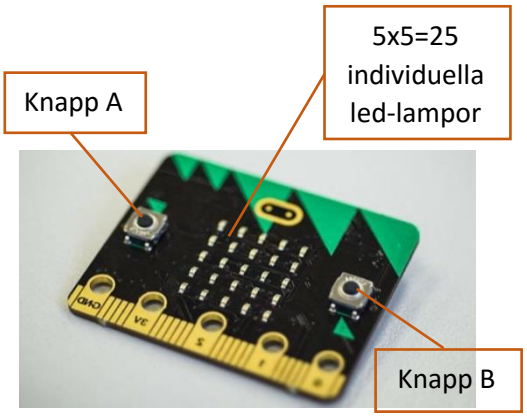
Introduktion till Micro Bit med inriktning på vädermätningar och sensorer

Ray Pörn och Mats Braskén (Åbo Akademi) och Jonas Waxlax (Gymnasiet Lärkan)

Detta är en introduktion till mikrodatorn Micro Bit. Den kan t.ex. användas för att skapa olika typer av enkla spel, elektroniska dekorationer och den kan relativt enkelt programmeras för att fjärrstyra och kommunicera med andra enheter, exempelvis mobiltelefonen eller en annan Micro Bit. Det centrala i denna introduktion är att studera hur olika yttre sensorer och givare kan kopplas till Micro Bit (Micro Bit har även inbyggda sensorer som accelerometer och kompass).

Eftersom grafiska programmeringsverktyg, som t.ex. Scratch, har blivit vanliga i grundskolan kommer vi i denna introduktion att genomgående använda ett motsvarande verktyg för att programmera Micro Bit. Detta medför vissa begränsningar, så har man behov av att utföra mer komplicerade beräkningar bör man använda något textbaserat programmeringsspråk (t.ex. Python eller JavaScript).



Micro Bit (eller micro:bit) är bara fyra gånger fem centimeter stor, men chippet rymmer en fullt fungerande liten mikrodator (ett s.k. inbyggt system). Det är public service-bolaget BBC tillsammans med Microsoft och Samsung (samt många fler) som ligger bakom projektet och den levererades i juni år 2015 till en miljon brittiska sjundeklassare, men har nu också börja säljas till allmänheten. Tanken har varit att skapa en kul, enkel, lärorik och förmånlig mikrodator för skolbruk. Datorn har 25 programmerbara LED-lampor, två programmerbara knappar samt inbyggd accelerometer och magnetometer*. Den kan också ge ut ungefärliga temperaturer och belysningsnivåer. Den har en USB-port, Bluetooth (trådlös kommunikation), fem in- och utgångar som går att ansluta till med krokodilklämmor och ca 20 små pinnar. Datorn kan antingen få ström via USB-kontakten eller sladdlöst i ett par dagar med hjälp av en adapter och två vanliga AA-batterier. Micro Bit kan programmeras med hjälp flera olika verktyg:

- JavaScript
- Python
- C/C++-kod
- Grafiskt verktyg Microsoft Block Editor (liknar mycket Scratch)
- Microsoft Touch Develop.

*En mera noggrann teknisk specifikation fås via Wikipedia med tillhörande länkar https://en.wikipedia.org/wiki/Micro_Bit. En utförlig dokumentation för C/C++-programmerare som innehåller en stor mängd funktionalitet har utarbetats av Lancaster University https://lancaster-university.github.io/microbit-docs/.

I denna introduktion ska vi lära oss att programmera Micro Bit med hjälp av Microsoft Block editorn. Fokus är på att använda mikrodatorn som sensor för att mäta och utforska olika saker i omgivningen kopplade till väder och vind. Strukturen på materialet/övningarna är följande:

1. Det inledande exemplet lär vi oss att skriva ett enkelt program, simulera det i Emulatorn samt spara det på Micro Bit. Vanliga programmeringskommandon presenteras kort.

