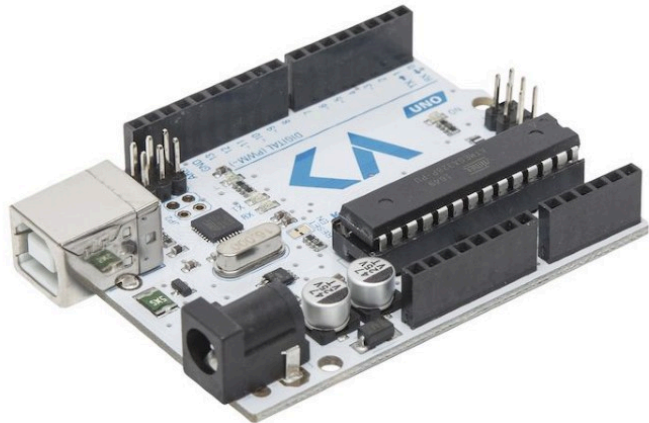


Arduino i kursen Teknik 1 - övningar

Inledning	2
Installation av IDE	2
Börja programmera	3
Anslut Uno	3
Variabler	3
Slumptal	3
if- och else-sats	3
OCH och ELLER	4
for-loop	4
Övningar	5
Digitala utgångar	6
Blinkande diod på 13	6
Övningar	6
Blinkande dioder på andra utgångar	7
Övningar	7
Digitala ingångar	8
Strömbrytare och lysdiod	8
Övningar	9
Lösningar	9
Dimra med PWM-styrning	10
Övningar	10
Temperatursensor	11
Övningar	11
Diodringen Neopixel	13
Övningar	13
Ohms lag...	14
Övningar:	14
Elektrisk effekt	15
Facit	17

Inledning

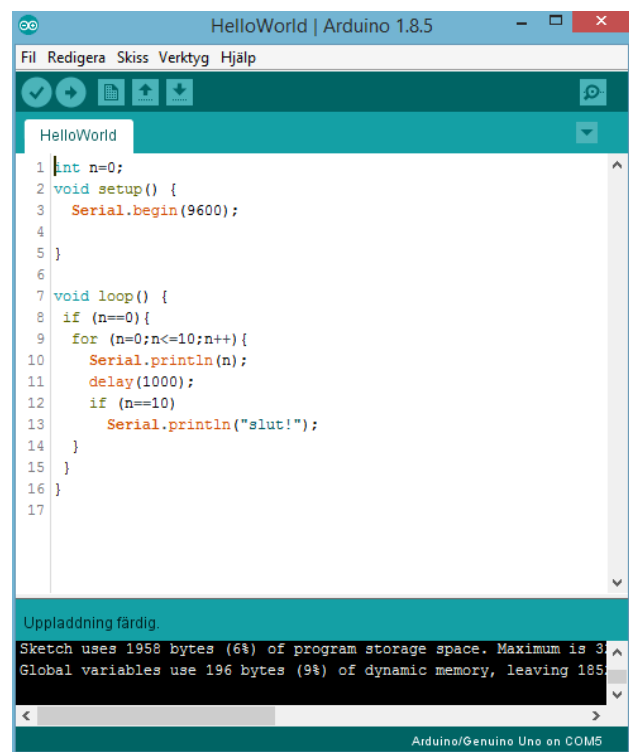
Tills för några år sedan byggdes elektronikkonstruktioner med transistorer, kondensatorer och resistorer. Vill man dessutom bygga något som skulle styra hur exempelvis lysdioder blinkar blev det genast väldigt komplicerat. För att inte tala om svårigheter om elektroniken dessutom skulle interagera med en dator för att exempelvis samla i mätvärden för hur temperaturen avtar i en termos under några timmar. Nu finns emellertid Arduino som både underlättar elektronikbyggandet och gör det relativt enkelt styra det som byggs via datorn. Arduino har stora möjligheter (kolla [här!](#)) och i denna minikurs ska vi lära oss att programmera i språket APL (Arduino Program Language) i utvecklingsmiljön Arduino IDE för att kunna styra elektroniken vi skapar (blinkande lysdioder och temperaturmätning) via utvecklingskortet Arduino Uno.



Installation av IDE

Vi börjar med att installera utvecklingsmiljön Arduino IDE där programmen skrivs som styr elektroniken vi senare bygger. Klicka på kjl.cm/arduino-103 och ladda ned och installera Windowsversionen. Klicka [här](#) om du vill ha hjälp med installationen. Välj porten COM5 under menyvalet **Verktyg/Port** i IDE:t. Nu är det bara att sätta igång. Här följer först några bra länkar:

- Kjell & Company:s [webbplats](#) om Arduino
- Sida med [projektideer](#) för Arduino
- Arduinos hemsida: <https://www.arduino.cc/>
- Kursen här använder startkitet [Kom igång med ARDUINO](#) för 800 kr från Kjell & Company.



Börja programmera

Anslut Uno

Anslut hårdvaran, utvecklingskortet Arduino Uno, till datorn med den blå USB-kabeln. Nu är vi redo att skriva vårt första program i APL, som skriver ut *Programmering är kul* på den seriella monitorn. I den första funktionen *setup()* anges hur många bitar som ska överföras varje sekund mellan datorn och kortet. Det kallas *bandbredden*. Här är alltså bandbredden 9600 b/s, medan mobilens bandbredd på 4G kanske är 100 Mb/s. I funktionen *loop()* upprepas ständigt raderna i funktionen. I det här fallet upprepas utskriften av texten *Programmering är kul* i den seriella monitorn (klicka på ikonen till höger!) med en fördröjning av 1000 ms. På följande länk finns många färdiga APL-funktioner:

<https://www.arduino.cc/reference/en/>

```
test1 $
1 void setup() {
2   Serial.begin(9600);
3 }
4
5 void loop() {
6   Serial.println("Programmering är kul");
7   delay(1000);
8 }
```

```
test1
1 int antal=0;
2 void setup() {
3   Serial.begin(9600);
4 }
5
6 void loop() {
7   antal=antal+1;
8   Serial.println(antal);
9   delay(1000);
10 }
```

