

CANSAT & ARDUINO

step by step

Jens Dalsgaard Nielsen
SATLAB
Aalborg Universitet
Danmark

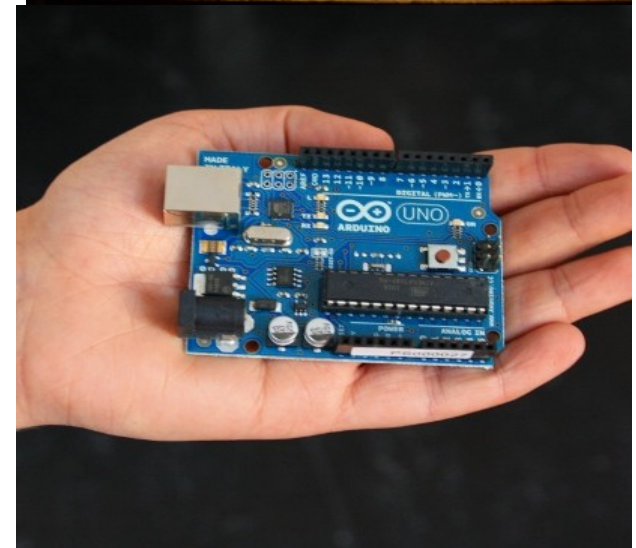
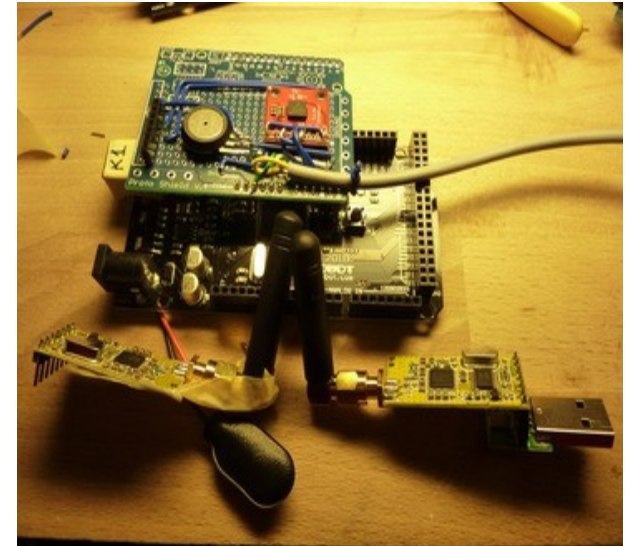
jdn@space.aau.dk

Arduino CANSAT - MÅL

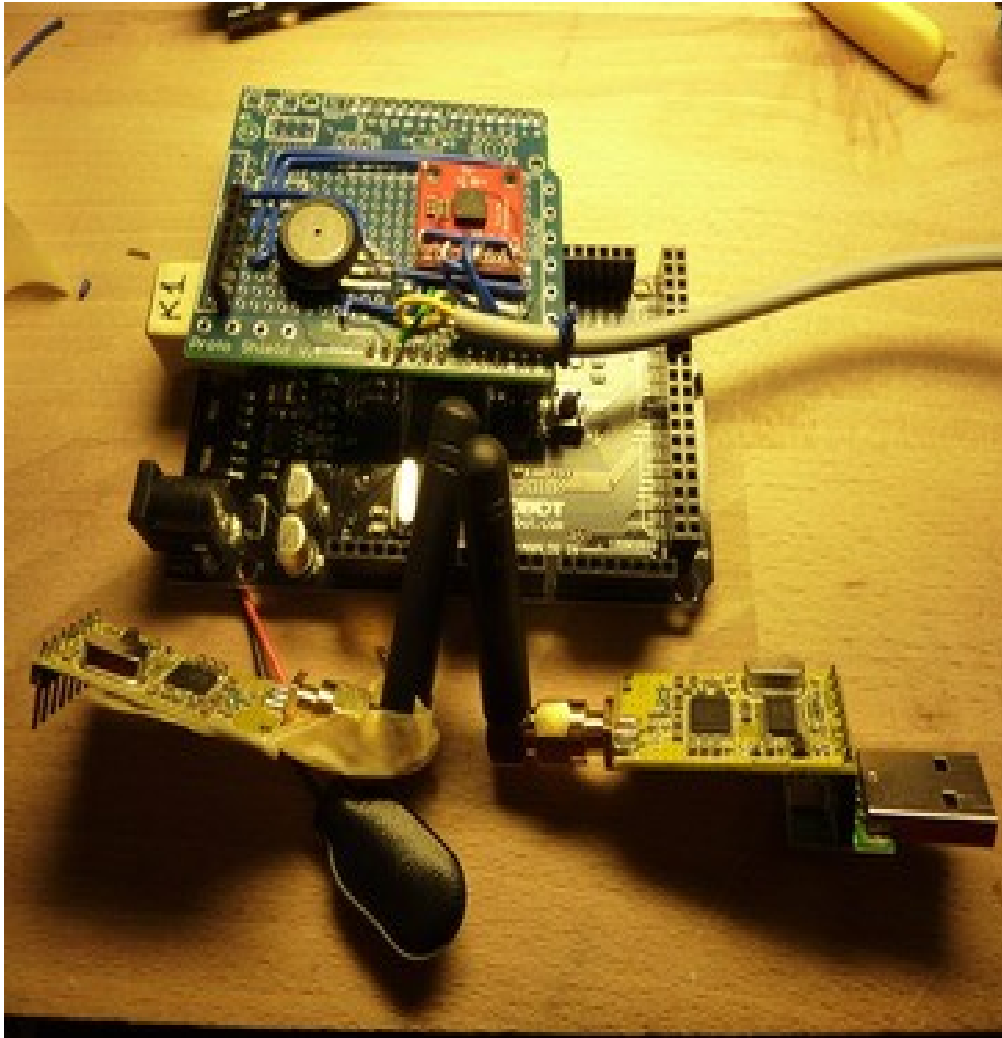
- At måle ved hjælp af sensor
 - temperatur, tryk, acceleration, CO2, lys,...
- At sende målte data til jord
- At gemme målte data ombord i CANSAT (backup)

