



# Fakta och praktiska tips

## Smartare produkter







## Inledning

Dokumentet innehåller fakta, handledning, praktiska tips mm samt stöd till de laborativa momenten inom digitalteknik för ämnet Teknik på grundskolan. Materialet är avsett för främst NTA:s temautbildare samt lärare och pedagoger i NTA-temat Smartare Produkter. Delar av innehållet kan mycket väl anpassas för eleverna där läraren finner det lämpligt. I ett NTA-tema kallas en uppgift för ett uppdrag.

Uppdragen i de båda temana omfattar moment ur läroplanen, som exempelvis:

- Tekniska lösningar som utnyttjar elektriska komponenter och enkel elektronik för att åstadkomma ljud, ljus eller rörelse, till exempel larm och belysning
- Hur komponenter och delsystem samverkar i ett större system
- Tekniska lösningar för styrning och reglering
- Samspel mellan människa och teknik, samt människans möjligheter att skapa tekniska lösningar som bidrar till hållbar utveckling.
- Dokumentation och terminologi

#### Micro:bit och Teknik

De praktiska momenten i de båda NTA-temana utgår från en s k plattform, ett datorsystem, som kallas micro:bit. En micro:bit är en enkortsdator och kan användas som ett inbyggt system i produkter. Flera namn på samma tekniska system, som är komplext men lättanvänt och innefattar ett antal kompetensområden inom digitalteknikområdet. Att arbeta med och lära sig tekniken runt ett inbyggt system innefattar områdena elektronik, datateknik, datavetenskap (programmering) samt kunskap om gränssnittet till sensorer och ställdon (lampor, motorer etc.) som ska anslutas till systemet. Förutom det rent tekniska innebär det också att förstå

andra behov som olika applikationsområden ställer på ett elektroniksystem som ska byggas in en produkt.

Micro:bit är framtaget för utbildningsändamål på grundskolenivå, men skulle i princip kunna användas som ett system för exempelvis att styra en tvättmaskin och andra vitvaror eller ingå som ett elektroniksystemen i ett fordon. Användningsområdena är många, det är bara fantasin som sätter gränser.

Användningen av elektronik är i högsta grad branschöverskridande och utgör en allt större och viktigare del av de produkter och tjänster som utvecklas. I kombination med programvara är systemen ett oslagbart verktyg i takt med behovet av nya lösningar för att möta de globala utmaningar världen står inför. Det kan gälla minskad energiförbrukning, hållbar miljöutveckling, vård och omsorg för en åldrande befolkning, bara för att nämna några.







## Smartare Elektroniksystem

ELECTRONIC COMPONENTS & SYSTEMS

# Innehåll

Inledning	2
Micro:bit och Teknik	2
Bli bekant med Micro:bit och programmeringsmiljön	4
Micro:bit kopplad till en dator	4
Tema Smartare produkter	6
Elektroniksystemet med komponenter som finns med i vårt tema	6
Uppdrag 2 – 5	6
Exempel programmering	6
Mer om displayen	7
Flödesdiagram	7
Symboler flödesdiagram	8
Mäta temperaturen (uppdrag 5)	9
Ladda ner program till Micro:bit	10
Kan något gå fel?	10
ESD – fakta mm	11
Praktiska tips och trix	12
Före lektion	12
Under lektion	12
Efter lektion	12
Micro:bit och surfplattor med trådlös överföring av program	12
Support – vad finns på nätet?	12
Uppdrag 6	13
Mjukvara	13
Sensor i uppdrag 6	14
Uppdrag 7	14
Mjukvara	14
Uppdrag 8	15
Mjukvara	15
Forts uppdrag 8: Mer om hårdvaran lysdiod – ljussensor hos Micro:bit	16
Sammanfattning	17
Utvecklingsmiljön för Micro:bit	17
Systemet Micro:bit	18
Mer fakta vid hantering av Micro:bit-kortet	19
Talsystem	20
Decimala talsystemet	20
Binära talsystemet	20