Variabler

I programmet ovan skrivs ju samma sak ut hela tiden. Med en *variabel* kan utskriften ändras. Variabeln *antal* är 0 (noll) när programmet startar och ökar sedan med 1 för varje varv i loopen. Variabler måste alltid *deklarerars*. Ordet *int* anger att variabeln *antal* är ett heltal (integer) i programmet till höger.

```
test1 $
1 int antal=0;
2 void setup() {
3   Serial.begin(9600);
4 }
5 void loop() {
6   antal=antal+1;
7   Serial.println(antal);
8   if (antal==10)
9     Serial.println("Nu är det tio!");
10  delay(1000);
11 }
```

Slumptal

Vill vi skriva ut texten "Nu är slumptalet ..." innan slumptalet på samma rad kan vi lösa detta med en print-sats som inte gör radbrytning, enligt programmet till höger. Med funktionen *random(1, 7)* genereras slumptalen 1-6 och med *random(10)* genereras slumptalet 0-9. Funktionen *randomSeed()* skapar en ny följd av slumptal.

```
slump
1 void setup() {
2   Serial.begin(9600);
3   randomSeed(analogRead(0));
4 }
5 void loop() {
6   Serial.print("Nu är slumptalet: ");
7   Serial.println(random(1,7));
8   delay(1000);
9 }
```

if- och else-sats

I följande program skrivs texten "Nu är det tio!" ut när antal är 10. Detta kallas med en if-sats, enligt exemplet till höger.

OCH och ELLER

En variant är att generera slumpstal och kolla i vilket intervall talen ligger. Med funktionen `random(1, 7)` genereras slumpstalen 1-6 (Kolla!). 1-3 ska det skrivas ut "Ett till tre"

Om slumptalet är och är slumptalet 4-6 ska det skrivas ut "Fyra till sex". Koden längst till höger fixar detta med OCH-operatorm `&&`. En annan variant är att slumpstalen 2 eller 4 ger vinst. Detta löses med ELLER-operatorm `||`, enligt exemplet ovan till vänster.

```
1 int tal;
2 void setup() {
3   Serial.begin(9600);
4   randomSeed(analogRead(0));
5 }
6 void loop() {
7   tal=random(1,7);
8   Serial.print(tal);
9   if (tal==2 || tal==4)
10    Serial.print(" Vinst!");
11   Serial.println();
12   delay(1000);
13 }
```

```
1 int tal;
2 void setup() {
3   Serial.begin(9600);
4   randomSeed(analogRead(0));
5 }
6 void loop() {
7   tal=random(1,7);
8   Serial.print(tal);
9   if (tal>=1 && tal<=3)
10    Serial.println(" Ett till tre");
11   if (tal>=4 && tal<=6)
12    Serial.println(" Fyra till sex");
13   delay(1000);
14 }
```

for-loop

Programmen ovan vi hittills gjort pågår ju i all oändlighet. Vill vi upprepa något ett visst antal gånger passar en for-loop. Programmet till höger skriver ut 10 slumpstal. Nästa program skriver ut summan av heltalen 1-10.

```
1 int sum;
2 void setup() {
3   Serial.begin(9600);
4   randomSeed(analogRead(0));
5 }
6 void loop() {
7   sum=0;
8   for (int i=1;i<=10;i++){
9     sum=sum+i;
10    Serial.println(sum);
11    delay(1000);
12 }
```

```
1 int i=1;
2 void setup() {
3   Serial.begin(9600);
4   randomSeed(analogRead(0));
5 }
6 void loop() {
7   if (i==1){
8     for (i=1;i<=10;i++){
9       Serial.println(random(1,7));
10      delay(1000);
11    }
12   }
13 }
```

```
1 int antal=0;
2 void setup() {
3   Serial.begin(9600);
4 }
5 void loop() {
6   antal=antal+1;
7   if (antal<=10)
8     Serial.println("Antal mindre än eller lika med än 10!");
9   else
10    Serial.println("Antal större än 10!");
11   delay(1000);
12 }
```

Övningar

Börja med att göra en mapp med namnet **arduino** i din teknikmapp på datorn. Spara sedan dina arduinoprogram nedan i den mappen.

- 1) Skriv ett program som skriver ut ditt namn på den seriella monitorn fem gånger i sekunden på varsin rad. Spara som **uppgift1**.
- 2) Modifiera föregående program så att ditt namn skrivs ut på samma rad med ett blankteckens mellanrum. Spara som **uppgift2**.
- 3) Skriv ett program som skriver ut uppräkningsen 1 2 3 4 5... på en ny rad varje sekund. Spara som **uppgift3**. Ledning: Använd heltalsvariabeln *antal*.
- 4) Modifiera föregående program så att det står *Numret är 1, Numret är 2, Numret är 3* o.s.v. på varje rad och så att det görs två utskrifter i sekunden. Spara som **uppgift4**.
- 5) Modifiera föregående program så att det skrivs ut *Nu börjas det om!* på en egen rad när antalet är 10 och då börjar uppräkningsen om från 1 igen. Spara som **uppgift5**.
- 6) Skriv ett program som beräknar summan av 1+2+3+4... med summan utskriven på en ny rad två gånger varje sekund. Ledning: Använd variablerna *antal* och *summa*. Spara som **uppgift6**.
- 7) Modifiera föregående program så att antalet står framför summan på varje rad, enligt urklippet till höger. Spara som **uppgift7**.
- 8) Modifiera föregående program så att istället produkten 1·2·3·4·.... (kallas faktoriell) skrivs ut på en egen rad, enligt urklippet till höger. Spara som **uppgift8**. Ganska snart blir produkten negativ. Varför då och vid vilket tal exakt? Lös problemet så att det räknar rätt! Kolla här: <https://www.arduino.cc/reference/en/> Hur länge räknar det rätt nu?
- 9) Skriv ett program som skriver ut hela slumpetal mellan 1-10 på var sin rad. Spara som **uppgift9**.
- 10) Modifiera föregående program och så att det efter varje slumpetal står *Talet ligger mellan 1-5* respektive *Talet ligger mellan 6-10*. Spara som **uppgift10**.
- 11) Använd en for-loop och beräkna summan av hundra slumpetal mellan 1-10 och skriv ut det på varje rad. Spara som **uppgift11**.
- 12) Modifiera föregående uppgift och beräkna medelvärdet av hundra slumpetal mellan 1-10 och skriv ut det på varje rad. Spara som **uppgift12**. Vad bör medelvärdet bli?