CANSAT & ARDUINO

- CAN SAT HW
 - dåse :-)
 - Ramme/skelet
 - Indmad
 - Arduino micro controller
 - Sensorer
 - (aktuatorer)
 - Radiolink til jord
 - Onboard Storage (SD kort)
 - Faldskærm
 - Strømforsyning: 9V batteri



SW & HW



CANSAT & ARDUINO

- CANSAT SW
 - Bruger skrevet program i sproget C/C++
 - minimalversion på mindre end 10 linier kode
- Jordstation
 - Serielt standard program
 - teraterm(windows), arduino konsol, ...
 - et program der kan sende og modtage karakterer på en seriel port (rs232, usb-rs232)

Det første problem - Installation

- Arduino SW <http://arduino.cc/en/Main/Software>
 - specielt for windows xp/vista/7 – special attention
 - der skal installeres drivere til det specifikke Arduino board:

- uno, mega,...

- Følg instrukser



Arduino Uno



Arduino Leonardo



Arduino WiFi Shield



Arduino Mega 2560



Arduino LilyPad



Arduino Ethernet Shield



Er det muligt ?

- Gjorde det i lørdags på XP og windows 7 for Arduino Uno – så JA :-)
- Er det svært ?
- NEJ ? men læs vejledning først.
- Et check: W7 uden drivere blinker TX lysdioden IKKE når arduino sender data over serielle forbindelse

Andre operativsystemer ?

- Linux – ingen installation af ekstra drivere
- MAC – ingen installation af ekstra drivere