#### Smartare Elektroniksystem ELECTRONIC COMPONENTS & SYSTEMS



## Bli bekant med Micro:bit och programmeringsmiljön

Figur 1a och 1b: Micro:bit-kortets fram- respektive baksida

#### Micro:bit kopplad till en dator

- Förbered din dator för programmering av en micro:bit
  - o Koppla in micro:bit-kortet till ett USB-uttag på din dator
  - Skapa ett bibliotek på datorn där du vill spara dina filer, dina program
  - o Datorn ska vara ansluten till Internet
  - Använd gärna webbläsaren Google Chrome
  - o ...
- För att komma igång med att programmera micro:bit följer några steg på vägen
  - o Gå till länken <u>https://microbit.org</u>
  - Välj Lets code (adress <u>https://microbit.org/code/</u>)
  - o ...
- Välj Let's code igen under MakeCode Editor och fortsätt eventuellt med att öppna ett nytt
  Projekt (Projects) om något "gammalt projekt" dyker





#### Nytt fönster - utvecklingsfönstret

- Nu kommer det upp ett nytt fönster (figur 2) med tre fält, varav ett är **programmeringsytan** för block- eller kodprogrammering
  - o I denna rullist, välj språk (Langauge) klicka och välj Svenska om du tycker det är bättre
- Notera även de två övriga ytorna:
  - Simuleringsbild, där en första test kan göras av det program man skapar
  - o ... men först ska programmet skapas med block från Kodbiblioteket



 När du känner dig klar med din kod har du möjlighet att direkt Simulera ditt program och se resultatet i simuleringsfönster i figur 2 ovan.

Simulatorn i programmeringsmiljön har utvecklats och förbättrats och är ett bra verktyg för att analysera det program som utvecklats. Knapparna under simulatorfönstret har olika funktioner. När du börjar med de första uppdragen, testa dessa för att bli bekant med deras funktion.





## Tema Smartare produkter

Under det här avsnittet av temat beskrivs micro:bit-systemets komponenter eller delsystem utifrån uppdraget det handlar om. Se bilden nedan, där flera av delsystemen finns med och används i temat Smartare Produkter. Förutom fakta om komponenter eller delsystemen finns även exempel på programblock som kan användas som hjälp till uppdragen i temat. Vi återkommer till bilden nedan senare i dokumentet.

#### Elektroniksystemet med komponenter som finns med i vårt tema



#### Uppdrag 2 – 5

I dessa uppdrag används micro:bitens lysdiodmatris (LED lights) för att skriva olika tecken och texter. Displayen består av 25 lysdioder med rött ljus, vilka är monterade i en matris om 5 x 5 dioder.

#### Exempel programmering

Med instruktioner (=block) under **kategorin Grundläggande** samt **more** kan olika tecken eller texter skrivas på displayen. I bilden ses några exempel på text och tecken.

Notera också

- att det som görs under funktionen vid start exekveras (=körs) bara en gång när programmet startar.
- att det som görs under funktionen för alltid exekveras (=körs) om och om igen så länge som programmet är igång.





Testa gärna Simulatorns olika möjligheter med knapparna under från vänster:

- starta programmet
- starta om programmet
- kör i "slow motion", mycket användbar vid felsökning
- visa helskärm (5:e knappen)

#### Mer om displayen

Med instruktioner under **kategorin Led** kan enskilda dioder tändas eller släckas, exempelvis **tänd x 3 y 4.** I detta fall tänds lysdioden med koordinaten (3,4).



Dioderna i displayen är placerade i ett koordinatsystem (x,y) med origo i övre vänstra hörnet (0,0) och dioden (4,4) i det nedre högra diagonalt. Talen 0, 1, 2, 3, 4 anger som sagt de fem positionerna i varje riktning.

Med blocket **växla x 4 y 4** i kombination med **pausa** kan en diod, i detta fall (4,4), fås att blinka med vald frekvens.

Det finns fler block under **Led (... more)**, exempelvis att ställa in ljusstyrkan på texten, **ställ in ljusstyrka tal**, där **tal** ska vara i intervallet 0 – 255.

Allt beskrivs inte det här dokumentet, men prova gärna fler block under **Grundläggande och Led** i "egna tester". För det mesta går det att köra programmen i simulatorn innan nedladdning.