2. Vi studerar hur Micro Bit kan användas som en vanlig termometer. Temperaturen visas i realtid på displayen.

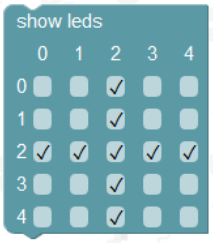
3. Vi gör ett program som mäter ljusstyrkan (illuminansen) med hjälp av Micro Bit. En grov modell konstrueras för att Micro Bit ska kunna ge ut approximativa illuminanser i lux.

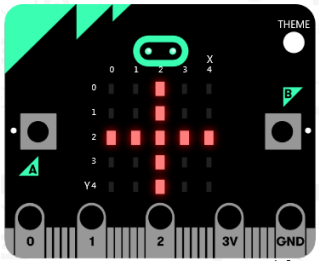
1. Ett första program i Micro Bit

a) Gå till http://microbit.org/ välj **Lets's Code** och **Microsoft Block Editor**

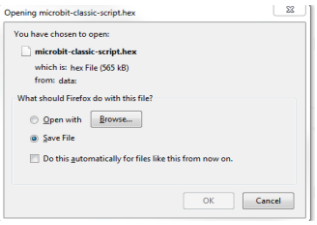
b) Nu öppnas en s.k. **emulator** där man kan programmera och testa sitt program. Då man är nöjd överför man programmet till Micro Bit och kan sedan köra det direkt från den.

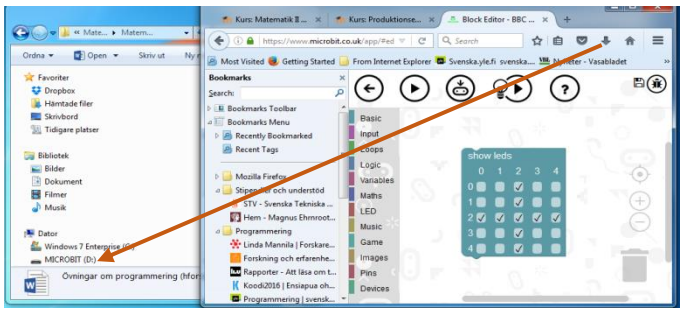
c) Välj **Basic** och dra **show leds** till arbetsytan. Bocka i ledarna som ska lysa (t.ex. ett plustecken +) och tryck på **run**. Nu visas resultatet i Emulatorn till höger.





d) För att överföra detta program (att ett plustecken visas) till Micro Bit utför vi följande steg (Mozilla Firefox).





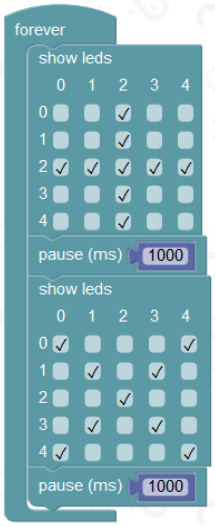
Kompilera → Spara (Save – Ok) → Dra filen från Downloads till MICROBIT (D:).

Nu börjar Micro Bit blinka och när överföringen är klar så slutar blinkningarna. Nu kan vi koppla lös Micro Bit från datorn och den klarar sig själv (batteri behövs).

Att spara kodfilen lokalt

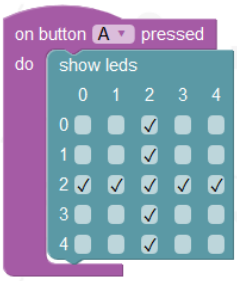
Om man vill spara sitt program lokalt (utanför Block Editorn) görs detta genom att spara en s.k. JSON-fil (extension .jsz) på valfritt ställe. Denna fil kan sedan öppnas via block editorn om man koda vidare.

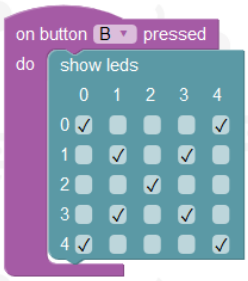
Kort intro till några andra kommandon

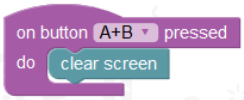


Forever-loop: Konstruera följande blockprogram i editorn. Alla använda block hittas under Basic. Testa att koden fungerar genom att köra programmet i emulatorn (run). Vad gör programmet och hur fungerar det? För över programmet till Micro Bit och testa.

