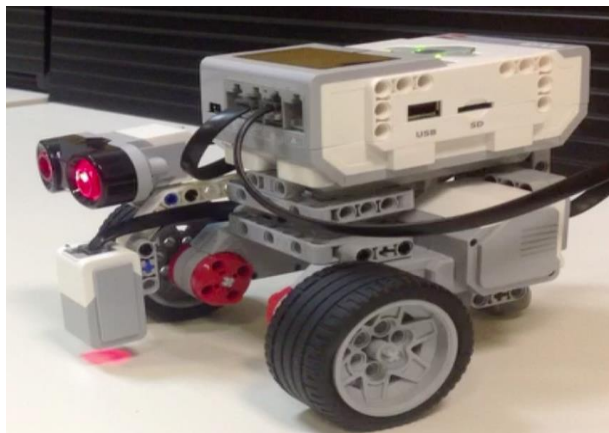
UPPGIFT 1

Skapa ett blockprogram med din Lego-robot som kör rakt och stoppa direkt när färgsensorn hittar en svart linje och visa färg på skärmen.



UPPGIFT 2

Skapa ett blockprogram med din Lego-robot som undersöker bordsskivan, stå och kör bak när du når bordets kant och byter riktning.



LABORATION 4

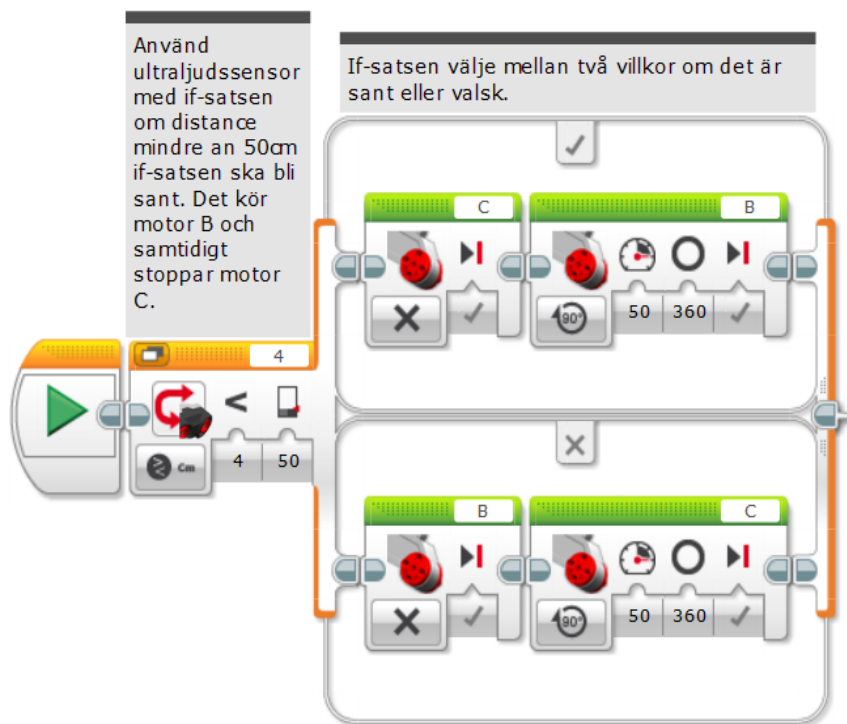
Förgrening (IF satsen)

Blocket Förgrening är en behållare som kan innehålla två eller flera sekvenser av programmeringsblock. Varje sekvens kallas för ett fall. Ett test i början av förgreningen styr vilket fall som ska köras. Bara ett fall åt gången körs varje gång förgreningen körs.

Förgreningstestet som visas här kan avgöra vilket fall som ska köras baserat på ett sensordatavärde eller ett värde från en dataledning. När ett fall har valts och körts fortsätter programmet med eventuella block som följer efter förgreningen.

If satsen är en konstruktion som du kan använda för att styra vad som ska göras i ditt program. Nästan alla program reagerar olika beroende på vad du matar in i dem.

Med följande bild ska vi använda block som kontroll av avstånd mellan något objekt och roboten om det är mindre än 50CM (det är sant) roboten ska kör B motor och stop C motor, också om avstånd är mer än 50CM (det är falsk) roboten ska kör C motor och stop B motor.