#### Flödesdiagram

Programmet nedan till vänster är lite förenklat ett exempel från trafiken där hastigheten mäts och skrivs ut på en display. Om vi skulle göra en beskrivning i förväg på vad som ska hända, dvs hur och vad som ska skrivas ut, kan det göras med ett s k flödesdiagram. Se flödesdiagrammet i bilden till höger och jämför med koden. Mer på nästa sida!



**Hastighet** är en variabel som har det senaste värdet på hastigheten. En ny mätning på nästa bilist ger ett nytt värde och en ny utskrift, eftersom programmet löper runt i en oändlig loop ("för alltid").

#### Smartare Elektroniksystem Electronic components & systems







#### Forts. Flödesdiagram

Se uppdrag 2 i Smartare produkter. Utifrån specifikationen kan ni börja att utveckla programmet tillsammans. Se bild och punktlista nedan som kan komplettera arbetsgången och vara ett alternativ till bildspelets arbetsgång vid införandet av flödesdiagram vid bild 10:



- 1. Specifikation av uppdraget
- 2. Ett första utkast till lösning utifrån specifikation
- 3. Flera problem att lösa finns kvar, exempelvis:
  - a. att text mellan bilder på display flyter ihop.
  - b. att texter/info kommer för tätt, ingen paus i informationen
  - Blinka, hur löses det? c.
- 4. Utkast 3/4
- 5. Mycket att hålla reda på
  - a. inför en metod att planera programmet
  - b. inför flödesdiagram
- 6. osv... följ bildspelets arbetsgång och programmet blir klart.

#### Symboler flödesdiagram

När ett program planeras kan ett verktyg, en metod, som kallas flödesdiagram användas. I bilden till höger ses några olika standardiserade symboler för detta.

I Smartare Produkter används inte alla symbolerna i bilden. Om tid finns, öva gärna på dessa symboler i kommande uppdrag då program ska utvecklas eller analyseras.



Start Slut

utför

Fråga



Jal (Sant) Två utgångar) Nej! (Falskt) Indata

Pilar för att

länka ihop

aktiviteterna



micro:bit

#### Mäta temperaturen (uppdrag 5)

Temperatursensorn som används i uppdrag 5 sitter integrerad i Micro:bitens processor. Mätområdet är därmed begränsat till att visa rumstemperatur och dess variationer. Det är svårt att värma sensorn utan att riskera att processorn kan gå sönder, möjligtvis att med försiktighet värma med ett finger eller med andningen, men det stannar då vid ca 30 grader Celsius.

För uppdrag 5 i temat finns det en färdig funktion för temperaturen under **kategorin Input**, den heter **temperatur (°C).** Funktionen returnerar ett temperaturvärde som kan användas dels för att skriva ut temperaturen i klartext direkt eller för att tilldela **en variabel** detta värde för att användas senare i programmet. Kanske för att räkna ut ett medelvärde över tid eller visa vilken maxtemperatur det varit i rummet under en period. Se exempel bredvid på en programstomme där

variabeln kallas **Temp**.

I nästa tema kommer medel- och maxtemperatur att finnas med i ett av uppdragen där en annan sensor används.

Programmet i bild går att simulera!

#### Mer programmering

En annan nyhet i programmeringen av uppdrag 5 är en s k loop, en repetition av en programsekvens. Det innebär att sekvensen återupprepas så många gånger man vill, i vårt fall ska temperaturen visas två gånger. Se bildspelet på sidan 29 exempelvis.

En variant på detta är att använda en annan loop, **medan sant – gör.** Programmet till höger upprepar också två gånger och notera att ett startvärde för variabeln Antal sätts innan loopen. Sekvensen körs så länge som Antal är mindre än 3. Eftersom Antal räknas upp med ett vid varje varv blir det 2 ggr som sekvensen körs.

Vad skulle behöva förändras för att upprepa 4 gånger?

Svar: Villkoret medan Antal<5 ska det stå i så fall.







## Ladda ner program till Micro:bit

Nästa moment i uppdraget är att kunna ladda ner ett program till systemet, dvs till micro:biten. Innan du/ni börjar med att ladda ner program i uppdragen 5 – 8, titta igenom punkterna nedan.



annat bibliotek (mapp) på datorn, ex. vis Uppdrag\_5.

