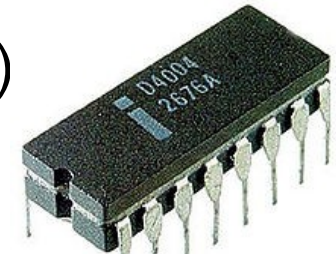


Begrepp

- **Mikroprocessor** - "Enchips-CPU"

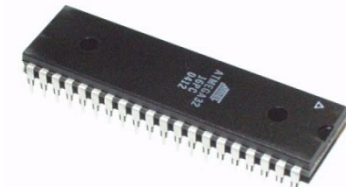
Dedicerade (specialiserade) varianter

- DSP (signalprocessor) – Ex: TMS5100 (1976)
- GPU (grafikprocessor) – Ex: Nvidia Geforce 256 (1999)
- CPU (centralprocessor) – Ex: Intel 4004 (1971)



- **Mikrokontroller**

Mikroprocessor med arbetsminne, programlagring och I/O inbyggd i samma chip eller på samma kort.



- **Inbyggda system**

Specialiserade datorsystem, ofta med realtidskrav, som utgör en integrerad del av någon maskin eller apparat.

Ingår i t.ex. bilar, mobiltelefoner, industrirobotar, biljettautomater, flygplan, mediaspelare, symaskiner och spelkonsoler.



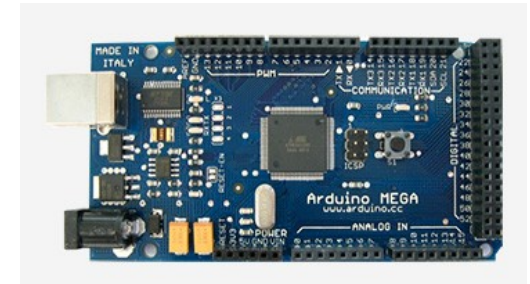
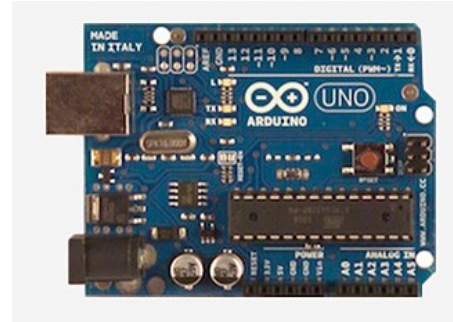
Arduino & Co

- Arduino UNO
- Raspberry PI
- Beaglebone Black
- PIC-mikrokontrollers (Microchip)
- MSP-EXP430G2 Launchpad (TI)
- Nanode
- Pinguino PIC32
- STM32 Discovery



Arduino-familjen

- Arduino UNO
- Arduino Mega 2560
- Arduino Leonardo
- Arduino Nano
- Arduino Duemilanove



Arduino-historia

- Ursprung: Studentprojekt i interaktionsdesign 2005 (Ivrea, norra Italien)
- Namnet Arduino härrör från en bar i Ivrea vars namn i sin tur kommer från *Arduin I* (955-1014), kung av Italien 1002-1014
- Grundarna:

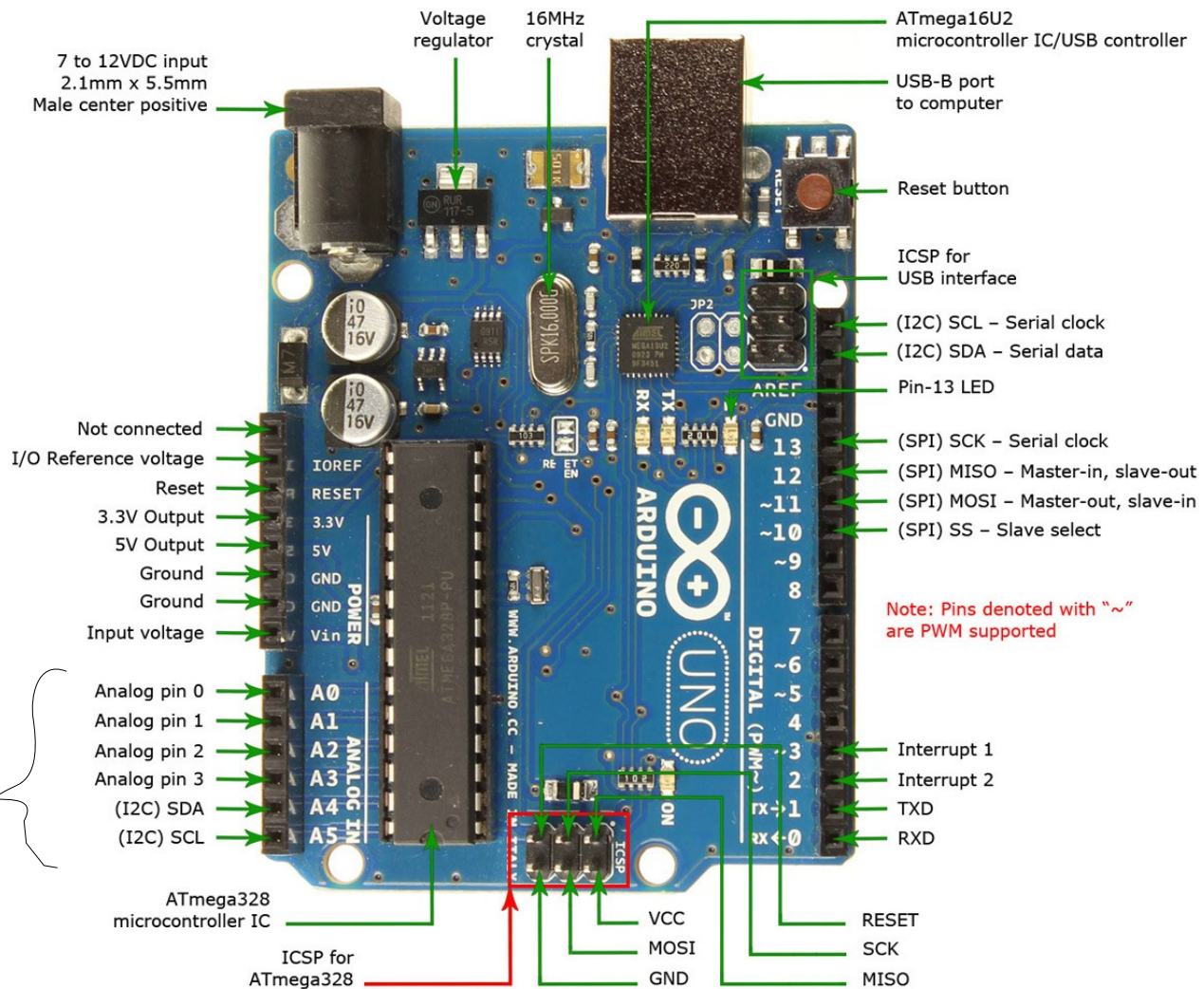
Gianluca Martino,
David Mellis,
David Cuartielles,
Tom Igoe,
och Massimo Banzi