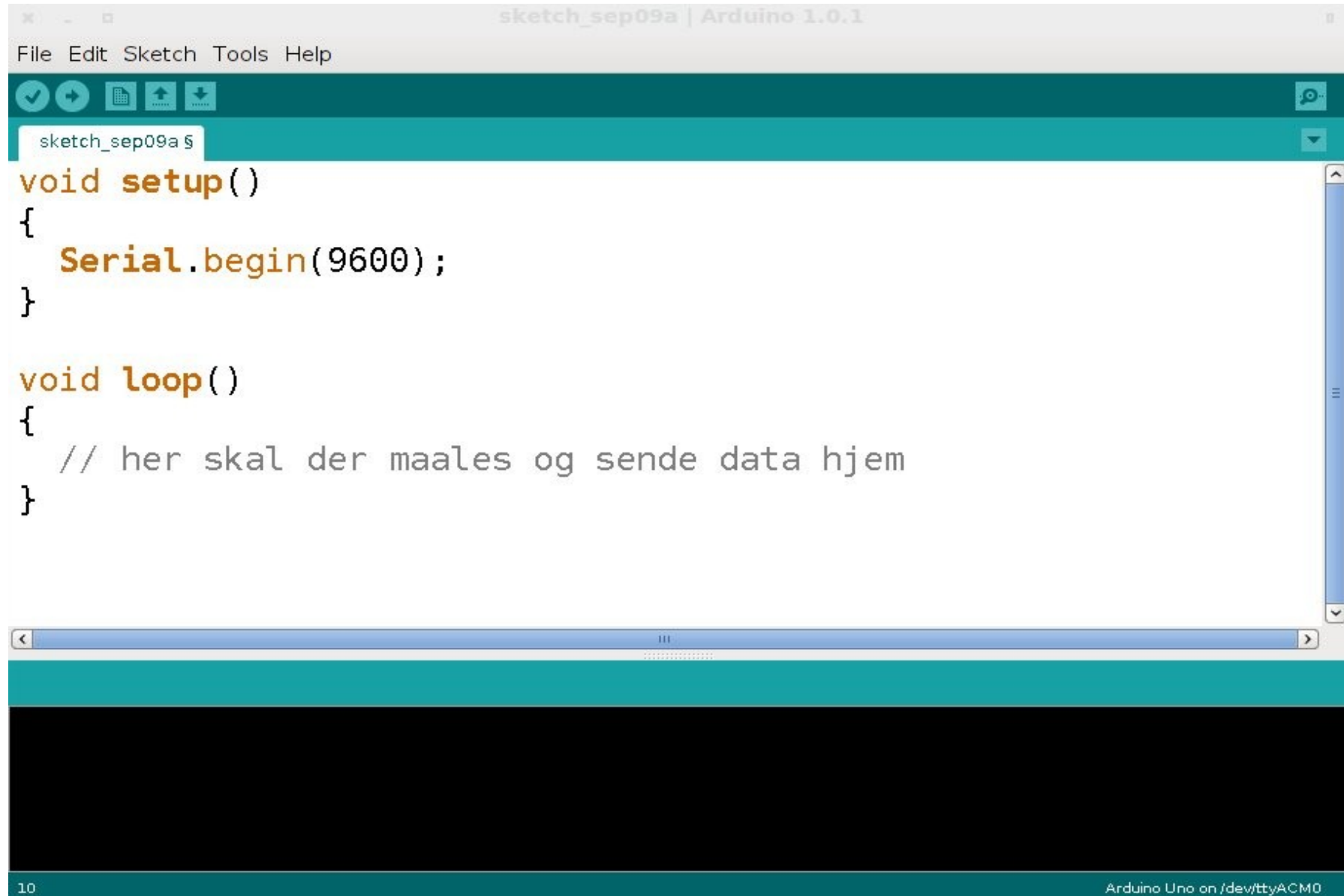
Jord segment

- Windows : teraterm – gratis
- Linux: minicom gratis
- MAC : lignende terminal

mere om dette senere

- Arduino programstruktur
- Læs sensor
- fortolke læsning
- sende data hjem

Arduino programstruktur



```
sketch_sep09a | Arduino 1.0.1
File Edit Sketch Tools Help
sketch_sep09a $
void setup()
{
  Serial.begin(9600);
}

void loop()
{
  // her skal der maales og sende data hjem
}
```