1	1	
2	3	
3	6	
4	10	
5	15	

1	1
2	2
3	6
4	24
5	120
6	720

Digitala utgångar

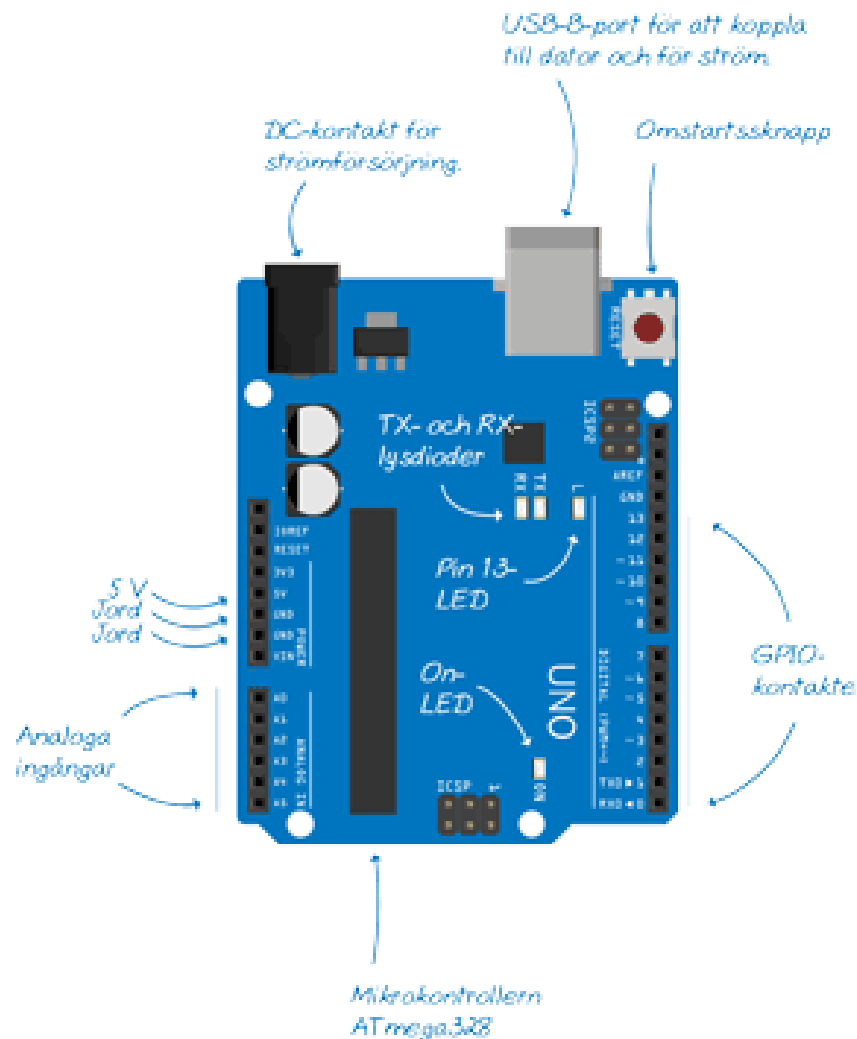
Blinkande diod på 13

Det är nu dags att börja styra utgångarna på UNO-kortet. Den digitala utgången 13 på kortet är kopplad till den gula lysdioden. Följande kod öppnar, tändar och släcker dioden, som är tänd en sekund och släckt en sekund:

```
1 void setup() {  
2   pinMode(13, OUTPUT);  
3 }  
4 void loop() {  
5   digitalWrite(13, HIGH);  
6   delay(1000);  
7   digitalWrite(13, LOW);  
8   delay(1000);  
9 }
```

Övningar

- 13) Programmera exemplet ovan så att det fungerar. Spara uppgiften som **uppgift13**.
- 14) Morsekoden för SOS-nödssignal är tre korta, tre långa och tre korta, vilken låter så [här](#). Observera att efter de tre sista korta är det en längre paus innan signalen upprepas. Skriv ett program så att de korta signalerna lyser i 100 ms, de långa signalerna i 1000 ms och lysdioden är släckt i 500 ms mellan varje signal. Efter den sista korta är det släckt i 2000 ms. Sedan upprepas det hela igen. Lös förslagsvis uppgiften med tre for-loopar. Spara uppgiften som **uppgift14**.
- 15) Skapa ett program där dioden blinkar snabbare och snabbare. Första blinkningen lyser den en sekund och är släckt en sekund. Andra blinkningen är den tänd respektive släckt 90 % av föregående tid, d.v.s 900 ms. Nästa varv i loopen är den tänd och släckt 90 % av den tiden o s v. Varför slutar dioden att blinka efter ett tag? Spara uppgiften som **uppgift15**.
- 16) Lös nu föregående uppgift omvänt så att dioden först blinkar 1 ms och sedan ökar lystiden med 10 % för varje varv i loopen. Tiden blir nu ett decimaltal och därför måste datatypen *int* ersättas med *float*. Spara uppgiften som **uppgift16**.