Arduino-anatomi

<http://www.jameco.com/Jameco/workshop/circuitnotes/CN-arduino-uno.html>

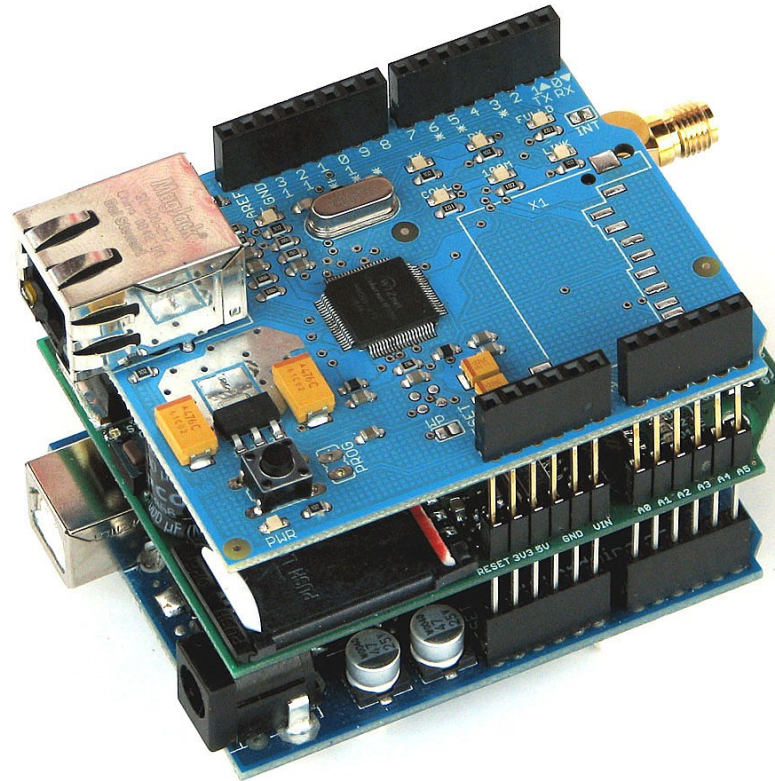
- Anatomin hos Arduino UNO



6 st 10-bitars
analoga
ingångar

Arduino-sköldar

- Standardmodulkort – ”Sköldar” (shields)
 - Koppla ihop ytterligare kort via ”piggybacking” (stapling)



Arduino-sköldar

- Exempel på "shields"



Motor-shield



Ethernet-shield



Servo-shield



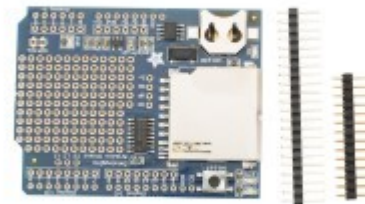
RGB-shield



Velleman IO-shield



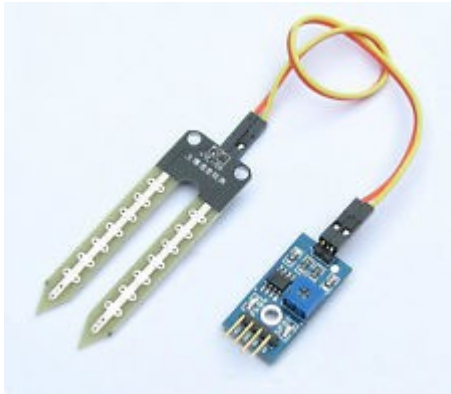
LCD-shield



Datalogger-shield

Arduino-tillbehör

- Sensorer



Hygrometer (fuktighetssensor)



Ultraljudssensor (avståndsmätning)



Mikrofonmodul (ljuddetektor)



IR-sensor (rörelsedetektor)



Accelerometer



IR-modul (hinderdetektering)

Arduino-tilbehör

- Annan kringutrustning till Arduino



Fjärrkontroll (IR)



Kameramodul



LCD display (20x4)



OLED-display (128x64)



Relämodul



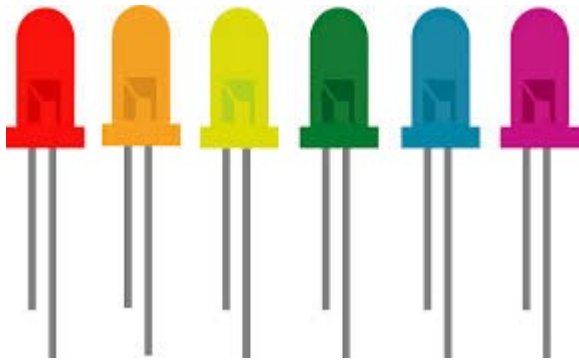
Stegmotordrivenhet



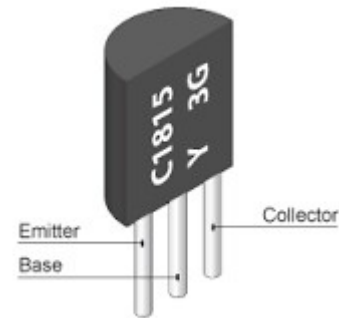
DC-motordrivenhet

Allmänna tillbehör

- Diskreta komponenter

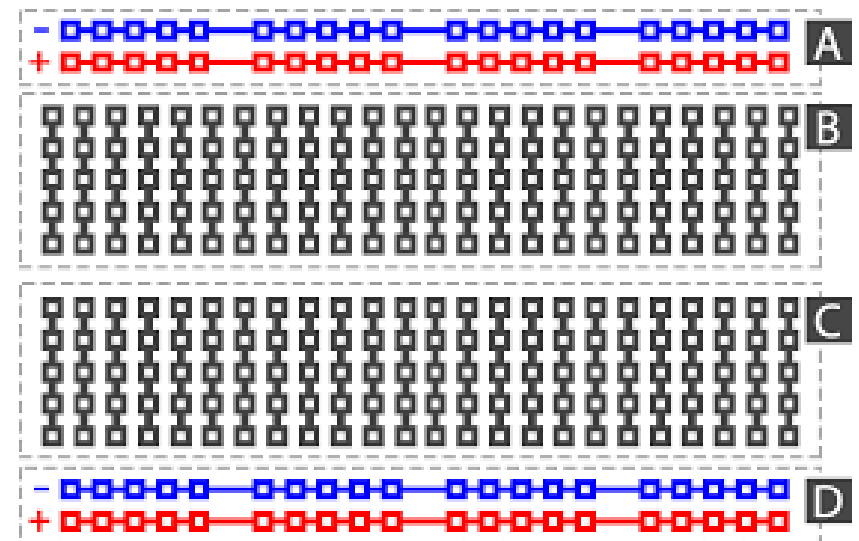
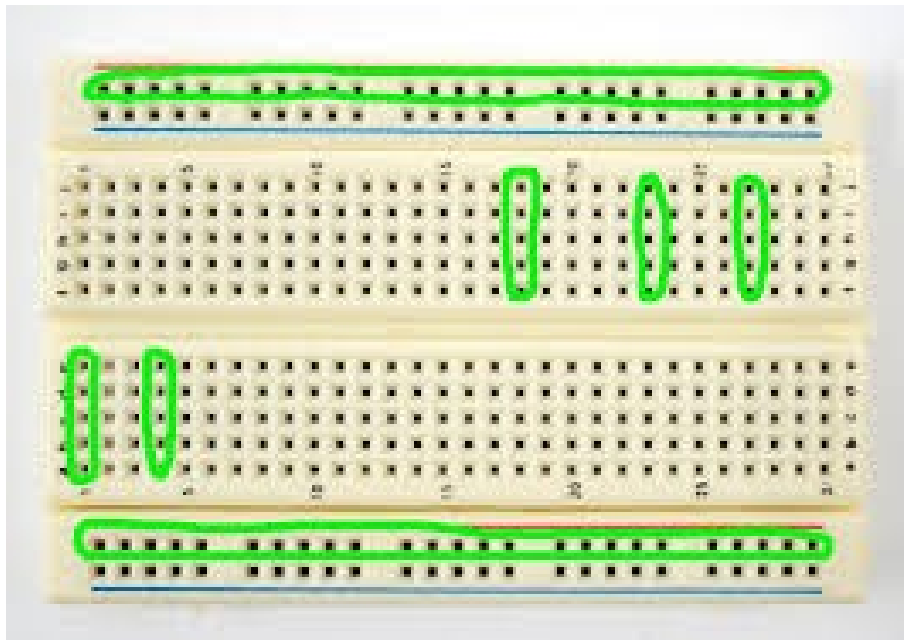