Knapptryck-händelser: Gör ett program som visar ett **+**-tecken då man trycker på knappen A, ett **x**-tecken då man trycker på B och tömmer skärmen då man trycker på A och B samtidigt.







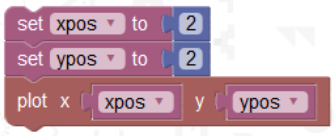
Villkorssatser: Vi skapar ett program som flyttar en enskild LED-lampa runt i 5x5-matrisen. Genom att trycka på A-knappen flyttas markören ett steg åt höger och genom att trycka på B-knappen flyttas den ett steg nedåt. Villkorssatser används för att börja om från rad 0 respektive kolumn 0.

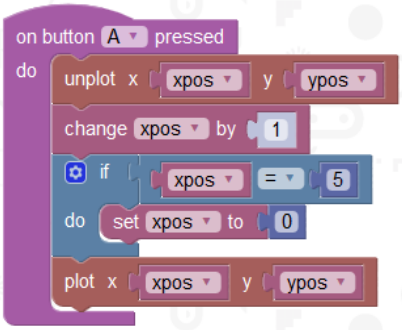
Initialisering av aktiv LED

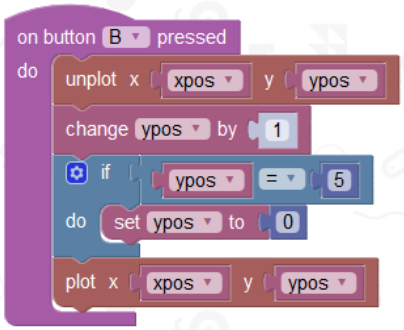
Händelse för tryck på knapp A

- Släck aktiv LED
- Öka x-position med 1
- Om vi når position 5 börja om från början (0)

- Tänd aktiv LED

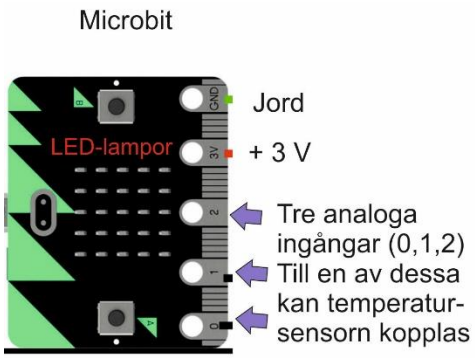


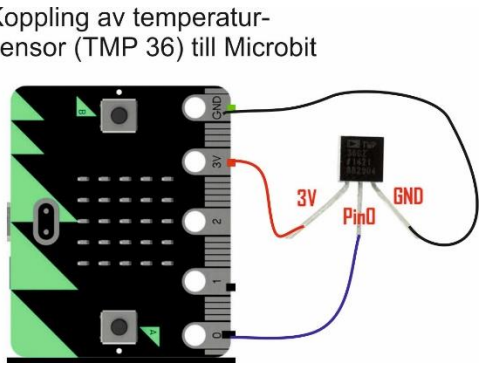




2. Microbit + sensorer (mäta temperatur)

En sensor gör det möjligt att mäta en fysikalisk storhet, så som temperatur, ljusstyrka, tryck eller luftfuktighet. De enklaste sensorerna fungerar så att det finns ett enkelt (ofta linjärt) samband mellan storheten (t.ex. temperatur) och den spänning som sensorn ger ut. Vad vi därför behöver göra för att tillverka en digital termometer, är att läsa av spänningen (vanligen mellan 0 och 5 V) sensorn ger ut med vår Micro Bit. Sensorns spänning tas in på en av Micro Bits tre analoga ingångar (se bilden nedan). Därefter behöver vi skriva några rader kod som tar det inlästa spänningsvärdet, beräknar motsvarar temperatur och därefter visar värdet på Micro Bits inbyggda LED-display.