10 Arduino Uno on /dev/ttyACM0

Kørbart eksempel



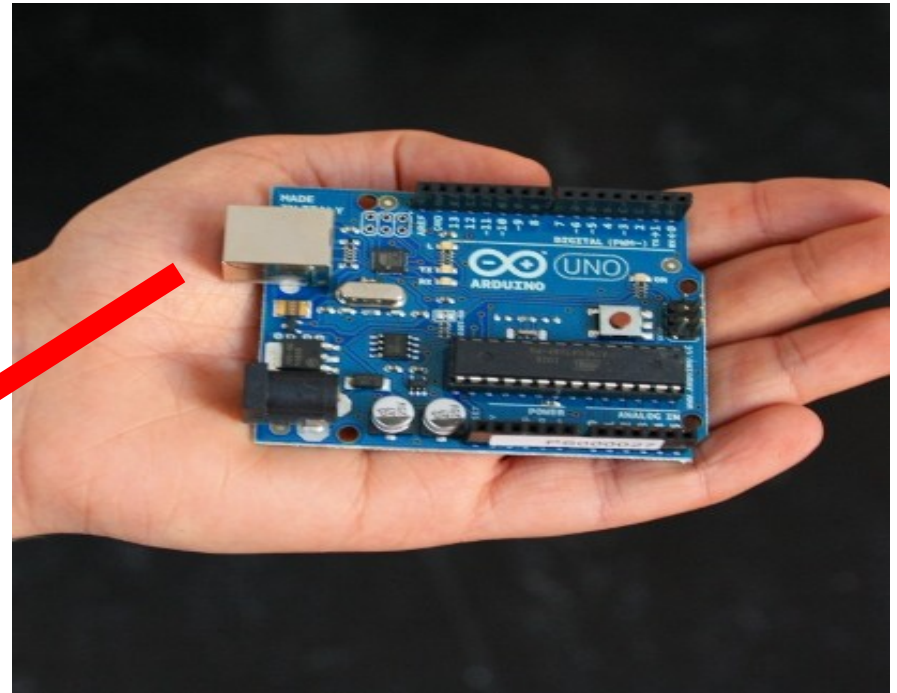
The image shows a screenshot of the Arduino IDE interface. The window title is "sketch_sep09a | Arduino 1.0.1". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for checkmark, refresh, print, upload, and download. The main editor area shows the following code:

```
sketch_sep09a 5
{
  Serial.begin(9600);
}

void loop()
{
  Serial.println("Her er jeg");
  delay(1000); // vent i 1000 millisekunder = 1 sekund
  // her skal der maales og sende data hjem
}
```

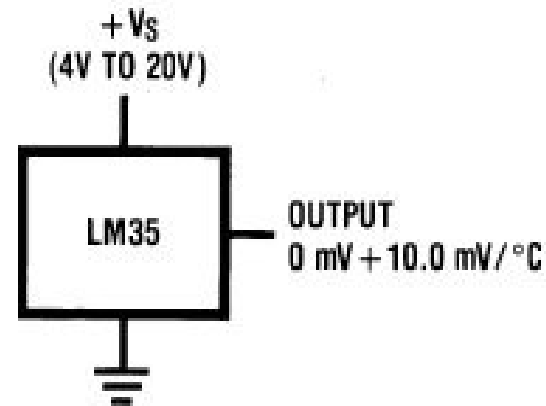
Eksempel

her er jeg
her er jeg
her er jeg
her er jeg
(med 1 sekund mellem linierne)



Sensor – temperatur - LM35

- Lineær:
 - 10 mV/grad Celcius
 - fra +2 til +150 gr Celcius



DS005516-3

**FIGURE 1. Basic Centigrade Temperature Sensor
(+2°C to +150°C)**

- `analogRead(kanal)`

analog måling:

- Arduino har 6 analoge kanaler der kan måle spænding i området 0-5V.

- Temperatur
 - `analogRead(5)`

sensor	input	analog kanal
NTC	temperatur	0
MPX4115	tryk	1
MMA7361L	accelerometer x akse	2
MMA7361L	accelerometer y akse	3
MMA7361L	accelerometer z akse	4
LM35	temperatur	5

Gemning af værdier

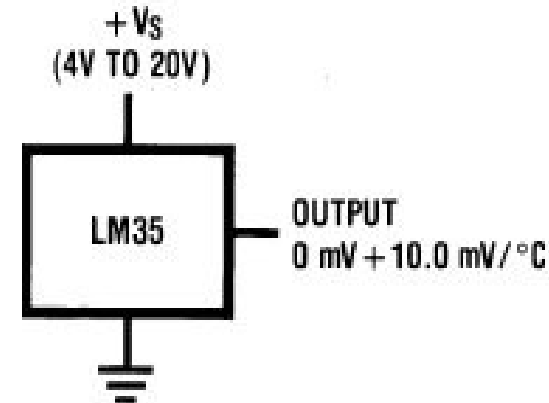
- heltal (int) [-32768 ... 0 ... 32767]
- kommmatal (float) - $1 \cdot 10^{38} \dots 1 \cdot 10^{38}$ ca
- Arduino måler spænding i heltal
 - 0V \approx 0 - i måletal
 - 5V \approx 1023
- så man skal selv regne om !

Første måling

```
sketch_sep09a §  
void setup()  
{  
  Serial.begin(9600);  
}  
  
int maaling; // oprettelse af variabel/storage/"register"  
void loop()  
{  
  maaling = analogRead(5); // temperatur kanal  
  Serial.println(maaling);  
  delay(1000); // vent i 1000 millisekunder = 1 sekund  
}
```

Første måling - II

- Indendøre : 20 grader
- $20 * 10\text{mV} = 200 \text{ mV}$
 - $= 0.2\text{V}$



DS005516-3

FIGURE 1. Basic Centigrade Temperature Sensor (+2°C to +150°C)

- Vi måler:
 - 40
- hvorfor ?