wiseGEEK



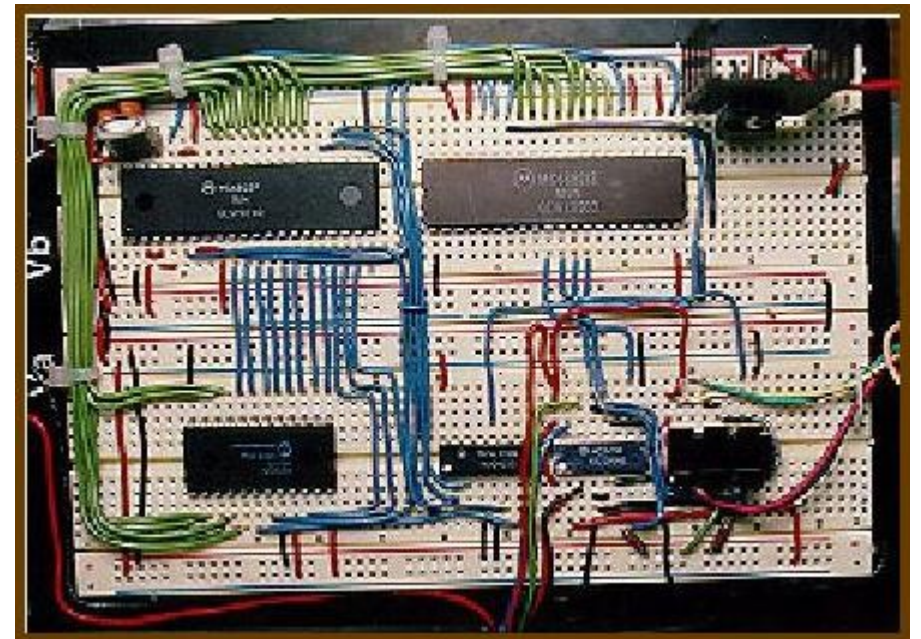
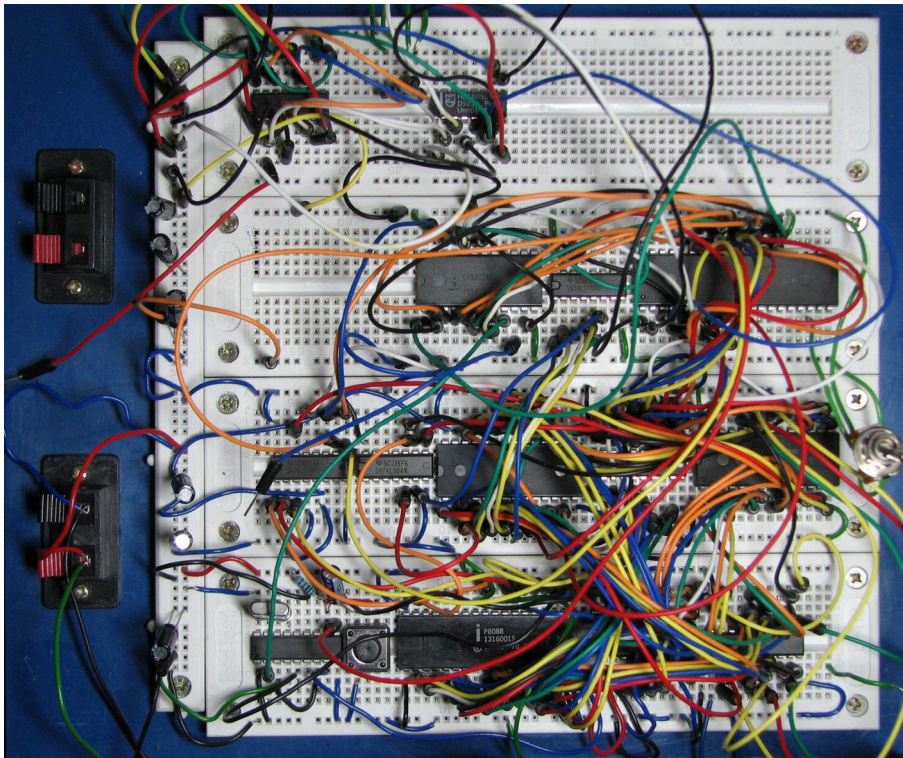
Kopplingsdäck

- Ledningsbanor på ett kopplingsdäck (breadboard)
 - Spänningskenorna (de långa skenorna på tvären)
 - Komponentskenorna (de korta skenorna på höjden)



Kopplingsdäck

- Olika kopplingsstilar



Programmering

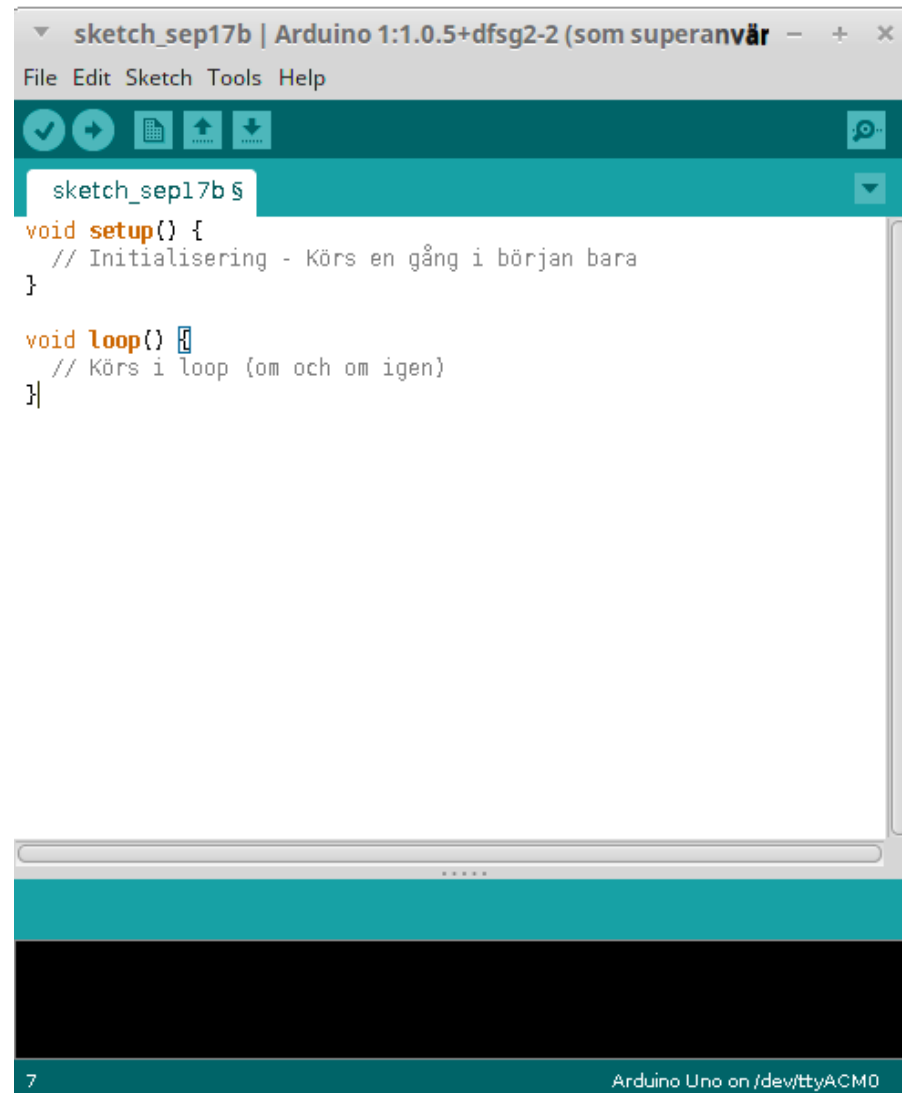
- Hämta programutvecklingsmiljön från
<https://www.arduino.cc/en/Main/Software>
 - Linux (32- el. 64-bitars)
 - Mac OS X (10.7 el. nyare)
 - Windows