Att mäta temperatur med hjälp av temperatursensorn TMP36*.

Sensorn TMP36 är temperatursensor som behöver en matningsspänning mellan 2,7 V och 5,5 V för att fungera (Microbit ger ut 3 V vilket alltså duger bra). Temperatursensorns utspänning är direkt proportionell (eller linjär) mot temperaturen i grader Celsius. Noggrannheten i temperaturangivelsen är av storleksordningen $1\ ^\circ\mathrm{C} - 2\ ^\circ\mathrm{C}$ och mätområdet $-40\ ^\circ\mathrm{C}$ till $125\ ^\circ\mathrm{C}$ (graf **b** nedan). Sambandet mellan temperatur (T) och utspänning (U) ges av linjen **b** i grafen nedan. Linjens riktningskoefficient är $10\ \mathrm{mV}/^\circ\mathrm{C}$ och linjen passerar exakt genom punkten $(50\ ^\circ\mathrm{C}, 1000\ \mathrm{mV})$.

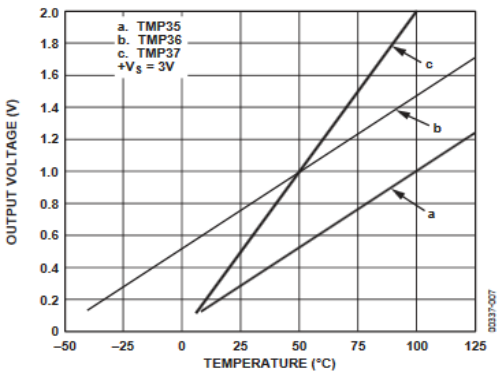
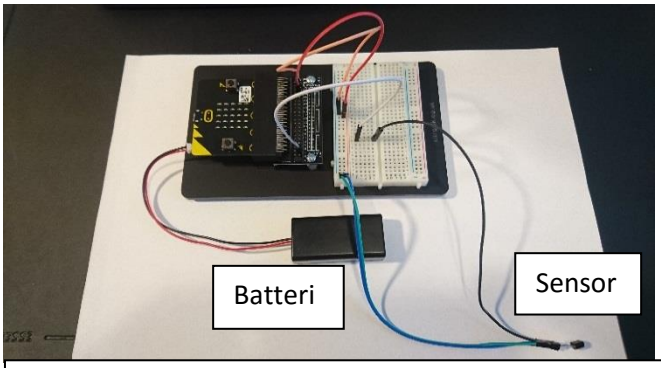


Figure 6. Output Voltage vs. Temperature



Sensor och batteri kopplad till Microbit. I bilden har en kopplingsbräda använts för att förenkla kopplingsarbetet.

**Denna komponent kan enkelt köpas på nätet (styckepris 2,30 €), t.ex. från Robomaa:*

http://robomaa.fi/tmp36-temperature-sensor

Sambandet *T* och *U* blir alltså:

$$U - 1000\ \mathrm{mV} = 10\ \mathrm{mV/°C}(T - 50\mathrm{°C}) \rightarrow$$

$$T = \left(\frac{U - 500}{10}\right)\mathrm{°C}$$

där spänningen *U* bör ges i milli-Volt (mV).

Ingången på Microbit (pin 0) registrerar temperatursensorns spänningsvärde i intervallet 0 V till 3 V med en upplösning på 1024 steg. Spänningen 0 V motsvarar alltså värdet 0 och spänningen 3 V motsvarar värdet 1023. Omvandlingen från Microbitvärde (mb) till spänningsvärden (U) sker då enligt:

$$U = \frac{3000}{1024} \times mb$$

där enheten för spänningen U är mV. Stegen från spänning till temperatur blir då:

1. Avläsning av analog ingång (pin P0) för att få Microbitvärdet mb.
2. Beräkning av motsvarande spänningsvärde i millivolt: $= \frac{3000}{1024} \times mb$.
3. Beräkning av motsvarande temperatur med hjälp av: $T = \frac{U-500}{10} = \left(\frac{U}{10} - 50\right)$ °C.