- 1 Sant-fall
- 2 Falskt-fall

Trycksensor

Trycksensorn detekterar om knappen på sensorns framsida är intryckt. Med trycksensorn kan du till exempel detektera om roboten kör in i något. Du kan också utlösa en händelse genom att själv trycka på sensorns knapp.



Trycksensorn kan indikera om den är intryckt eller inte. Däremot kan den inte mäta hur länge knappen hålls ner eller hur hårt den trycks in. Trycksensorn genererar logiska data, dvs. värdet Sant eller värdet Falskt. Trycksensorknappens position är densamma som dess status. Om knappen är intryckt är statusen Sant. Är knappen uppsläppt är statusen Falskt.

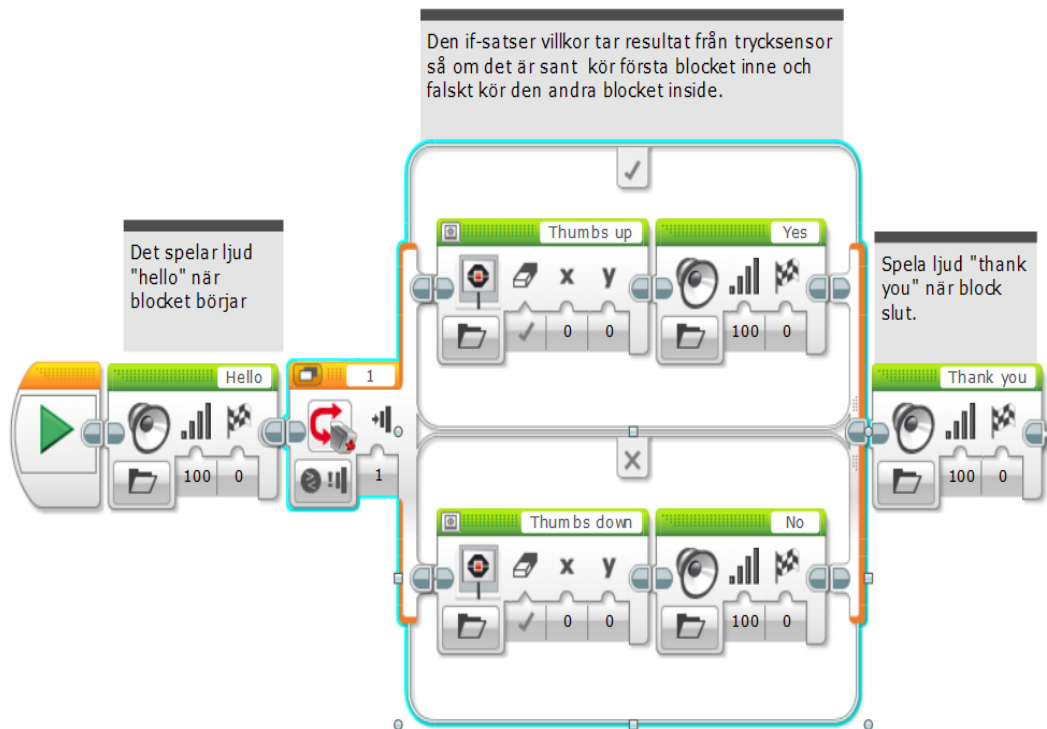
EXEMPEL MED ”INTRYCKT”

Trycksensorn genererar följande data:

Data	Typ	Anteckningar
Status	Logik	Sant om knappen är intryckt, annars Falskt.
Intryckt	Logik	Sant om intryckt, annars Falskt (samma som Status).
Uppsläppt	Logik	Falskt om intryckt, annars Sant (motsatsen till Status).
Snabbtryckt	Logik	Sant om knappen har tryckts in och släppts upp vid ett tidigare tillfälle. Därefter, om en ny snabbtryckning ska lagras, måste en ny intryckning och uppsläppning utföras.

EXEMPEL: ROBOTPROGRAMMERING

Programmet nedan får en robot att säga "Touch" (tryck). Sedan testar det om trycksensorn är intryckt. Om så är fallet körs Sant-fallet (överst), och texten "Thumbs up" (tummen upp) visas på roboten samtidigt som den säger "Yes" (ja). Om trycksensorn inte är intryckt körs Falskt-fallet (längst ned), och texten "Thumbs down" (tummen ned) visas på roboten samtidigt som den säger "No" (nej). Efter förgreningen säger roboten "Thank you" (tack).