Programmering

- Grundstruktur

- `setup()`
Initiering:
Körs bara en gång i början

- `loop()`
Slinga:
Körs hela tiden upprepat



```
sketch_sep17b | Arduino 1:1.0.5+dfsg2-2 (som superanvär - + x
File Edit Sketch Tools Help
sketch_sep17b §
void setup() {
  // Initialisering - Körs en gång i början bara
}
void loop() {
  // Körs i loop (om och om igen)
}
```

7 Arduino Uno on /dev/ttyACM0

Programmering

- Exempel:
"Blink"

```
int led = 13;
```

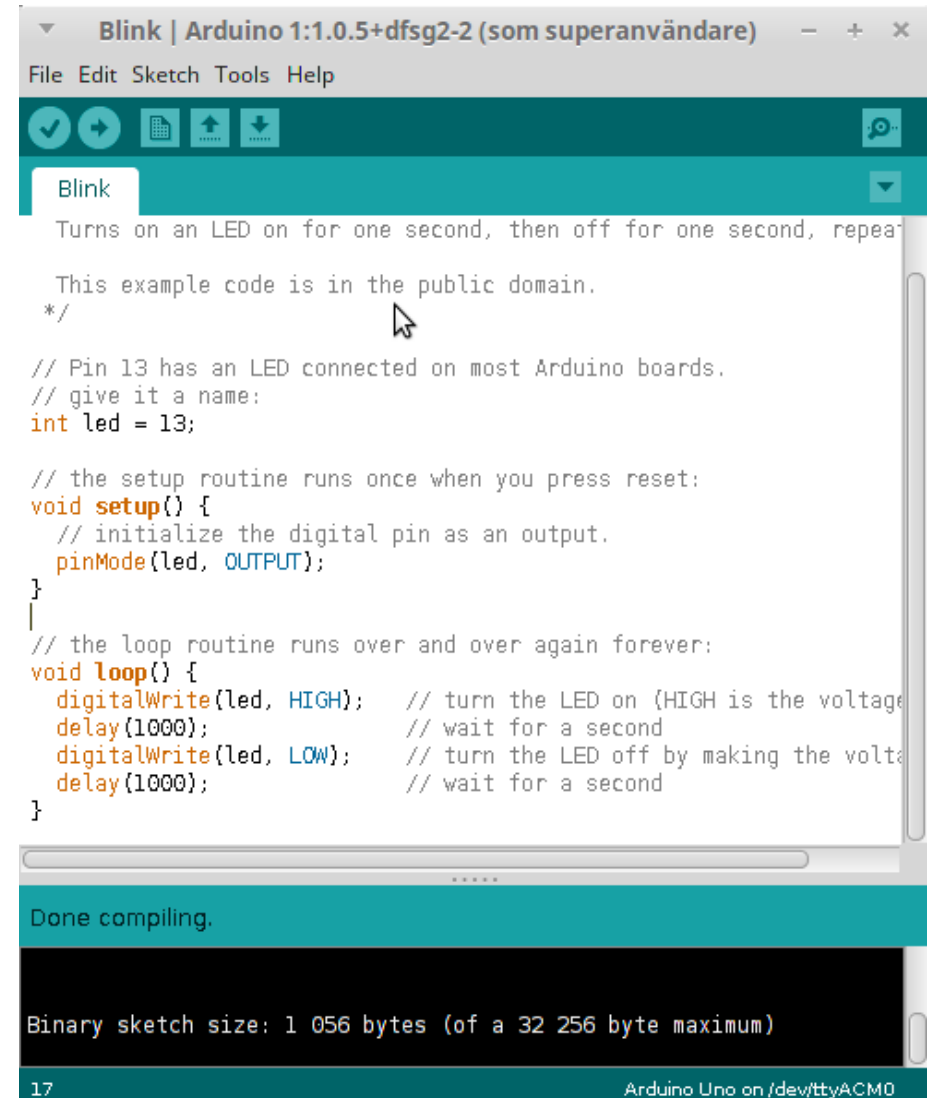
- Heltalet led tilldelas 13

```
digitalWrite(led, HIGH);
```

- Sätt digital utgång 13 hög

```
delay(1000);
```

- Fördröjning med 1000 ms



The screenshot shows the Arduino IDE interface. The title bar reads "Blink | Arduino 1:1.0.5+dfsg2-2 (som superanvändare)". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". The toolbar contains icons for saving, running, and uploading. The sketch name "Blink" is shown in a dropdown menu. The main text area contains the following code:

```
Turns on an LED on for one second, then off for one second, repeats.

This example code is in the public domain.
*/

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

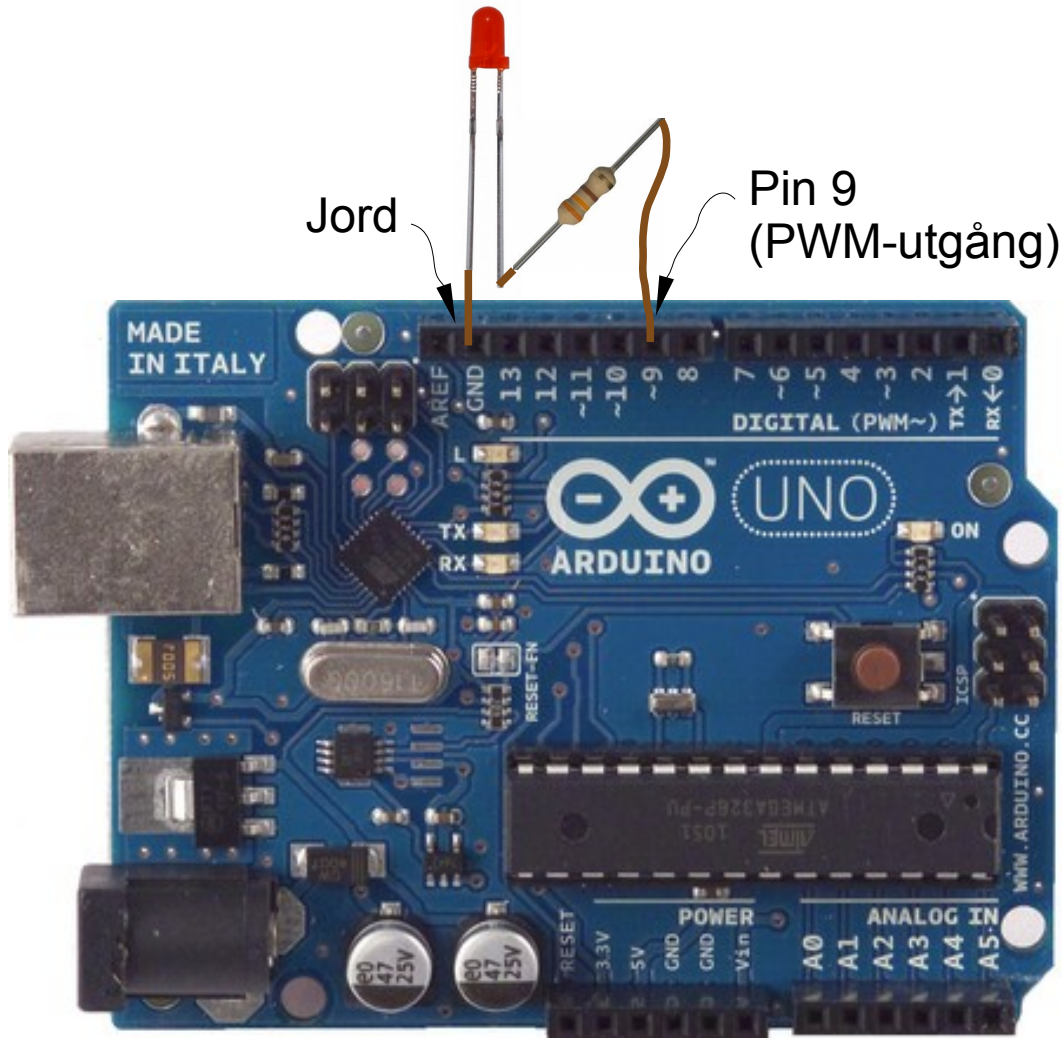
// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
```