Exempel

1. Microbit avläser värdet *mb* = 230 på pin P0.

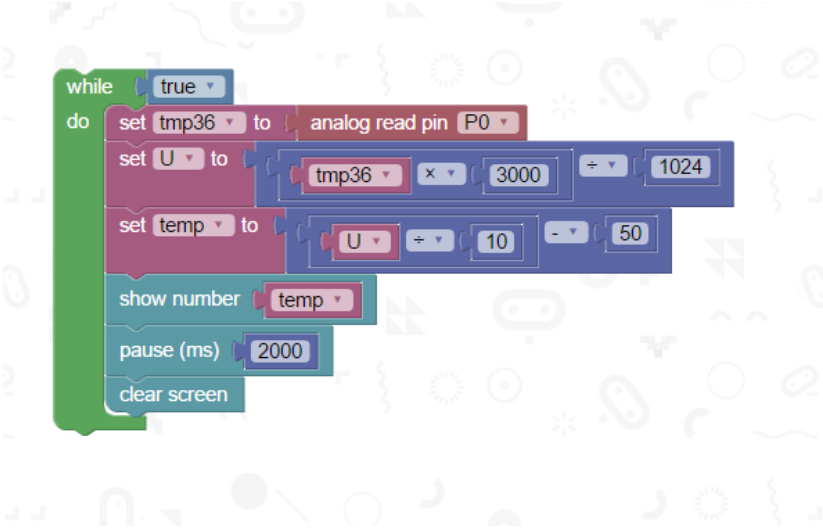
2. Detta värde motsvarar en spänning på: $U = \frac{3000 \times 230}{1024} = 673\ \mathrm{mV}$

3. Vilket motsvarar en temperatur på: $T = \frac{673-500}{10} = +17\ \mathrm{°C}$

Observera att Microbit använder sig av heltalsaritmetik, varför det är bättre att multiplicera före man dividerar och använda sig av tillräckligt stora heltal (därför enheten millivolt). Microbitkoden för en enkel termometer som visar den aktuella temperaturen, samt lagrar temperaturens extremvärden (max, min), ges på följande sida.

Koden består av en oändlig loop, i vilken temperatursensorvärdet läses av från pin P0, spänningen U beräknas, varefter temperaturen beräknas och visas. Efter en paus på 2 sekunder (2000 ms) töms skärmen och ett nytt temperaturvärde läses in.

Exempel på en Microbitkod som läser av temperatursensorn TMP36 på pin 0.



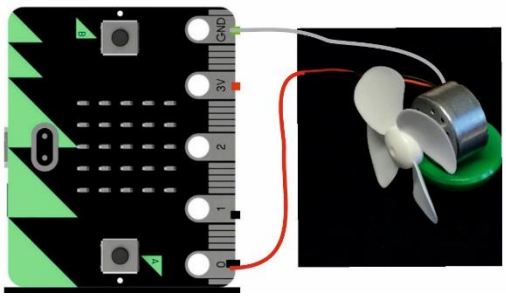
Andra sensorer vilka fungerar på liknande sätt som TMP36

De enklaste sensorerna att använda är av samma typ som TMP36 ovan. Denna typ av sensorer ger ut en spänning *U* (vanligen 0 – 5 V) som är direkt proportionell mot den storhet *Y* man vill mäta. Sambandet kan uttryckas som $Y = aU + b$, där *a* och *b* är konstanter (i fallet med vår temperatursensor är *a* = 1/10 och *b* = -50). Hittar vi en sådan sensor, vilken mäter den storhet vi vill mäta, kan vi direkt använda samma kod som ovan, endast byta ut konstanterna *a* och *b*.