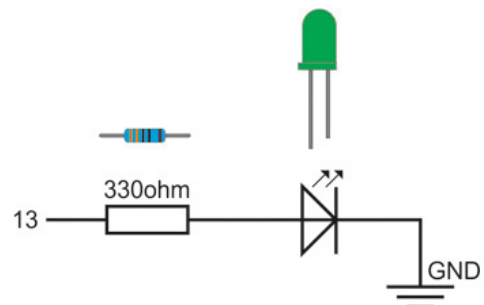
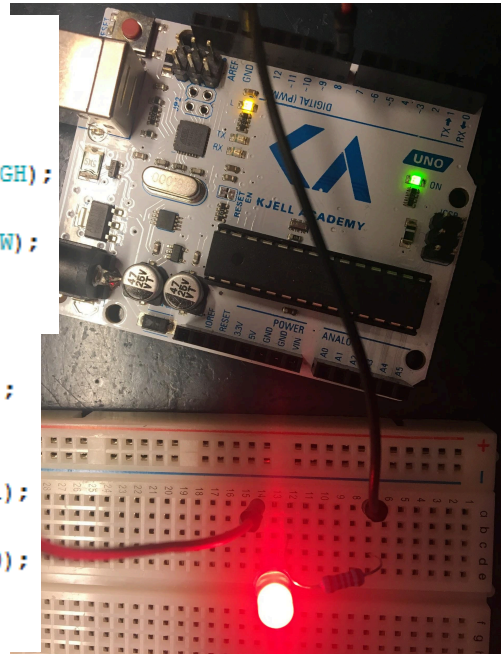
- 17) Nu ska uppgift 16 och 15 kombineras så att dioden först börjar blinka som i uppgift 16. När blinktiden är 1 sekund kortas blinktiden med 10 % som i uppgift 15. När blinktiden åter är 1 ms förlängs blinktiden åter med 10 % o s v. Spara uppgiften som **uppgift17**. Läraren kan visa hur det ska fungera.

Blinkande dioder på andra utgångar

Hittills har endast den fasta lysdioden på utgång 13 blinkat. Vi vill dock även få röda och gröna dioder att blinka på andra utgångar. Detta görs på i princip samma sätt som tidigare, men vi måste dessutom ansluta dioden till en [kopplingsplatta](#) i serie med ett motstånd (resistor) på omkring 100 ohm. Motståndet begränsar strömmen så att inte dioden förstörs. Bilden visar uppkopplingen på kopplingsplattan. Den svarta sladden är ansluten till jord (GND eller -) och den röda sladden till digitala utgången 8. Observera att diodens långa ben ska anslutas mot utgång 8 (+) eftersom dioden bara leder ström i en riktning. Observera att HIGH och LOW kan ersättas med 1 och 0, enligt det undre programexemplet ovan.

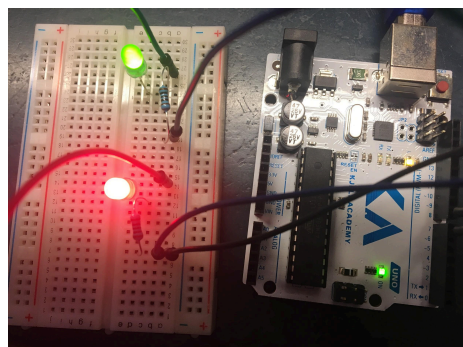
```
1 void setup() {
2   pinMode(8, OUTPUT);
3 }
4 void loop() {
5   digitalWrite(8, HIGH);
6   delay(1000);
7   digitalWrite(8, LOW);
8   delay(1000);
9 }

1 void setup() {
2   pinMode(8, OUTPUT);
3 }
4 void loop() {
5   digitalWrite(8, 1);
6   delay(100);
7   digitalWrite(8, 0);
8   delay(1000);
9 }
```



Övningar

- 18) Programmera och koppla upp exemplet ovan så att dioden blinkar. Undersök vid vilken längsta tid (delay) ögat inte kan uppfatta blinkningen. Prova att vända dioden - vad händer? Spara uppgiften som **uppgift18**.
- 19) Modifiera föregående program så att dioden lyser 1000 ms var femte gång, annars lyser den 100 ms. Mellan blinkningarna är dioden alltid släckt 1000 ms. Spara uppgiften som **uppgift19**.
- 20) Koppla nu även in en grön diod till utgång 7 och modifiera programmet så att den gröna dioden lyser 1000 ms tillsammans med den röda dioden endast var femte blinkning. Spara uppgiften som **uppgift20**.



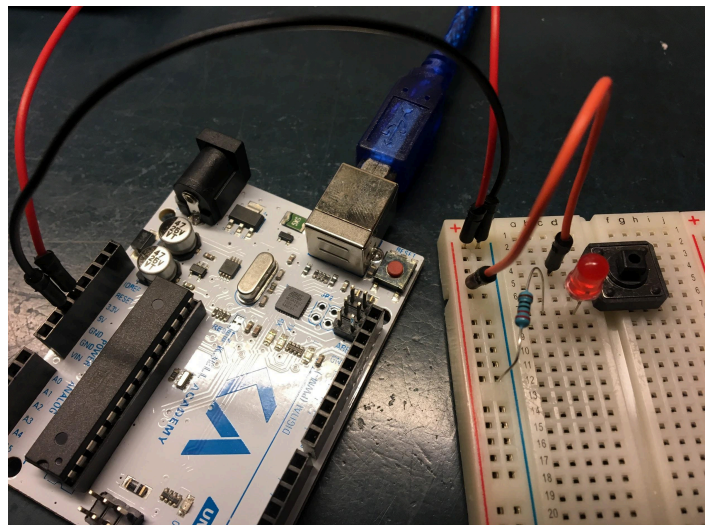
- 21) Justera sladdarna på kopplingsplattan så att den gröna dioden lyser samtidigt som den röda i uppgift 20.
- 22) Justera koden i uppgift 18 så att en den röda och gröna dioden blinkar varannan gång. Använd förslagsvis de digitala utgångarna 7 och 8. Spara uppgiften som **uppgift22**.
- 23) Koppla även in en gul lysdiod till förslagsvis port 6 så att en diod lyser i taget i ordningen gul, grön, röd, gul grön röd... Spara uppgiften som **uppgift23**. Lös uppgiften med endast en if-sats.
- 24) Programmera en trafiksignal med följande lystider: Grön: 5 s, gul: 1 s, röd: 3 s, röd+gul: 1 s, grön: 5 s o.s.v. Spara uppgiften som **uppgift24**.
- 25) Skriv ett program som slumpar vilken av tre lysdioder som ska lysa, samt hur länge den ska lysa (10-1000 ms). Tiden den är släckt är 1000 ms. Spara uppgiften som **uppgift25**.
- 26) Skriv ett program som simulerar en tärning genom att slumpa ett tal mellan 1 och 6 och som tänder lika många lysdioder, som lyser i två sekunder. Innan ett nytt slumpantal genereras är lysdioderna släckta i 0,5 s. För att kontrollera att uppkopplingen stämmer ska slumptalet även skrivas ut i den seriella monitorn. Spara uppgiften som **uppgift26**.