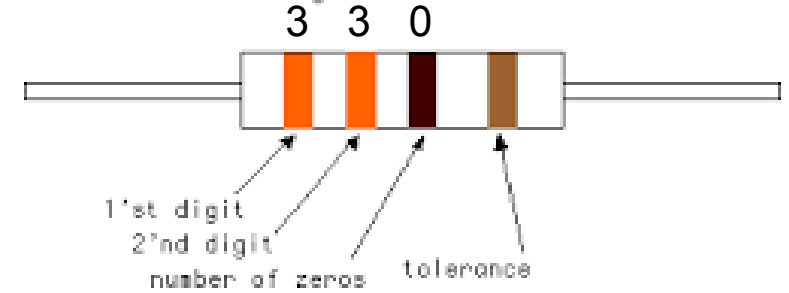
At the bottom, a status bar indicates "Done compiling." and a terminal window shows "Binary sketch size: 1 056 bytes (of a 32 256 byte maximum)". The page number "17" and "Arduino Uno on /dev/ttyACM0" are visible in the footer.

Programmering

- Koppla in extern lysdiod + motstånd (330 Ω)



Resistor color Coding



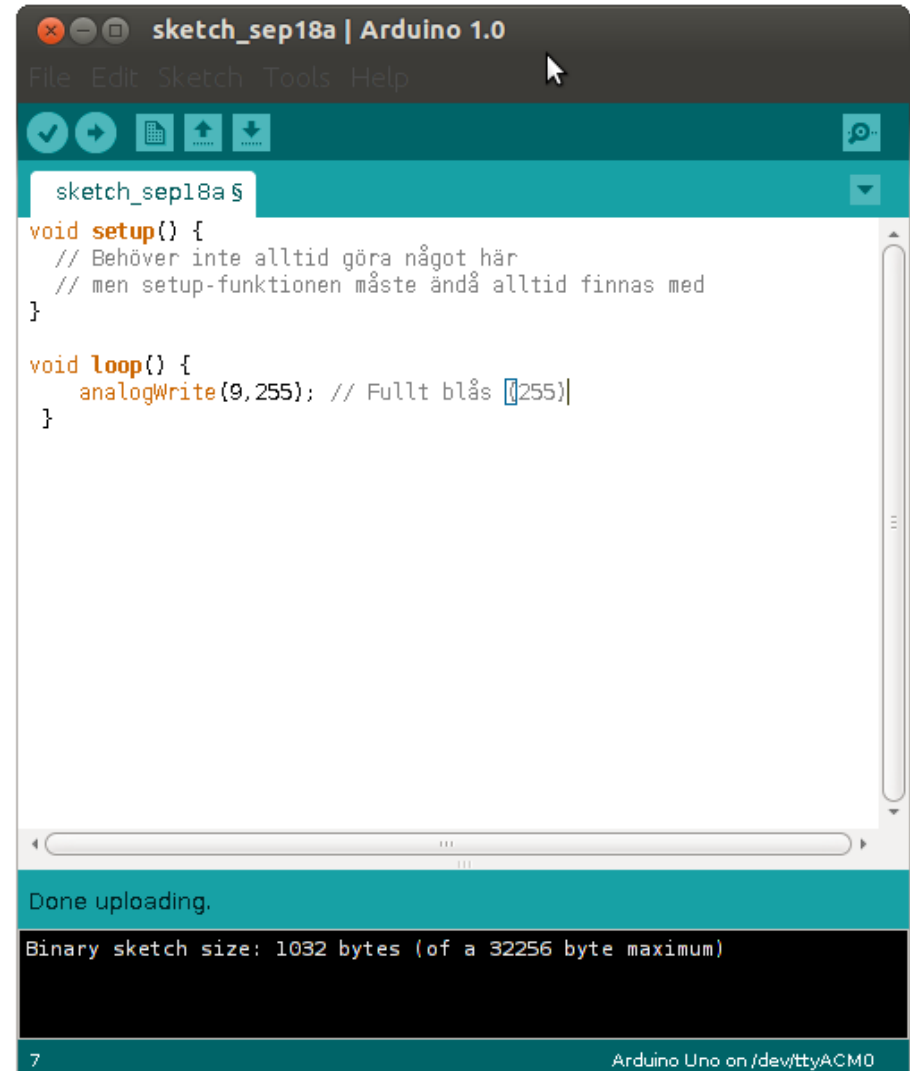
Digit	color	Tolerance	color
0	Black	20%	nothing
1	Brown	10%	Silver
2	Red	5%	Gold
3	Orange	2%	Red
4	Yellow	1%	Brown
5	Green		
6	Blue		
7	Violet		
8	Grey		
9	White		

Programmering

- Tända lysdioden
(max ljusstyrka = 255)

- Skriv 255 till pin 9:

```
analogWrite(9, 255);
```



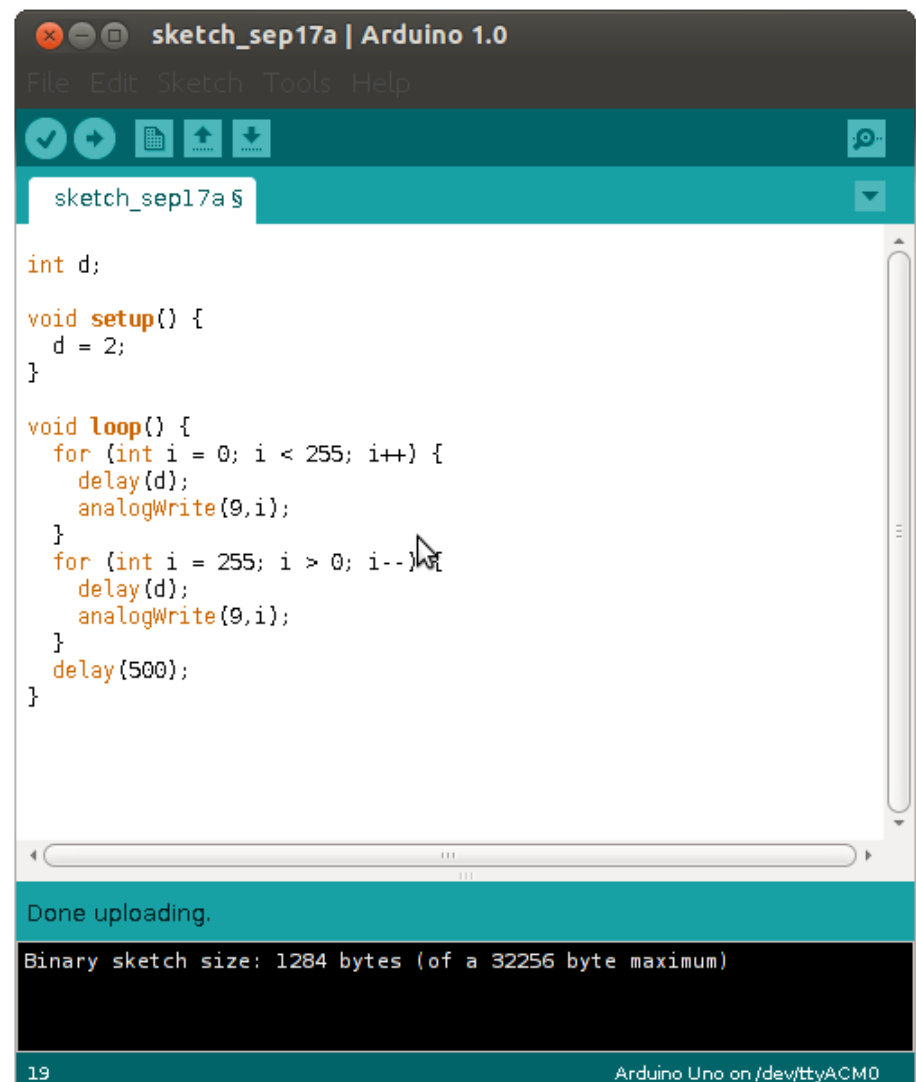
```
sketch_sep18a | Arduino 1.0
File Edit Sketch Tools Help
sketch_sep18a.g
void setup() {
  // Behöver inte alltid göra något här
  // men setup-funktionen måste ändå alltid finnas med
}

void loop() {
  analogWrite(9, 255); // Fullt blås [255]
}

Done uploading.
Binary sketch size: 1032 bytes (of a 32256 byte maximum)
7 Arduino Uno on /dev/ttyACM0
```