- Leta reda på HEX-filen och "droppa" filen till micro:bit-systemet (filen förs över till micro:bit på samma sätt som man för över en fil till ett USB-minne).
- När hex-filen laddats ner exekveras programmet automatiskt och du kan testa det du gjort.
- Med Reset-knappen på micro:bit-kortets baksida kan programmet startas om.
- ...

#### Kan något gå fel?

- Kanske du eller eleven råkat trycka in resetknappen samtidigt som micro:bit ansluts till datorns USB?
- Micro:bit hamnar då i s k underhållsmod
  - Maintenance mode
- Inget applikationsprogram (HEX-fil) kan då laddas ner i micro:bitens Flash-minne.



• Åtgärd: Dra ur kabeln och anslut den igen utan att trycka på Reset-knappen i detta moment.



ESD – fakta mm

ESD är ett viktigt fenomen att ta hänsyn till vid såväl tillverkning som användning av elektronikkort, s k kretskort. ESD betyder **E**lectro**S**tatic **D**ischarge och innebär exempelvis att du som användare kan ladda ur statisk elektricitet via elektronikkortet och därmed skada eller förstöra kretskortet.

Micro:bit är konstruerat för säker användning men ska hanteras med viss försiktighet i vissa avseenden.

Det finns inget omgivande skydd, här några punkter att beakta:

#### ESD

- Det är viktigt att undvika statisk elektricitet.
- Förvara alltid din micro:bit i den antistatiska påsen när den inte används.
- Rör gärna vid ett metallföremål som är anslutet till jord innan du börjar med de praktiska uppdraget i Temat. Det kan vara metallbenen på den stol du sitter på.

#### Övrigt

- Lägg inte metallföremål över anslutningar, de kan kortsluta signaler och störa funktionen
- Använd bara batterier och USB-kabel som är avsedda för systemet
- Micro:bit ska inte bli varm vid användning. Om den blir varm, dra ur USB eller annan strömförsörjning för att kontrollera att det är rätt batterier, rätt ansluten USB osv. Det är ovanligt att micro:biten går sönder, men en hårdhänt hantering kan givetvis skada elektroniken.
- Använd inte bärbara batteriladdare eller USB-laddningsuttag för att driva din micro:bit.

Se länken <u>https://microbit.org/guide/safety-advice/</u> för mer information och vägledning vid användandet av micro:bit!













## Praktiska tips och trix

Under detta kapitel samlas en del praktiska tips i hanteringen av utrustningen kring micro:bit. Tips som kommit in och kan spara mycket tid och minska störningarna i klassrummet. Har du några fler – hör gärna av dig i så fall, vi fyller på!

#### Före lektion

- Första gången en **ny** micro:bit kopplas till dator via USB ska det göras kalibrering. Kan vara bra att ha det gjort innan lektion med elever då det kan ta låg tid för alla elever att genomföra.
- Ladda gärna ner ett program till varje micro:bit innan första lektion. Då ser eleverna att de kopplat in batterier rätt.

#### **Under lektion**

 Ta med gummisnoddar som eleven kan linda runt micro:bit och batterihållaren när övningar ska göras där eleverna ska röra sig i rummet med micro:bit. Detta för att undvika att kablar går av, batterier trillar ur hållare osv.



•

#### Efter lektion

• När en elev satt in batterihållaren i micro:bit, låt dem inte dra ut den igen. Kablarna går väldigt lätt av. Då är det bättre att ta för vana att pocka ur batterierna ut batterihållaren och låta batterihållaren vara inkopplad när micro:bit läggs tillbaka i sin antistatiska påse.

• ...

#### Micro:bit och surfplattor med trådlös överföring av program

- Ta med en dator och USB-sladd till micro:bit vid lektion. Om micro:bits programvara för Bluetooth inte är uppdaterad kommer en fil att behöva föras över till micro:bit. Detta kan bara göras via dator.
  - Den fil som ska laddas ner på micro:bit för ny programvara finns på: <u>https://microbit.org/guide/ble-ios/</u>
  - Se även länken på Youtube för mer information om hur det går till <u>https://www.youtube.com/watch?time\_continue=65&v=k6OQ5XzIroY</u>.
- Om eleverna använder surfplattor. Märk micro:biten så att samma elev kan arbeta med samma micro:bit nästa gång. Detta spar tid gällande parkoppling.
- Om program laddas ner till micro:bit med USB kabel kommer den tidigare parkopplingarna till surfplattor att "glömmas" av micro:biten och elever behöver ta bort gamla kopplingar och parkoppla igen.