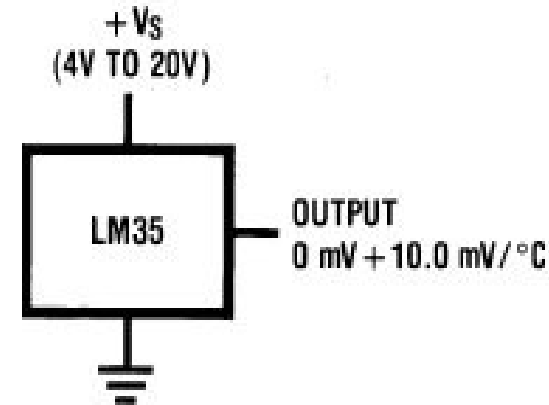
```
sketch_sep09a 5
void setup()
{
  Serial.begin(9600);
}

int maaling; // oprettelse af variabel/storage/"register"
void loop()
{
  maaling = analogRead(5); // temperatur kanal
  Serial.println(maaling);
  delay(1000); // vent i 1000 millisekunder = 1 sekund
}
```

Første måling - II

- Indendøre : 20 grader
- $20 * 10\text{mV} = 200\text{mV}$
- Vi måler 40
- hvorfor ?

- fordi
- $(0.2\text{V}/5\text{V}) * 1023 = 40$
- altså ingen Volt



DS005516-3

FIGURE 1. Basic Centigrade Temperature Sensor (+2°C to +150°C)

```
sketch_sep09a 5
void setup()
{
  Serial.begin(9600);
}

int maaling; // oprettelse af variabel/storage/"register"
void loop()
{
  maaling = analogRead(5); // temperatur kanal
  Serial.println(maaling);
  delay(1000); // vent i 1000 millisekunder = 1 sekund
}
```

Ingeniør/SI enheder

- 1 tælleantal svarer til 5/1023 V
- ok ???? - ledende spørgsmål

```
int maaling; // oprettelse af variabel/storage/"register"  
void loop()  
{  
  maaling = analogRead(5); // temperatur kanal  
  maaling = 5*maaling/1023; // FIRST TRIAL !!!  
  Serial.println(maaling);  
  delay(1000); // vent i 1000 millisekunder = 1 sekund  
}
```

Ingeniør/SI enheder

- NOPE – vi arbejder jo i heltal
- 20 grader – tælletal 40
- $5 \cdot 40 = 200$ $200 / 1023 = 0$!!! (der rundes ned)

```
int maaling; // oprettelse af variabel/storage/"register"  
void loop()  
{  
  maaling = analogRead(5); // temperatur kanal  
  maaling = 5*maaling/1023; // FIRST TRIAL !!!  
  Serial.println(maaling);  
  delay(1000); // vent i 1000 millisekunder = 1 sekund  
}
```

Løsning 1

- send tælletal (40)
- lav selv omregning “hjemme på PC”