Digitala ingångar

Strömbrytare och lysdiod

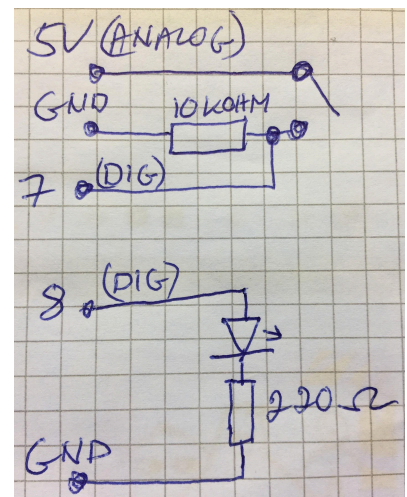
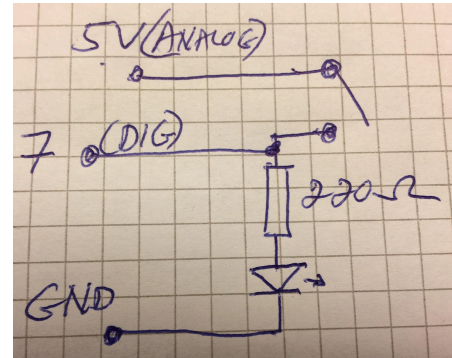
Vi börjar med att prova en strömbrytare som tänder en lysdiod när den är nedtryckt. Som spänningskälla använder vi nu den analoga utgångarna för 5 V (+) och GND (-). För att strömmen genom lysdioden ska bli mindre än tidigare använder vi ett motstånd på 220 ohm. Vi vill även registrera att dioden tänds via en signal till en digital ingång. Vi kopplar därför en sladd mellan lysdiodens plusben och den digitala porten 7 (exempelvis). När knappen är nedtryckt är det 5 V på diodens plusben och då blir insignalen hög (1) på den digitala ingången. Koden som registrerar detta och skriver ut 1 eller 0 i den seriella monitorn varje sekund beroende om knappen är nedtryckt eller inte visas till höger.



```
1 void setup() {
2   pinMode(7, INPUT);
3   Serial.begin(9600);
4 }
5 void loop() {
6   Serial.print(digitalRead(7));
7   delay(1000);
8 }
```


Övningar

- 27) Koppla upp lysdiod, motstånd och brytare och anslut till 5 V på den analoga spänningskällan så att dioden tänds när knappen trycks ned.
- 28) Modifiera föregående övning så att en etta eller nolla skrivs ut i den seriella konsolen varje sekund beroende på om knappen är nedtryckt eller inte. Om knappen är nedtryckt är det en etta (5 V) på ingång 7, enligt kopplingsschemat till höger. Använd koden i exemplet ovan. Vid korta tändningar av dioden registreras inte ettan. Varför inte? Hur kan man göra om även korta knapptryckningar ska registreras? Testa! Spara uppgiften som **uppgift28**.
- 29) Skriv ett program som tänder en lysdiod när strömbrytaren trycks ned och som släcker den efter två sekunder. Använd förslagsvis den digitala ingången 7 för att kolla om knappen är nedtryckt och den digitala utgången 8 som spänningskälla för lysdioden, enligt kopplingsschemat till höger. Som strömkälla för att kolla om knappen är nedtryckt (5 V på ingång 7) används den analoga utgången 5 V, som kopplas i serie med 10 kohm för att undvika höga strömmar. Spara uppgiften som **uppgift29**.
- 30) Modifiera föregående uppgift så att lysdioden tänds när knappen trycks ned utan att släckas efter två sekunder. Spara uppgiften som **uppgift30**.
- 31) Modifiera föregående uppgift så att lysdioden tänds när knappen trycks ned första gången och släcks när knappen trycks ned andra gången o.s.v. Vad händer om knappen trycks ned långsamt? Hur kan detta lösas? Spara uppgiften som **uppgift31**.
- 32) Skriv ett program som tänder dioden när knappen tryckts ned fem gånger. Spara uppgiften som **uppgift32**.
- 33) Skriv ett program som som tänder dioden med en knapp och släcker den med en annan. Spara uppgiften som **uppgift33**.
- 34) Skriv ett program som som tänder dioden när knappen varit intryckt i tio sekunder. Spara uppgiften som **uppgift34**.
- 35) Modifiera föregående uppgift så att dioden släcks om knappen trycks ned ytterligare en gång. Därefter tänds den igen om knappen är intryckt i tio sekunder. Spara uppgiften som **uppgift35**.

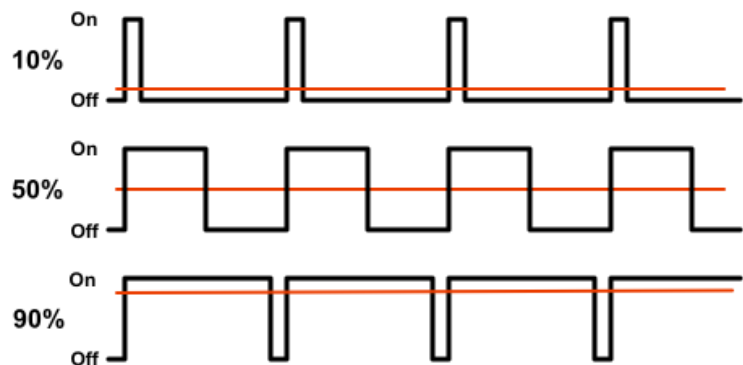


Lösningar

På denna [länk](#) finns förslag till lösningar på ovanstående uppgifter. Ladda ned den komprimerade filen och packa upp den med programmet 7-zip (finns på datorn).