Programmering

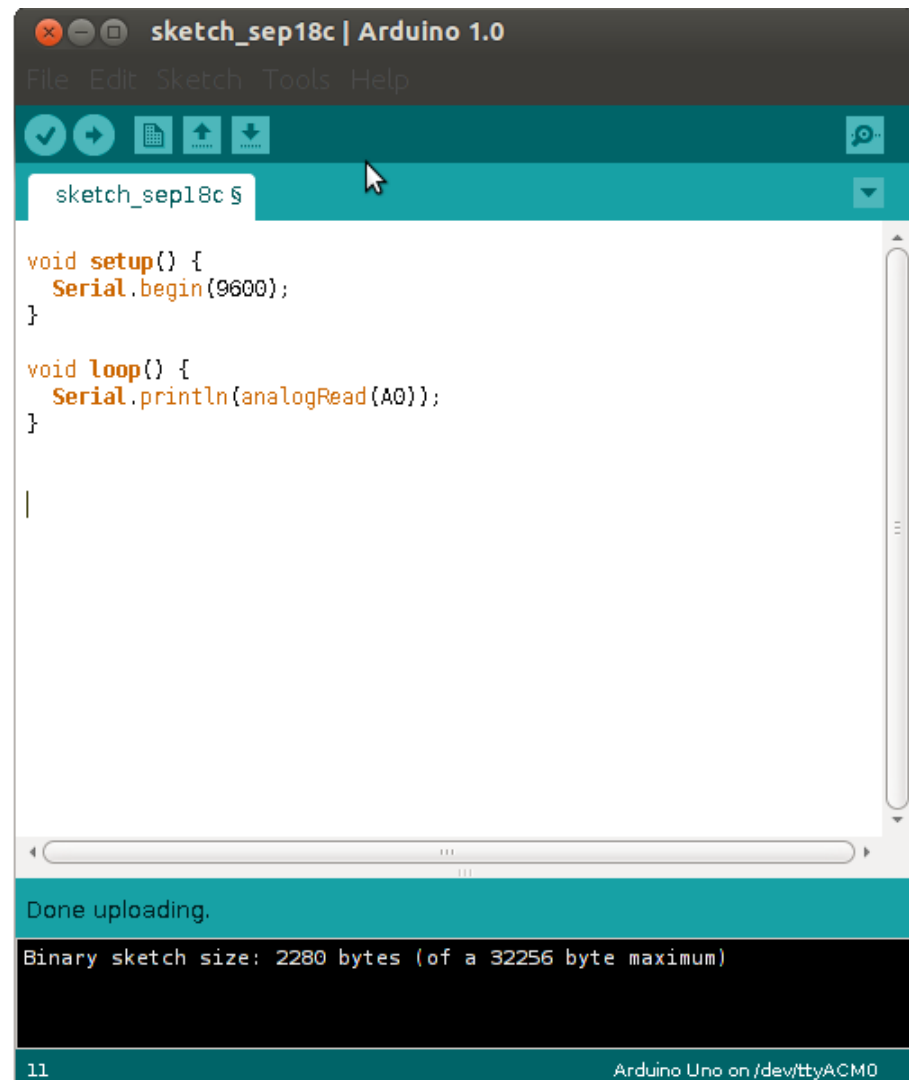
- Kul effekt: Tända och släcka lysdioden gradvis ("fade-in – fade-out")
 - Rampa upp från 0 till 255 och sen tillbaks igen
 - Heltalet d anger antalet ms mellan varje värde



```
sketch_sep17a | Arduino 1.0
File Edit Sketch Tools Help
sketch_sep17a S
int d;
void setup() {
  d = 2;
}
void loop() {
  for (int i = 0; i < 255; i++) {
    delay(d);
    analogWrite(9,i);
  }
  for (int i = 255; i > 0; i--) {
    delay(d);
    analogWrite(9,i);
  }
  delay(500);
}
Done uploading.
Binary sketch size: 1284 bytes (of a 32256 byte maximum)
19 Arduino Uno on /dev/ttyACM0
```

Programmering

- Hur får man in signaler till Arduinon då och hur ser man att det fungerar?
 - Testa `analogRead` och `Serial.println`
 - 10-bitars AD-omvandlare ger värden mellan 0 och 1023
 - Analogingången som används är A0



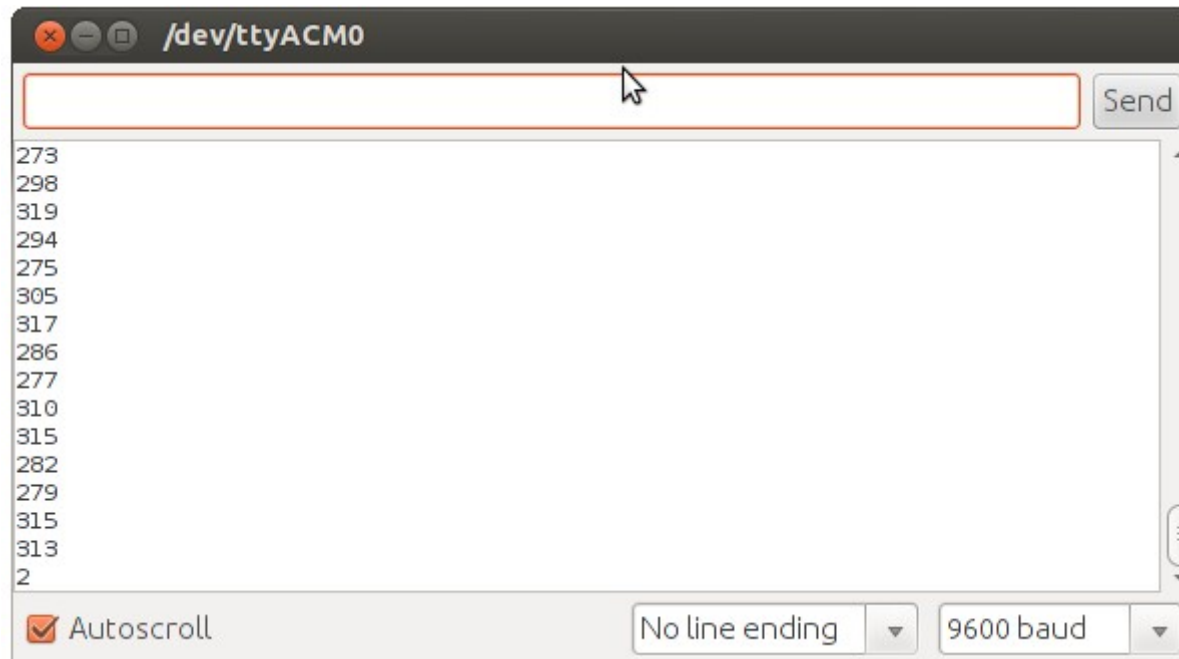
```
void setup() {  
  Serial.begin(9600);  
}  
  
void loop() {  
  Serial.println(analogRead(A0));  
}
```

Done uploading.
Binary sketch size: 2280 bytes (of a 32256 byte maximum)