#### Support – vad finns på nätet?

- Massor men ett par länkar att börja med för att komma vidare kan vara
  - https://support.microbit.org/support/home
  - https://microbit.org/guide/features/



#### Uppdrag 6

Här kommer lite mer kunskap kring uppdrag 6 och den hård- och mjukvara som används i uppdraget.

I uppdrag 6 används en accelerometer, en rörelsesensor som används i en mängd olika applikationer. Accelerometern sitter på micro:bits baksida, se bilden bredvid.

Den kanske mest kända tillämpningen är som krocksensor i bilar, där den registrerar en krocksituation och med tillhörande analys av detta indata kan utlösa krockkudden som skyddar förare eller passagerare. Det hela måste gå snabbt, analysen att ta beslut på om krockkudden ska lösas ut måste gå på ca 100 millisekunder.



Numera finns accelerometern även i andra

sammanhang, som exempelvis i mobiltelefoner. Den används för att skärmbilden ska visas i rätt ledd beroende på hur man håller sin mobiltelefon eller "Padda". Litenheten hos komponenten har gjort det möjligt att den kan komma till nytta i en mängd sammanhang! Diskutera gärna med eleverna i vilka andra sammanhang tekniken skulle kunna användas.

#### Mjukvara

Vid programmering med micro:bit finns det stöd för att hämta data från accelerometern. Nedan följer kompletterande fakta med exempel kring programmeringen av/med accelerometer.

- Funktionen acceleration (mg) x (eller y eller z) returnerar ett tal, ett värde mellan 0 – 1023 beroende på accelerationens storlek. I programmet får variabeln Acceleration detta värde.
  - Accelerometern på micro:bit är tre-axlig, dvs känner accelerationer och retardationer i tre riktningar, x – y – z.
  - När micro:bit-kortet ligger på
    bänken plant med displayen
    uppåt är x = 0, y = 0 och z = 1023.



 Händelsefunktionen när skaka kan ställas in för känslighet av olika typer av skakningar, som exempelvis 3g i vårt uppdrag. När micro:biten utsätts för en tillräckligt stor skakning aktiveras när skaka (3g osv), vilket kan användas i programmet för att mäta skakningen eller att i fallet med stegräknaren registrera ett steg.

Som ett komplement till uppdraget 6, gör programmet i bilden och testa de båda funktionerna. Funktionen **när skaka** är att betrakta som sann eller falsk, dvs den aktiveras bara om skakningen är tillräckligt stor. Vilken minsta Acceleration aktiverar funktionen **när skaka**? I programmet finns två variabler, dels **Acceleration** som är analog till sin karaktär men blir till ett digitalt värde av en separat enhet i micro:bits processor, dels **Steg** som räknas upp med antalet steg.



#### Sensor i uppdrag 6

#### Kort om accelerometern:

Accelerometern är en mikromekanisk komponent som i kombination med elektronik blir till en intelligent sensor. Den kallas för en MEMSkomponent. MEMS är en förkortning av **M**ikro **E**lektro **M**ekaniskt **S**ystem. Dimensionerna hos systemet är som sagt i mikrometerstorlek.

Se bild: Vid en krock kommer massan nr 1 att svänga mellan kontakterna med nr 6 i bilden. Utrymmet 5 kommer att variera i storlek.

Nr 6, nr 1 och nr 5 bildar en kondensator, där nr 6 är anslutningarna till plattorna i kondensatorn och nr 1



www.explaintilatstun.com

tillsammans med nr 5 bildar elektrolyten som finns mellan metallplattorna i kondensatorn. I och med att massan nr 1 rör sig vid en rörelse förändras kapacitansen hos kondensatorn och tillhörande elektronik kan avgöra om det är en krock!

#### Uppdrag 7

Den Smarta resultattavlan är ett uppdrag där vi blir mer bekant med knapparna A och B på Micro:biten, som deras funktion och programmering. Vid programmering finns det stöd för att använda data (=signal) från knapparna A och B. Att trycka på en knapp innebär en övergång från ett

till noll, från hög till låg etc., och betraktas som en händelse. För detta finns funktionen **när knapp A trycks** (eller B).