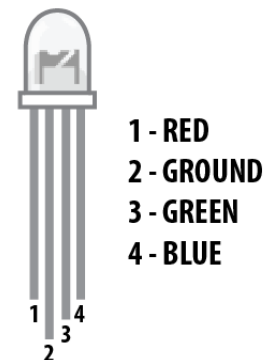
Dimra med PWM-styrning

En lysdiod är i princip på eller av, det finns inget mellanläge på belysningen. En variant är dock att pulsa dioden så att den blinkar med olika frekvens och att den är på och av i olika långa tider, vilket kan uppfattas som att dioden lyser med olika intensitet. Detta kallas *Puls Width Modulering* och går att åstadkomma med arduinokortets digitala utgångar 3, 5, 6, 9, 10 och 11. Med satsen `analogWrite(6, tal)` ges en intensiteten `tal` på den digitala utgången 6, där `tal` har ett värde mellan 0 och 255.



Övningar

- 36) Skriv ett program och gör en uppkoppling som slumpar en intensitet för lysdioden mellan 0 och 255 som lyser i 1 sekund. Använd en grön, gul och röd lysdiod samtidigt. Spara uppgiften som **uppgift36**.
- 37) Gör ett program som dimrar lysdioderna fram och tillbaka från 1-255. Tiden mellan varje ändring kan vara 100 ms. Spara uppgiften som **uppgift37**.
- 38) Prova rött, grönt och blått ljus på RGB-lysdioden. Det långa benet ansluts till den negativa sidan (GND) och de tre övriga benen ska anslutas i serie med var sitt motstånd på 220 ohm. Spara uppgiften som **uppgift38**.
- 39) Slumpa en RGB-färg med `analogWrite(6, tal)` som lyser i 500 ms. Spara uppgiften som **uppgift39**.



Temperatursensor

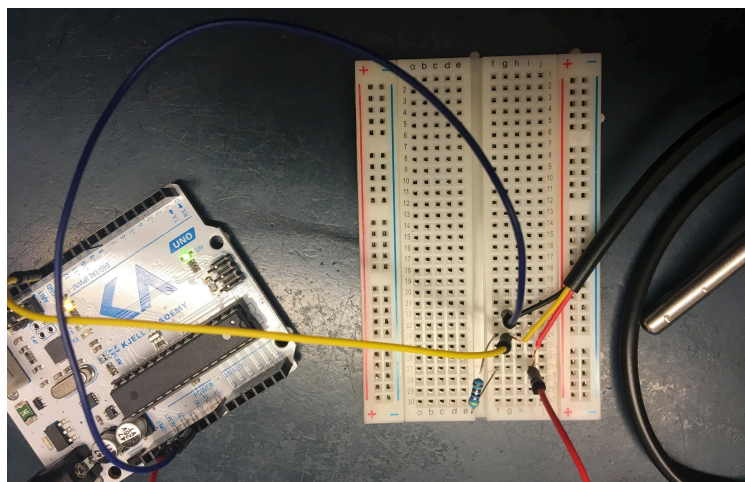
Till arduinokortet går det att ansluta olika typer av [sensorer](#) och motorer. Vi ska ansluta en sensor för temperaturmätning. För att detta ska fungera måste vi även ladda ned och installera två bibliotek, [Arduino Control Temperature Library](#) (DallasTemperature) och [OneWire](#). Packa upp och lägg dem i Arduinos mapp för bibliotek på c:/, C:\Program Files

(x86)\Arduino\libraries. De inkluderas i programmet via

menyvalet **Skiss/Inkludera bibliotek**. Följande film visar hur det går till:

<https://www.youtube.com/watch?v=qxEclOy6jpl>. Temperatursensorn har tre sladdar; den

röda ansluts till 5 V, den svarta till jord och den gula till en digital ingång. I exemplet nedan använde vi ingång 12. Mellan 5 V och den digitala ingången ska det sitta ett motstånd med resistansen 4,7 kohm. Programmet *temperatur* till höger skriver ut temperaturen i den seriella monitorn. Programmet finns även att ladda ned [här](#). Blir det mycket data i Serial Monitor är det bättre att lagra datan på en textfil. Detta görs med programmet [Cool Term](#). Kolla [här](#)! Om Serial Monitor stängs skrivs data från comporten (com3) istället till Cool Term, som kan sparas som en textfil via menyvalet **Connections/File Capture**.

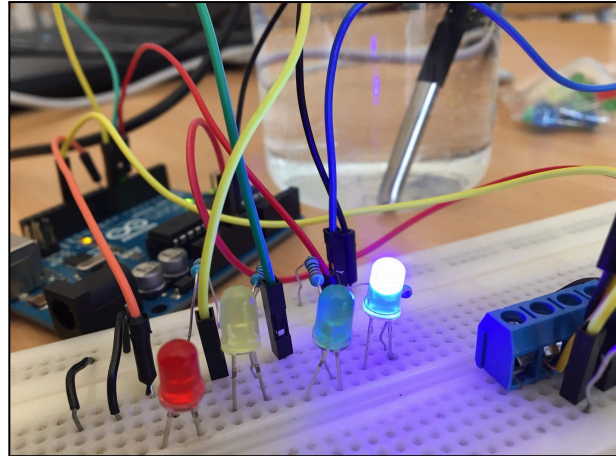


```
temperatur
1 // Include the libraries we need
2 #include <OneWire.h>
3 #include <DallasTemperature.h>
4 #define ONE_WIRE_BUS 12 //digitala porten 12
5 // Setup a oneWire instance to communicate with any OneWire devices
6 OneWire oneWire(ONE_WIRE_BUS);
7 // Pass our oneWire reference to Dallas Temperature.
8 DallasTemperature sensors(&oneWire);
9 void setup()
10 {
11     Serial.begin(9600); // start serial port
12     Serial.println("Dallas Temperature IC Control Library Demo");
13     sensors.begin(); // Start up the library
14 }
15 void loop()
16 {
17     sensors.requestTemperatures(); // Send the command to get temperatures
18     Serial.print("T=");
19     Serial.print(sensors.getTempCByIndex(0));
20     Serial.println(" grader");
21     delay(1000);
22 }
```