11 Arduino Uno on /dev/ttyACM0

Programmering

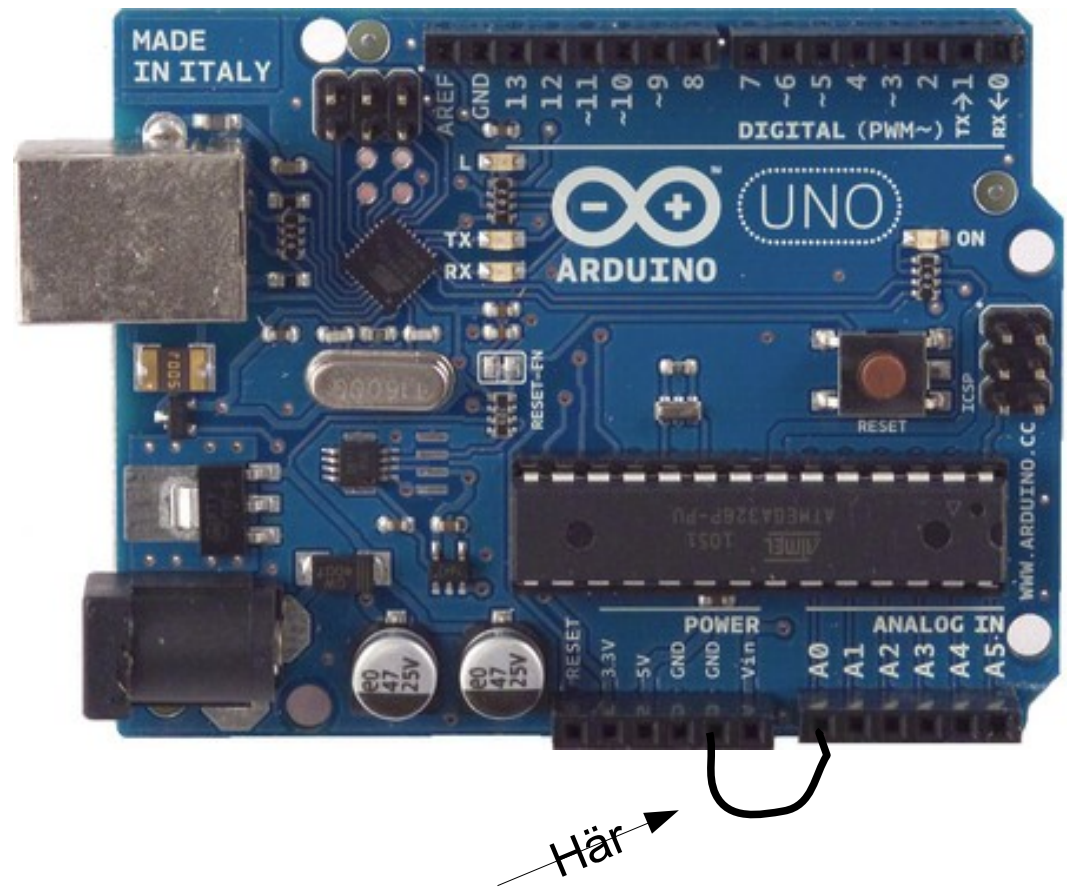
- Varför händer det inget när man kör programmet?
 - Aktivera SerialMonitor via menyn: Tools-->Serial Monitor eller via Ctrl+Shift+M



Programmering

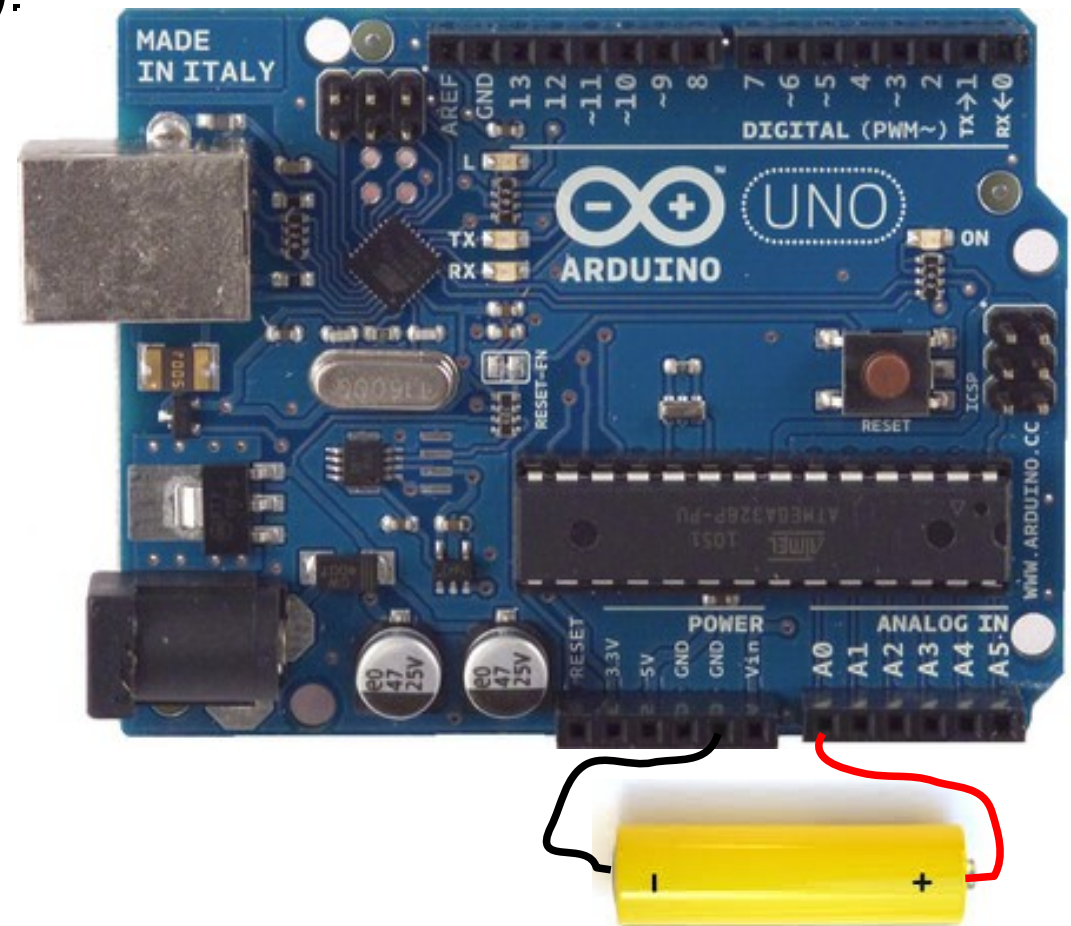
- Men det blev ju bara en massa tresiffriga tal som fladdrar förbi !?
- Testa att koppla in en ledning mellan GND (jord) och A0-ingången

Vad hände?



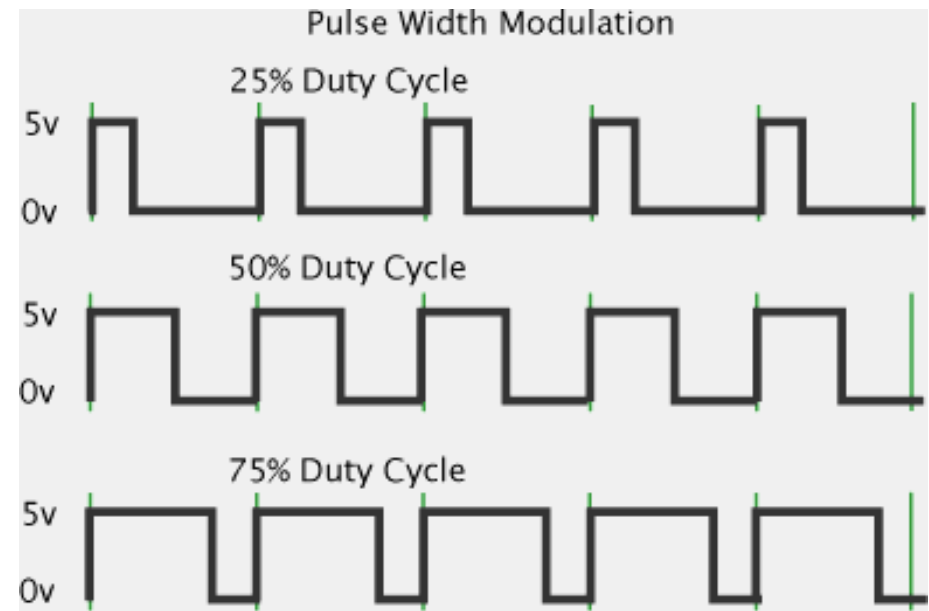
Programmering

- Testa att koppla in någon spänningskälla (0 – 5 V).
Tänk på att 0-1023 svarar mot 0-5 V