#### Mjukvara

Programmet bredvid är en stomme, en början, till det slutliga programmet i uppdraget. Händelsefunktionen **när knapp A trycks** aktiveras bara när knappen trycks in (=ned). Koden för denna del av programmet ska vara



kort, här ska händelsen hanteras som att exempelvis förändra värdet på variabeln Antal. Inga utskrifter i denna funktion (undantag kan finnas). Det är i loopen **för alltid** som koden finns för vad som ska utföras, som exempelvis att skriva ut information.

I uppdragets bildspel byggs programmets planering upp i form av en händelsekedja mm och som beskriver vad programmet ska utföra. Det är två parallella sekvenser, en för hantering av knapptryckningar, en för utskriftsdelen.

En annan nyhet i programmeringen är villkorssatsen **om <villkor=sant> då \_\_\_\_\_ annars \_\_\_\_.** Villkorssatsen är mycket användbar och här ställs man inför **om "sant" – gör det - annars - gör detta** uttryckt på annat sätt.

I det här uppdraget handlar det om att testa när ytterlighetsvärdena har nåtts, dvs när antal mål blir fler än 9 eller när man backar är antal mål inte kan bli färre än 0! Om det behövs, lägg ner lite tid på att eleverna förstår detta, det har alla glädje av i längden!



Smartare Elektroniksystem ELECTRONIC COMPONENTS & SYSTEMS

#### Forts uppdrag 7

I ett program finns det olika skrivsätt, olika metoder, att öka en variabel med ett värde. Som en fördjupning i programmering ses två alternativ till att öka en variabel med värdet ett (1).

- Med Knapp A
  "- ändra Hemma med 1", betyder att variabeln Hemma ökar sitt värde med 1.
- Med Knapp B
  "sätt Borta till Borta + 1", betyder att det nya värdet hos variabeln Borta blir det "gamla Borta" + 1.

Testa programmet, och gör kanske om något i programmet för Uppdrag 7 för att kontrollera om du/eleverna har förstått.



#### Uppdrag 8

Att mäta ljus med micro:bit sker med en ljussensor som består av 25 sensorelement (fototransistorer) som sitter i samma enhet som respektive lysdiod i matrisen. Se mer fakta på nästa sida. Med sensorelementens uppmätta värden bildas ett medelvärde, ett tal mellan 0 – 255. Känsligheten är inte så stor, vilket innebär att det krävs mycket ljus för att få ett högt mätvärde. Dessutom är sensorelementen utspridda på en förhållandevis stor yta, vilket gör att en del av ytan lätt kan skuggas och då förändras mätvärdet. En del att tänka på och att ta hänsyn till.



#### Mjukvara

Vid programmering av micro:biten finns det stöd för att få data från ljussensorn.

Funktionen **ljusnivå** returnerar ett värde mellan 0 – 255, vilket kan tilldelas en variabel, i detta fall **Ljus**. Se fakta om detta i bildspelet på sidorna 19 - 24.

En annat tillägg i programmeringen i uppdrag 8 är att villkorssatsen kan utvecklas ytterligare till att omfatta två gränser till tre områden, se bild. Lite svårare, men rita upp en axel med gränsvärdena på tavlan för att sedan diskutera logiken kring den bilden. Se bild 30 i bildspel som illustrerar detta och kör gärna simulatorn i **"slow motion"** för att förstå programmets funktion.





#### Forts uppdrag 8: Mer om hårdvaran lysdiod – ljussensor hos Micro:bit **Fråga:** Vad består egentligen varje lysdiod av i displayen hos Micro:bit?

**Svar:** Varje "lysdiod" är egentligen två komponenter, den har två funktioner, en lysdiod och en fototransistor. Se bilderna till höger nedan.

En diod som kan lysa och en fototransistor som kan mäta infallande ljus.

Det sitter alltså 25 sådana kombinationer i lysdiodmatrisen. I matrisen nedan är två inkopierade.



#### Programmering: Se exempel på program

#### Frågor:

- a. Var i programmet, i vilka block, används lysdioderna?
- b. Var i programmet, i vilka block, används fototransistorerna?