Övningar

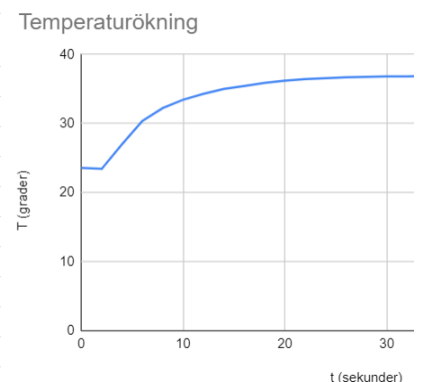
- 40) Installera de två biblioteken ovan och ladda sedan ned programmet ovan och provkör det. Spara uppgiften som **uppgift40**.
- 41) Modifiera föregående program så att endast temperaturen (utan text och enhet) skrivs ut var tionde sekund. Spara uppgiften som **uppgift41**.
- 42) Koppla in en diod så att den lyser om temperaturen är större än 40 grader, annars är den släckt.

- 43) Modifiera föregående uppgift så att en blå diod lyser om temperaturen $T < 25$ grader, en grön om $25 \leq T < 35$ grader, en gul om $35 \leq T < 45$ och en röd om $T \geq 45$ grader. Använd gärna andra temperaturintervall om de passar bättre.



- 44) Värm och håll upp ungefär 1 dl kokhett vatten i en liten bägare så att temperatursensorns metall del ligger under vattenytan. Registrera temperaturen i den seriella monitorn var 30:e sekund. Avbryt mätningen efter 20 min. Kopiera temperaturvärdena från den seriella monitorn och klistra in dem i Notepad++ och ersätt temperaturernas decimalpunkter (hittade ingen annan metod!) med decimalkomma och kopiera och klistra in dem i Google Kalkyl och infoga ett linjediagram som visar vattnets avsvälningskurva.

t (sekunder)	T (grader)
0	23,56
2	23,44
4	27
6	30,37
8	32,25
10	33,44
12	34,31
14	35
16	35,44
18	35,88



- 45) Koppla in en svart summer mellan mellan GND och exempelvis utgång 8 och kör den med koden i rutan till höger. Observera att det långa benet ansluts till pluspolen, d.v.s. mot utgång 8. Tonhöjden blir lägre om ett motstånd på några hundra ohm kopplas i serie med summern. Gör det!



```
void setup() {
  Serial.begin(9600);
  pinMode(8, OUTPUT);
  pinMode(7, OUTPUT);
}

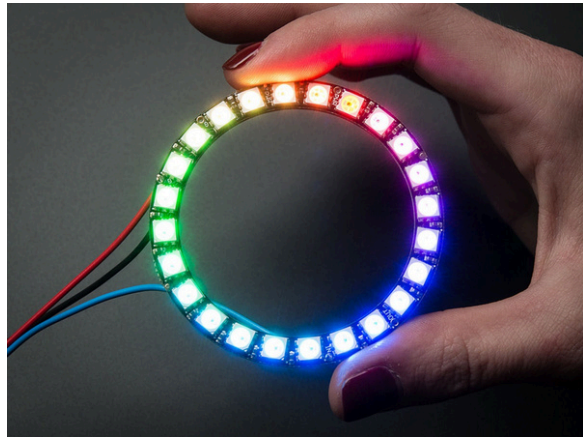
void loop() {
  digitalWrite(8, 1);
  delay(1000);
  digitalWrite(8, 0);
  delay(1000);
}
```

- 46) Skriv ett program som startar med tonlängden 1000 ms, enligt exemplet ovan, och som sedan kortar ned tonlängden med faktorn 0,9 (förändringsfaktorn) tills tonen låter 1 ms. Då ökar tonlängden med faktorn 1,1 tills tonen blir 1000 ms lång o s v.
- 47) Skriv ett program som startar summern när temperaturen $T > 45$ grader (eller den största temperaturen) i uppgift 43 ovan, men delay 0,1 s på ton och paus.

Diodringen Neopixel

Detta är en ring med 24 RGB-dioder, som kan tändas individuellt med ljusstyrka och färg via en digitalingång, dit datainput ansluts i serie med ett motstånd på 100 ohm (funkar även med 300 ohm). Ladda ned och packa upp biblioteket

Adafruit_NeoPixel-master till Arduinos biblioteksmapp på hårddisken och inkludera biblioteket till ditt program. Programmet till höger, som kan laddas ned [här](#), visar hur hur diod nummer 6 tänds och släcks varje sekund med grön färg. Ladda ned och provkör programmet.



Övningar

Modifiera programmet så att en slumpmässig diod lyser med en slumpmässig färg, så att alla dioder lyser med samma färg eller slumpmässiga färger, så att dioderna tänds i ordning varje sekund som en sekundvisare.

```
neopixel $
#include <Adafruit_NeoPixel.h>
const byte neoPin=10; //Ansluten till digitalingång 10
const byte neoPixels=24; //24 dioder
byte neoBright=10; //Ljusstyrkan 0-255, men använd max 100

Adafruit_NeoPixel ring=Adafruit_NeoPixel(neoPixels,neoPin,NEO_GRB);
//Skapar objektet ring
void setup() {
  ring.begin();
  ring.setBrightness(neoBright);
  ring.show();
}

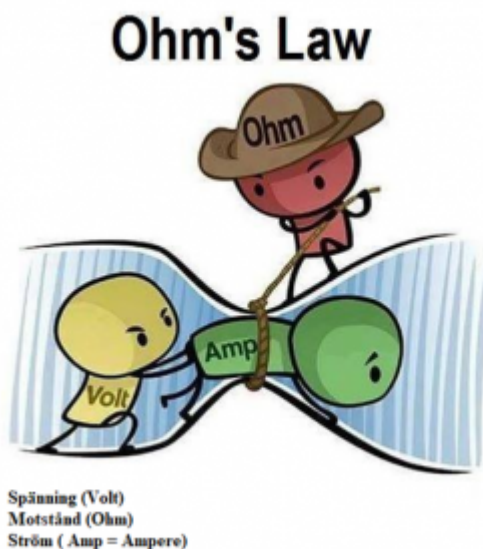
void loop() {
  ring.setPixelColor(6, ring.Color(0,250,0));
  ring.show();
  delay(1000);
  ring.setPixelColor(6, ring.Color(0,0,0));
  ring.show();
  delay(1000);
}
```


Ohms lag...