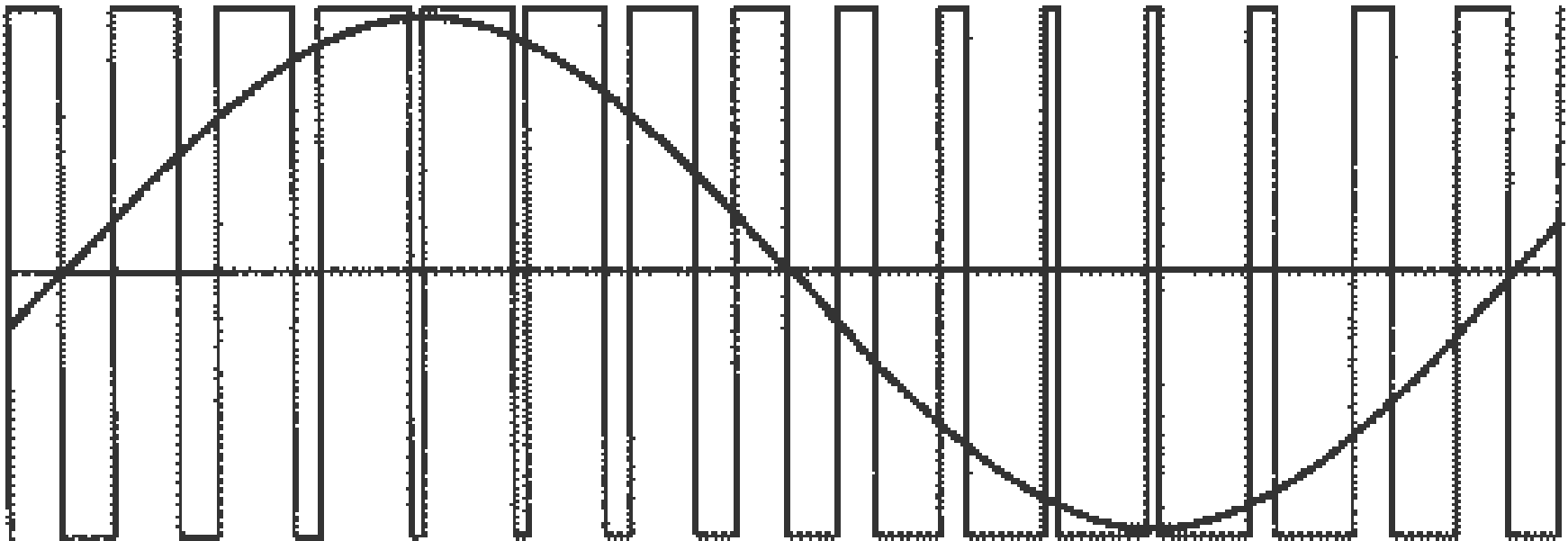
Vadå PWM, förresten?

- Vad var nu PWM för något?
 - Pulsbreddsmodulering (Pulse Width Modulation)
 - Snabb växling mellan två lägen (min och max)
 - Bredare puls istället för högre värde



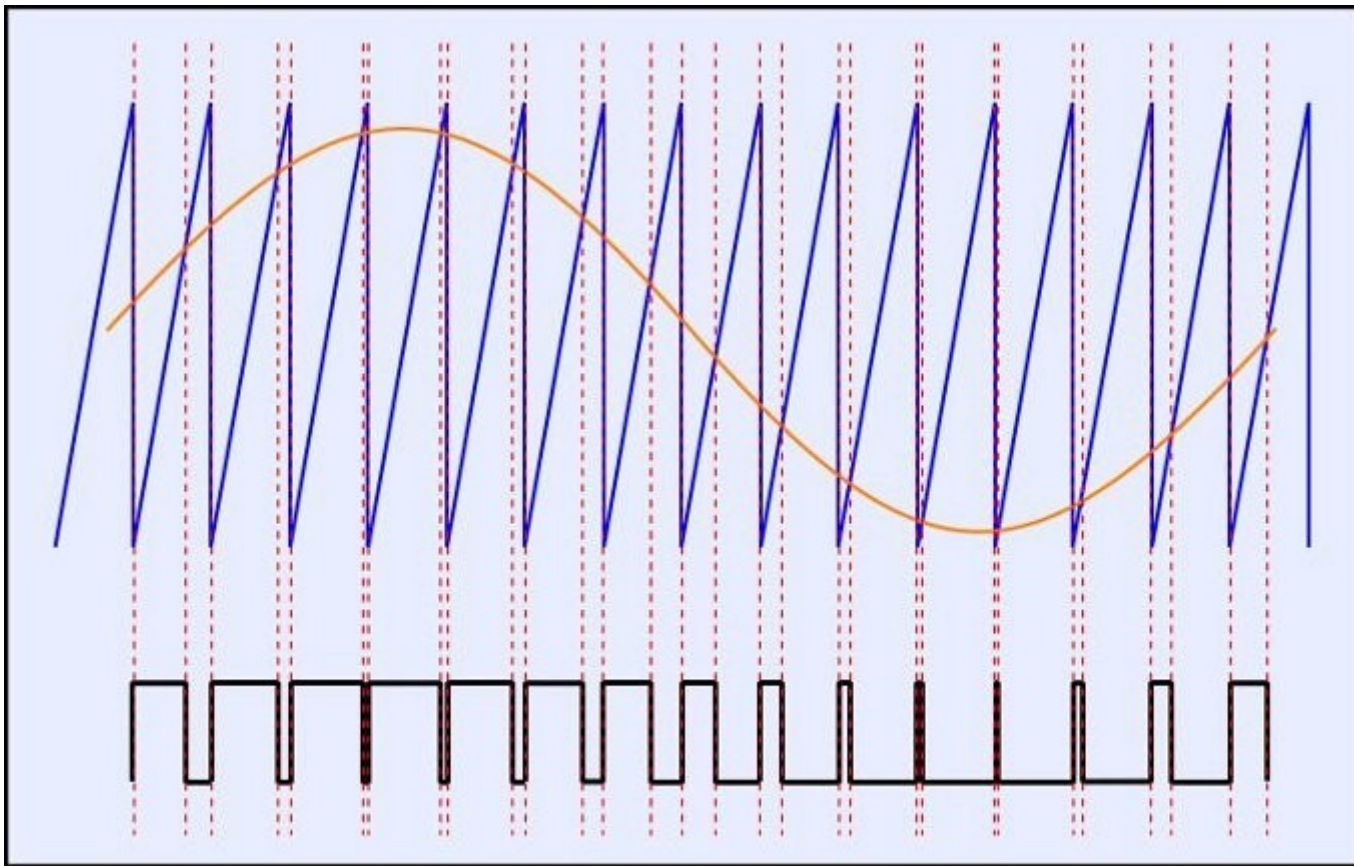
PWM

- Exempel: Sinussignal översatt till PWM



PWM

- Hur kodas sinussignalen till PWM?



PWM

- Hur kan det fungera att använda PWM istället för en vanlig analog signal? Svar: Lågpassfiltrering.

PWM to analog voltage