#### Svar:

- a. Blocken visa siffra 4 och visa siffra Ljus
  (Respektive block tänder lysdioder för att visa siffran 4 samt värdet som variabeln Ljus har)
- Blocket sätt Ljus till ljusnivå
  (Funktionen ljusnivå returnerar ett medelvärde från de 25 sensorelementen, dvs fototransistorena)

**Fakta**: Som ljussensor mäter micro:bit med ett antal av "lysdioderna" och gör en medelvärdesbildning av resultatet.

**Test:** Håll för ett antal "lysdioder" så minskar mätvärdet vid samma mätplats och ljusförhållande.











## Sammanfattning

#### Utvecklingsmiljön för Micro:bit

I temat Smartare Produkter arbetar eleverna med programvaruutveckling i en utvecklingsmiljö med blockprogrammering för en mikrodator. Utvecklingsmiljön består av några olika verktyg med tillhörande möjligheter:

- Editor den högra delen av fönstret där programmet har sin kod. I detta fall är koden i form av funktionsblock. Koden är den som programmeraren gör och ser och kallas ibland för källkod.
- Blockkategorier rubriker i mittendelen av fönstret där programmeraren väljer block för programmet som ska utvecklas.



- Simulator den vänstra delen av fönstret där koden kan testas (verifieras). Simulatorn har sina begränsningar, men fungerar för de flesta av de program som ska utvecklas under temat Smartare Produkter.
- Ladda ned knappen nere till vänster i fönstret, med vilken man startar processen för att få en exekverbar (=körbar) kod. Koden kompileras och en s k Hex-fil skapas. Hex-filen kan laddas ner ("droppas") till Micro:bits flashminne och är en kod som Micro:bits processor "förstår". Hex är förkortning av hexadecimal och är ett sätt att komprimera binär kod.

Se bilden nedan för process och terminologi – Från källkod till exekverbar kod





#### Systemet Micro:bit

I detta avsnitt sammanfattar vi systemet Micro:bit med några kommentarer kring de komponenter eller delsystem vi använt.

Den processor som används är en ARM Microcontroller och är plattformen i systemet. Micro:bit heter produkten, ett elektronikkort som är framtaget för utbildnings- och utvecklingsändamål. I bilden nedan finns de flesta delsystem och komponenter angivna, varav vi använder ett flertal av dessa:



- 25 LED lights, där även ljussensorer finns ihop med lysdioderna
- 2 buttons, Knapp A och Knapp B
- reset button, resetknapp för att starta om program
- *battery socket*, batterianslutning
- processor, där även temperatursensorn finns integrerad
- accelerometer
- ...

Notera att:

En Microcontroller är en processor som innehåller enheter för att hantera in- och utsignaler till extern elektronik med kompletterande stöd i mjukvarublock, som exempelvis för knappar, accelerometer och lysdioder.

Till skillnad från en processor som sitter i en Lap Top etc. där behovet är annorlunda, som att hantera internet, skärm, applikationer som olika Officeprogram, spelprogram osv. I en Lap Top finns också stora program som operativsystemet Windows eller iOS, vilka tar mycket datakraft i sig själva!





#### Mer fakta vid hantering av Micro:bit-kortet

Enheter på micro:bit, se bilden med enheter varav de flesta finns med i förra bilden på systemet.

Men ett par nyheter att notera:

- I Chipet (NXP KL26Z) finns det styrprogram, den *firmware*, som hanterar micro:bit-kortets hårdvaruenheter. Vid anslutning till dator för utveckling av uppdragen är det inget vi normalt ska kunna, mer än det som står omnämnt på sidan 10 i detta dokument. "- Att hamna i *Maintenance mode eller* underhållsmod" av misstag.
- Men när en iPad eller motsvarande används för utveckling av uppdragen kan det finnas behov av att byta ut



versionen av styrprogrammet i chipet NXP. Då gäller följande:

- i filen **Details.txt** finns information om vilken *Interface Version* som det micro:bitkort du använder har.
- om micro:bits programvara för Bluetooth inte är uppdaterad kommer en fil att behöva föras över till micro:bit. Det är styrprogrammet som behöver uppdateras på micro:biten. Den fil med ny programvara som ska laddas ner till micro:bit sker via en dator med USB-sladd och finns på: <u>https://microbit.org/guide/ble-ios/</u>. Detta sker med micro:bit i underhållsmod.