...anger sambandet mellan ström (I)
spänning (U) och resistans (R):

$$U = R \cdot I \quad I = \frac{U}{R} \quad R = \frac{U}{I}$$

Storheten spänning mäts enheten Volt [V],
ström i Ampere [A] och resistans i Ohm [Ω].
Ström är ett mått på hur många elektroner
(vilken laddning) som passerar en tvärsnitt
av en ledning per sekund. Spänningen är
lite svårare att på grepp om, men
uppkommer på grund av laddningsskillnad. I
analogi med en vattenledning kan strömmen
ses som hur mycket vatten som passerar
ledningen per sekund, spänningen är trycket
i ledningen och resistansen är motståndet
vattnet upplever i ledningen. Motståndet är
alltså större i en tunnare än i en grovre
ledning. Läs mer om Ohms lag på http://sv.wikipedia.org/wiki/Ohms_lag. Här följer ett par
exempel.



Exempel 1:

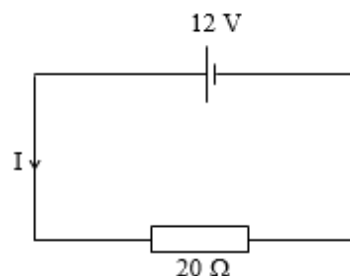
Bestäm strömmen I i kopplingsschemat till höger.

Lösning:

$$U=12 \text{ V}$$

$$R=20 \text{ } \Omega$$

$$\text{Strömmen } I = \frac{U}{R} = \frac{12}{20} = 0,6 \text{ A}$$



Exempel 2:

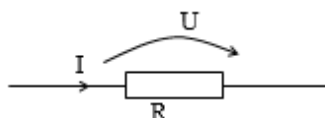
Strömmen genom ett motstånd med resistansen
 $R=220 \text{ } \Omega$ är 25 mA. Bestäm spänningen över
motståndet.

Lösning:

$$I=25 \text{ mA}=0,025 \text{ A}$$

$$R=220 \text{ } \Omega$$

$$\text{Spänningen } U = RI = 220 \cdot 0,025 = 5,5 \text{ V}$$



Övningar:

- 48) Bestäm strömmen genom ett motstånd med resistansen 22 ohm, om spänningen över motståndet är 1,5 V.
- 49) Strömmen genom ett motstånd med resistansen 5 ohm är 2,1 A. Bestäm spänningen över motståndet.
- 50) Bestäm motståndets resistans, om spänningen över ett motstånd är 1,5 V och strömmen genom motståndet är 120 mA.
- 51) Ett elektriskt element som är anslutet till 230 V drar strömmen 6,5 A. Beräkna elementets resistans.

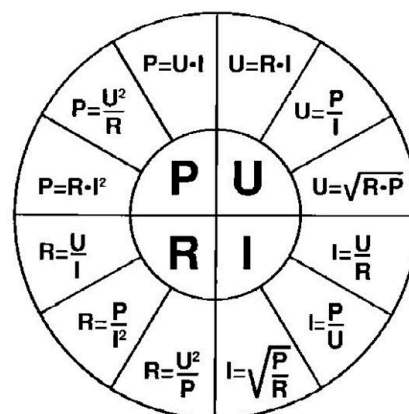
- 52) Ett annat element har resistansen 25 ohm. Vilken ström drar det när det anslut till 230 V?

Elektrisk effekt

En glödlampa och ett elektriskt element utvecklar ju värme när de ansluts till en spänning, vilket tyder på att de utvecklar effekt (P). Sambandet för effektutveckling i en elektrisk komponent är enkelt:

$$P = U \cdot I$$

Den elektriska effekten P mäts i enheten Watt [W]. Här är ett par exempel.



Exempel:

Vilken effekt utvecklas i en glödlampa som ansluts till spänningen 225 V, om strömmen genom lampan är 0,28 A?

Lösning:

$$U = 225 \text{ V}$$

$$I = 0,28 \text{ A}$$

$$\text{Effekten: } P = UI = 225 \cdot 0,28 = 63 \text{ W}$$

Exempel:

Vilken effekt utvecklas i en tråd med resistansen 40 ohm som ansluts till 225 V?

Lösning:

$$U = 225 \text{ V}$$

$$R = 40 \Omega$$

$$\text{Effekt: } P = \frac{U^2}{R} = \frac{225^2}{40} = 1266 \text{ W}$$

Övningar

- 53) När ett elektriskt element ansluts till spänningen 230 V drar det strömmen 6 A. Vilken effekt utvecklar elementet?
- 54) Vilken resistans har ett elektriskt element som utvecklar effekten 1000 W när det ansluts till nätspänningen 230 V?
- 55) Vilken effekt utvecklar ett elektriskt element (elradiator) som har resistansen 25 ohm när det ansluts till spänningen 230 V?
- 56) Vilken effekt utvecklar ett elektriskt element som drar strömmen 8,3 A och har resistansen 29 ohm?
- 57) Ett elektriskt element med resistansen 23 ohm utvecklar effekten 1400 W. Hur stor ström drar elementet?
- 58) Till vilken spänning ska elementet med resistansen 23 ohm anslutas för att det ska utveckla effekten 500 W?



59) Strömmen genom en lysdiod får vara högst 12 mA. Lysdioden ansluts till 5 V och tändes vid 2 V. Vilken resistans ska motståndet ha som kopplas i serie med lysdioden?

Facit

- 48) 68 mA
- 49) 10,5 V
- 50) 12,5 ohm
- 51) 35,4 ohm
- 52) 9,2 A
- 53) 1380 W
- 54) 52,9 ohm

- 55) 2116 W
- 56) 1998 W
- 57) 7,8 A
- 58) 107 V
- 59) 250 ohm