Att mäta vindhastigheten

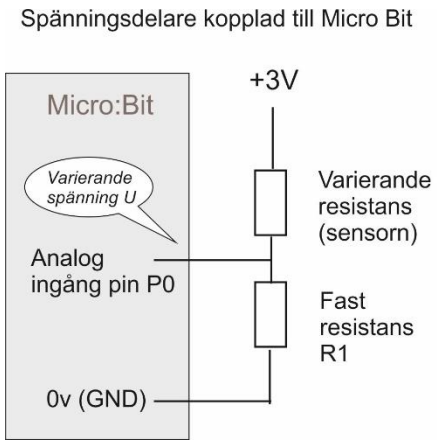
En vanlig likströmsmotor (DC-motor) fungerar som en elgenerator om man snurrar motorns axel. Till första approximation gäller ett linjärt samband mellan motorns rotationshastighet (ω) och spänningen (*U*): $\omega = aU + b$. Kopplar man en propeller till elmotorns axel och låter vinden blåsa, så kommer rotationshastigheten (och därmed spänningen ut) att bero av vindhastigheten (*v*). Antar vi att sambandet mellan rotationshastighet och vindhastighet också är linjärt, får vi att $v = \bar{a}U + \bar{b}$. Utmaningen är att hitta konstanterna $\bar{a}$ och $\bar{b}$, vilket enklast görs experimentellt genom att mäta spänningen *U* vid kända vindhastigheter *v*.

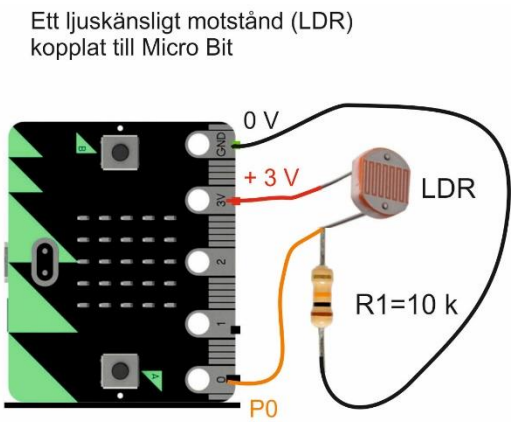
Koppling av DC-motor med propeller till Microbit



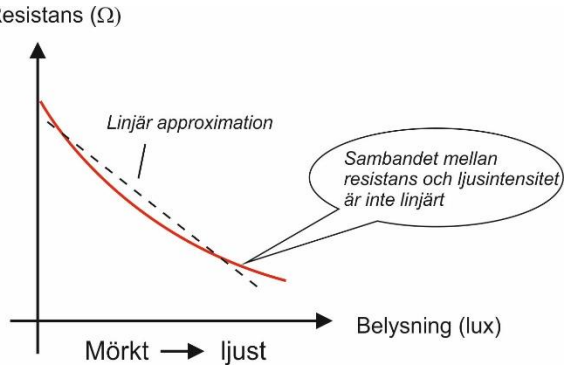
2. Microbit + sensorer (mäta ljusintensiteten)

En annan stor klass av sensorer ger inte direkt ut en spänning, utan fungerar så att sensorns resistans ändras när någon yttre faktor ändrar (t.ex. temperaturen eller ljusintensiteten). För att använda oss av denna typ av sensorer med Micro Bit, måste vi omvandla den varierande resistansen till en varierande spänning. Detta kan enkelt göras med hjälp av en s.k. spänningsdelare (se bilden nedan). När sensorns resistans *R* ändrar, ändrar strömmen *I* genom sensorn och därmed (enligt Ohms lag) också spänningen *U* över sensorn. Spänningen *U* kan vi därefter läsa av med vår Micro Bit på samma sätt som i fallet med temperatursensorn tidigare.