Trycksensorn testas så fort roboten har sagt "Touch" (tryck). Om sensorn hålls nedtryckt i det ögonblicket kör förgreningen Sant-fallet. Annars kör den Falskt-fallet.

UPPGIFT 1

Skapa ett blockprogram med din Lego robot som spelar ett visst ljud när du trycker på trycksensor.

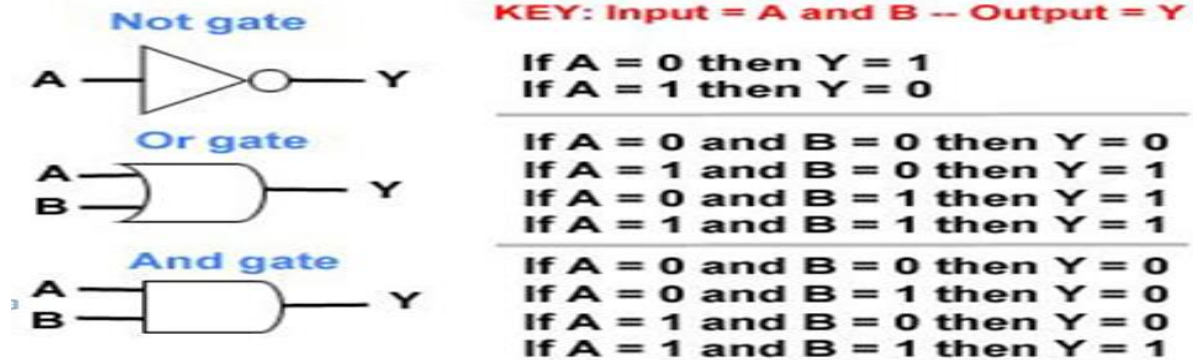


UPPGIFT 2

Skapa ett blockprogram med din Lego robot som kör framåt tills den kolliderar med ett objekt t ex väggen. Trycksensoren trycks ner så att roboten kör bakåt och stannar.



OBS: nästa bild förklara hur logikporten fungerar.



Gyrosensorn

Gyrosensorn identifierar rotationsrörelser. Om du roterar gyrosensorn i samma riktning som pilarna på sensorns hölje, kan sensorn identifiera rotationsfrekvensen i grader per sekund. Du kan använda rotationsfrekvensen för att identifiera till exempel när en del av roboten roterar eller när roboten faller omkull.

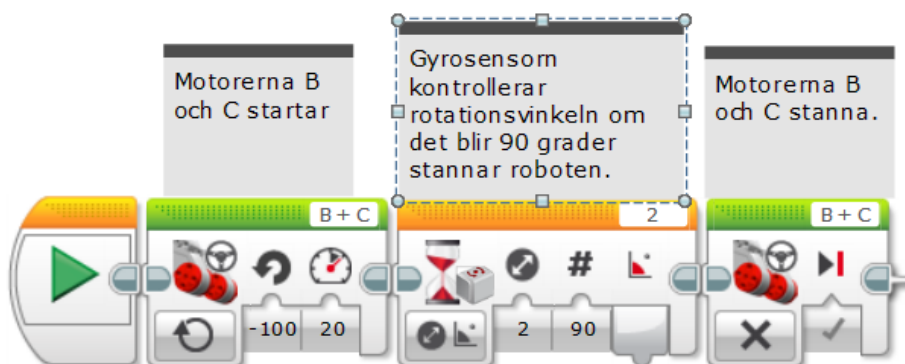
Gyrosensorn registrerar dessutom den totala rotationsvinkeln i grader. Du kan använda rotationsvinkeln om du till exempel vill ta reda på hur långt roboten har roterat.

Gyrosensorn kan generera följande data:

Data	Typ	Anteckningar
Vinkel	Numerisk	Rotationsvinkel i antal grader. Mäts från den senaste återställningen. Återställs med läget Återställ i blocket Gyrosensor .
Frekvens	Numerisk	Rotationsfrekvens i grader per sekund.

EXEMPEL: SVÄNG MED EN SPECIFIK VINKEL

Med det här programmet svänger en robot 90 grader åt vänster. Programmet använder blocket Vänta med läget Gyrosensor – Ändra – Vinkel för att vänta tills rotationsvinkeln ändrats med 90 grader.