Se även länken på Youtube nedan för mer information om hur det går till: <u>https://www.youtube.com/watch?time\_continue=65&v=k6OQ5XzIroY</u>.

- Några fakta om trådlös kommunikation med micro:bit-kortet. I bilden ovan ses antennen för trådlös kommunikation:
  - Blue tooth Low Energy (BLE) är en teknologi som används av för att kommunicera med micro:bit via en Padda eller Smartphone. BLE är ett standardiserat protokoll som kräver lite energi och används i elektronik som micro:bit, vilka drivs med batterier.
  - Det finns också en radio på micro:bit. Den används i kommunikationen mellan olika micro:bits i applikationsprogrammen. Radion använder samma frekvensområde, dvs runt 2,4 GHz, men har ett annat och förmodligen "eget" protokoll. Jämför med exempelvis fjärrkontrollen för ett centrallås på bilar, där varje bilmärke har sitt eget "system", sitt eget protokoll. Det är en betydligt mindre datamängd som överförs i dessa sammanhang.





## Talsystem

I temat Smartare Produkter träffar vi på två exempel på hur in-enheter ger olika stora tal i micro:bit.

- Ljussensorn ger ett tal mellan 0 255
- Accelerometern ger ett tal mellan 0 --- 1023

Talen har sin grund i det binära talsystemet, ett talsystem som används av datorer för beräkningar. Det kan vara dags att veta mer om detta, speciellt inför nästa tema Den Smarta staden!

#### Decimala talsystemet

Några fakta att utgå från är att decimala talsystemet har Basen 10. Exempelvis:

• Talet  $736_{10}$  som betyder 7\*100 + 3\*10 + 6\*1, eller  $7*10^2 + 3*10^1 + 6*10^0$  om vi analyserar.

Tillåtna tecken i varje position är 0 – 9, med tre positioner kan talen 0 ---  $999_{10}$  skrivas och med fyra positioner 0 ---  $9999_{10}$ 

• Exempel med fyra siffror 8437<sub>10</sub>

Mest signifikant siffra är 8 (8\*10<sup>3</sup>) och Minst signifikant siffra är 7 (7\*10<sup>0</sup>)

#### Binära talsystemet

Nu är Basen 2. Exempelvis:

Talet 1101<sub>2</sub> som betyder 1\*8 + 1\*4 + 0\*2 + 1\*1 (=1310), eller 1\*2<sup>3</sup> + 1\*2<sup>2</sup> + 0\*2<sup>1</sup> + 1\*2<sup>0</sup> om vi analyserar.

Tillåtna tecken i varje position är 0 och 1, med tre positioner kan talen 000 --- 111  $(0 - 7_{10})$ , skrivas och med fyra positioner 0000 --- 1111  $(0 - 15_{10})$ .

Osv, talen ovan består av 3 respektive 4 bitar.

o Åtta bits (bitar) är en Byte

Med åtta bitar är talområdet 0 –  $255_{10}$ , exempelvis talet ( $\frac{1}{1}100 111 \frac{0}{2}$ )

Mest Signifikant Bit (MSB) är  $\frac{1}{2}$  (1\*2<sup>7</sup>=128) och Minst Signifikant Bit (LSB) är  $\frac{0}{2}$  (0\*2<sup>0</sup>)

• Hur gör jag om från decimalt till binärt?

En metod visas här!

De åtta positionerna i ett binärt tal är 128 ... 64 ... 32 ... 16 ... 8 ... 4 ... 2 ...1

Antag att talet 165 (decimalt) ska göras om till binärt tal.

- I talet 165 ryms en 128, 37 blir kvar alltså ingen 64 men 32 ryms. Då blir det 5 kvar, dvs ingen 16, ingen 8 men en 4, ingen 2 men en 1 ryms.
- Det betyder att det blir 1010 0101 binärt om man går igenom tallistan ovan.
  Med ett mellanrum mellan biten för16 och 8 blir det lite tydligare att läsa.

Fungerade den förklaringen?

Bra att kunna detta för vårt tema, och definitivt i nästa!