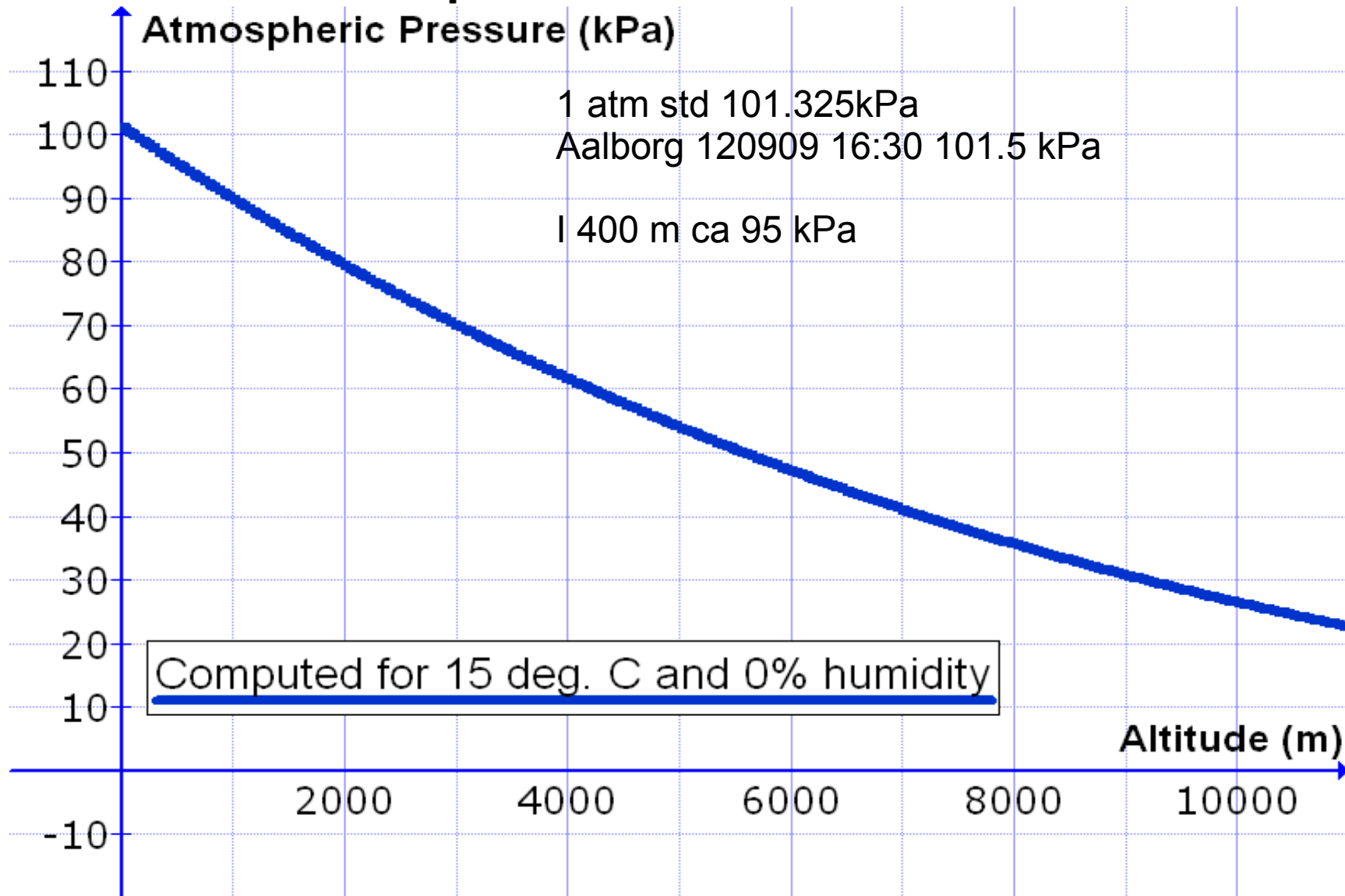
Løsning 2 - kommatatal

```
int maaling; // oprettelse af variabel/storage/"register"  
float temp; ←  
void loop()  
{  
  maaling = analogRead(5); // temperatur kanal  
  temp = (5.0*maaling)/1023.0; // 5.0 og 1023.0 ikke 5 og 1023 ! ←  
  Serial.println(temp);  
  delay(1000); // vent i 1000 millisekunder = 1 sekund  
}
```