Ett fotomotstånd eller en LDR (Light Dependent Resistor) är en komponent vars resistans varierar med intensiteten hos det ljus som belyser komponenten. En LDR kan därför fungera som en enkel ljusdetektor*. Då ljusets intensitet ökar kommer resistansen hos LDR:n att minska och omvänt om ljusintensiteten ökar. För komponenten vi valt varierar resistansen mellan $8\ \mathrm{k\Omega}$ (stark belysning) → $20\ \mathrm{k\Omega}$ (svag belysning) och 1 MΩ (helt mörkt). Sambandet mellan resistans och belysningen i lux är enligt tillverkaren (8 ~ 20) lux/kΩ (notera att sambandet inte är linjärt). En mulen dag motsvarar en belysning på ca. 1000 lux, medan direkt solljus motsvarar 32000 – 130 000 lux. En klar natt utan måne motsvarar 0,002 lux.

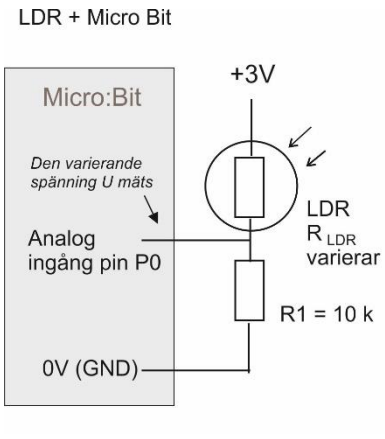


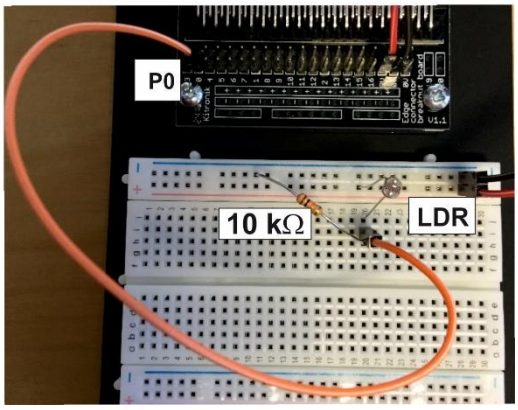
* LDR-motståndet vi använder har typbeteckning PGM5526 och kan enkelt beställas på nätet från t.ex. Vekoy för 0,25 €/styck: http://www.vekoy.com/product_info.php?products_id=26006

Då en LDR kopplas i serie med ett fast motstånd av lämplig storlek får vi en spänningsdelare (se första figuren på andra sidan). I vår koppling används ett motstånd $R1 = 10\ \mathrm{k\Omega}$. Motståndet R1 och LDR:n delar alltså på spänningen 3 V. Om LDR:n har en resistans på $8\ \mathrm{k\Omega}$ (dagsljus) så är spänningen över motståndet R1 enligt Ohms lag:

$$U = \left(\frac{R1}{R_{LDR} + R1}\right) 3\ \mathrm{V} = \left(\frac{10}{8 + 10}\right) 3\ \mathrm{V} \approx 1{,}67\ \mathrm{V}$$

När ljusintensiteten minskar ökar resistansen hos LDR-motståndet. Om resistansen är $20\ \mathrm{k\Omega}$ vid skymning, blir spänningen över motståndet R1 lika med $U = 1{,}0\ \mathrm{V}$. Är det helt mörkt ($R_{LDR} = 1000\ \mathrm{k\Omega}$) blir spänningen *U* över motståndet R1 nästan 0 V (≈0,03 V). Liksom i fallet med temperatursensorn avläser vi spänningen *U* på pin P0.