UPPGIFT 1

Skapa ett blockprogram med din Lego-robot som spelar ett visst ljud när roboten rotera 90 grader. (Använd gyrosensorn för att mäta rotationsrörelser).



UPPGIFT 2

Skapa ett blockprogram med din Lego-robot som kör runt och rotera när det kom nära väggen och var noga med att roboten spelar ljud när han roterar



LITEN CASE

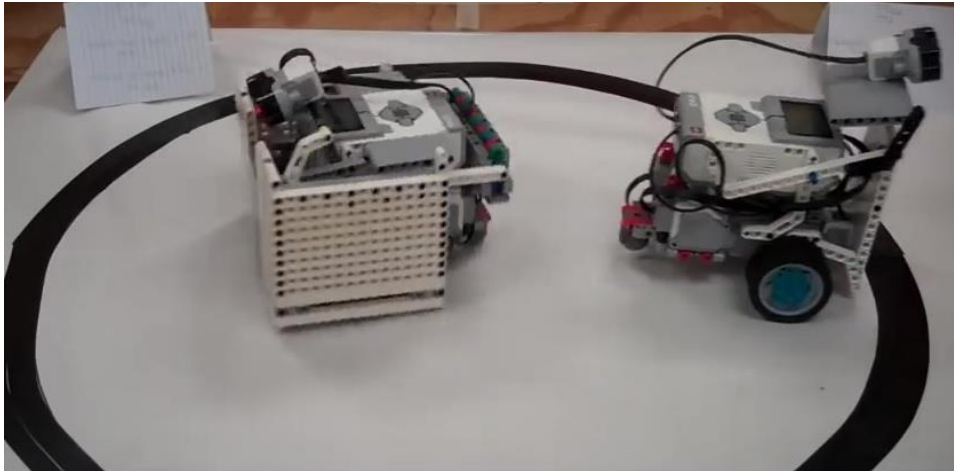
LABORATION 1

Två grupper ska tävla mot varandra samtidigt,

Ni Skapar en blockprogrammering med din lego robot som ska köra så snabbt så möjligt inne i en cirkelbana utan att den lämna ut cirkeln. Roboten som lämnar ut först cirkelbana förlorad.

Tips:

- 1- Använd färgsensor.
- 2- Skapa två loopar.



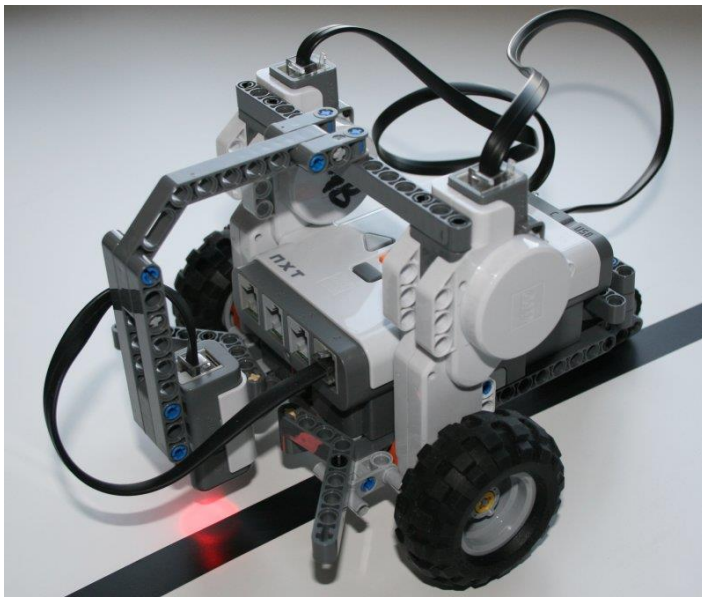
LABORATION 2

Två grupper ska tävla mot varandra samtidigt,

Skapa ett blockprogram med din Lego-robot som följer en svart linje.

Tips:

- 1- Använd färgsensor.
- 2- Skapa variabler.
- 3- Skapa loopar.



LITTERATURFÖRTECKNING

Manual 1:

<https://www.lego.com/sv-se/mindstorms/downloads/user-guide>

Manuel 2:

<https://ev3-help-online.api.education.lego.com/Retail/en-us/page.html?Path=blocks%2FLEGO%2FStartBlock.html>