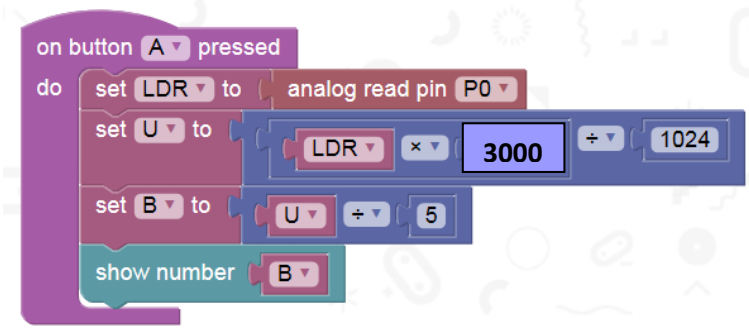




Direkta mätningar ger att för PGM5526 får vi att 500 lux motsvarar 2500 mV, varför algoritmen blir:

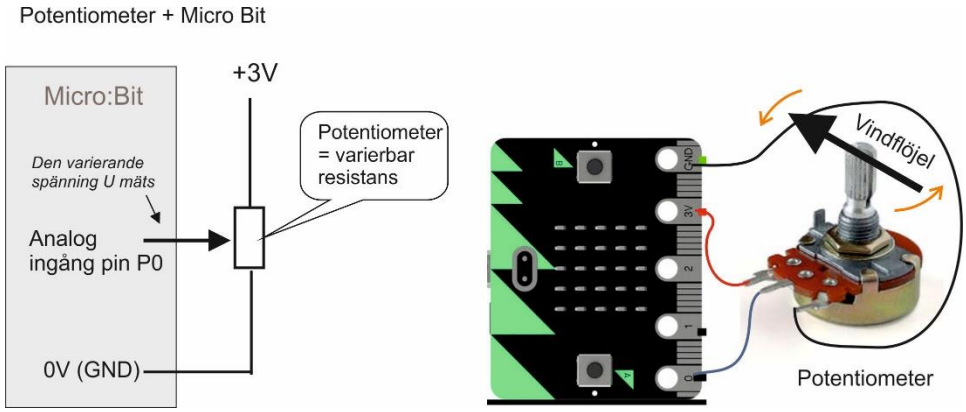
1. Avläs analog ingång (P0) för att få Microbitvärdet mb.
2. Beräkna motsvarande spänning: $= \left(\frac{3000}{1024} \times mb\right)\ \mathrm{mV}$.
3. Beräkna belysningen: $= \left(\frac{500}{2500} \times U\right) \mathrm{lux}$.

Algoritmen kan direkt översättas till följande kod:



Att mäta vindriktningen

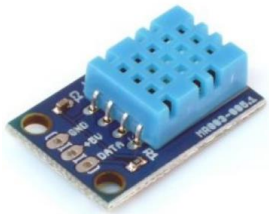
En spänningsdelarkoppling kan användas i flera sammanhang. Ett sådant är att använda sig av en varierbar resistans, en s.k. potentiometer, för att mäta vindriktningen. Tanken är att potentiometern kopplas till en vindflöjel, så att när vindflöjeln vrider sig med vinden, vrids också potentiometern. När potentiometern vrids ändrar resistansen och därmed och spänningen *U* som läses in på pin P0 (se bilden nedan).



3. Microbit + sensorer (mera komplicerade sensorer)

Tyvärr kan inte alla intressanta fysikaliska storheter mätas med hjälp av de sensortyper vi behandlat. Två exempel är sensorer vilka mäter lufttryck och luftfuktighet. Problemet är inte att sensorerna dyra, utan att de har en mer komplicerad output än bara en varierande spänning eller en varierande resistans.

Trycksensor BMP180



Fukt- temperatursensor DHT11

Den goda nyheten är att dessa mer komplicerade sensorer kan kopplas till Micro Bit. Kommunikationen mellan sensorerna invid och Micro Bit använder sig av ett dataöverföringsprotokoll som heter I2C. För att förstå vad sensorn läser av och skickar ut, räcker däremot inte de verktyg som finns i den grafiska programmeringsmiljön vi använt hittills, utan man bör gå över till ett textbaserat programmeringsspråk som Python eller Javascript (båda stöds av Micro Bit). Ett kanske bättre alternativ, som gör övergången från grafisk till textbaserad programmering mjukare, är att använda sig av ”Sniff” (se http://www.sniff.org.uk/). Sniff klarar också av att läsa av sensorer av typen DHT11.

CC BY NC ND