“tvinges” over og regne i kommatatal

Tryk og højde

Atmospheric Pressure vs. Altitude



Tryk sensor – mpx 4115

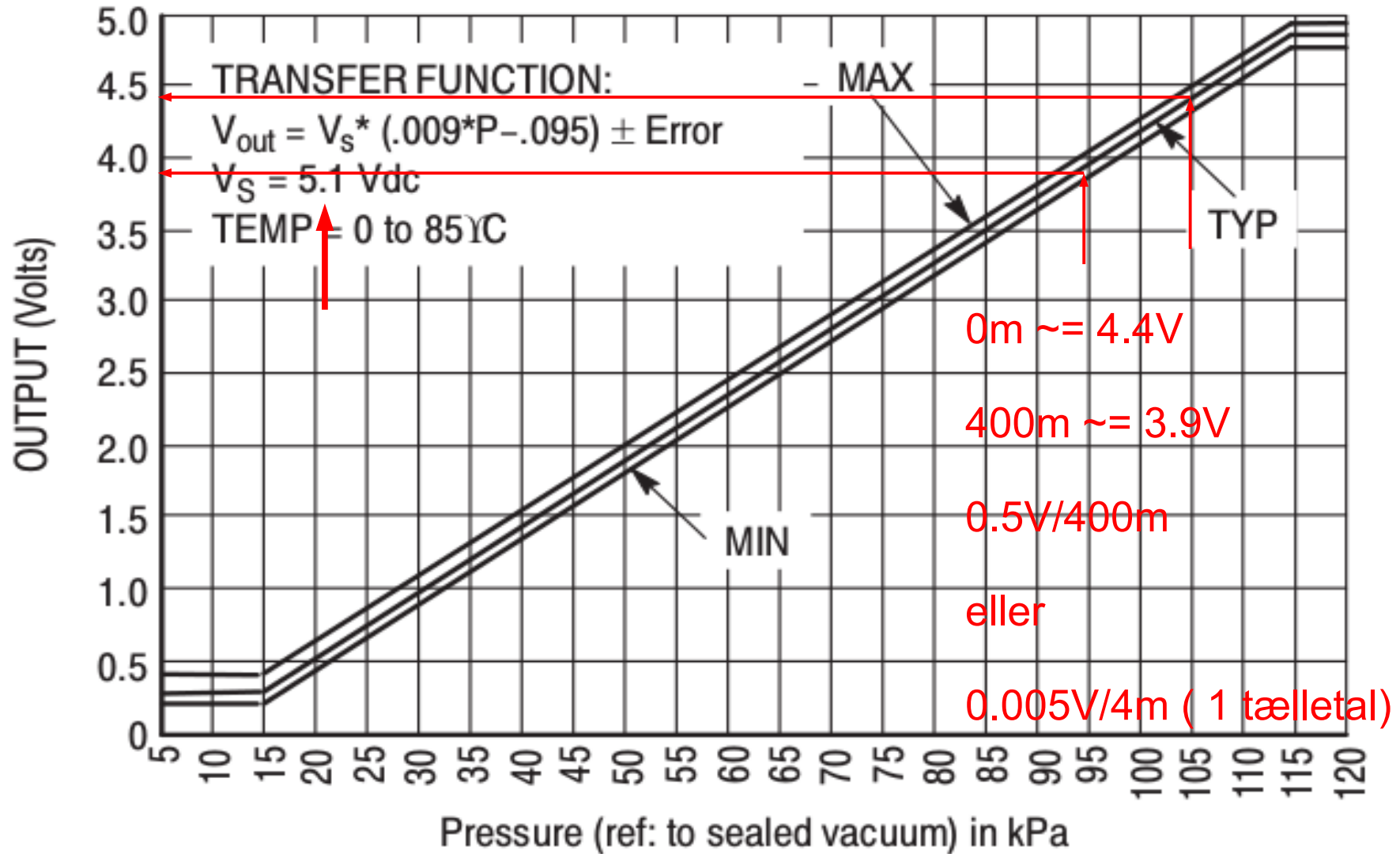


Figure 4. Output versus Absolute Pressure

Accelerometer

Output Signal

Zero-g ($T_A = 25^\circ\text{C}$, $V_{DD} = 3.3\text{ V}$)^{(5), (6)}

Zero-g⁽⁴⁾

Sensitivity ($T_A = 25^\circ\text{C}$, $V_{DD} = 3.3\text{ V}$)

1.5g

6g

Sensitivity⁽⁴⁾

Bandwidth Response

XY

Z

Output Impedance

0g-Detect

V_{OFF}	1.485	1.65	1.815	V
V_{OFF, T_A}	-2.0	± 0.5	+2.0	mg/ $^\circ\text{C}$
$S_{1.5g}$	740	800	860	mV/g
S_{6g}	190.6	206	221.5	mV/g
S, T_A	-0.0075	± 0.002	+0.0075	%/ $^\circ\text{C}$
f_{-3dBXY}	—	400	—	Hz
f_{-3dBZ}	—	300	—	Hz
Z_O	—	32	—	k Ω
$0g_{detect}$	-0.4	0	+0.4	g

Storage, radio osv osv

- intet specielt !
- Alt der skrives med `Serial.print(ln)`
- skrives også til
 - radio
 - sdkort

så ...

```
void setup()  
{  
  Serial.begin(9600);  
}  
int maaling; // oprettelse af variabel/storage/"register"  
void loop()  
{  
  int i;  
  
  for (i=0; i<6;i++)  
  {  
    maaling = analogRead(i);  
    Serial.print(maaling);  
    Serial.print(" , ");  
  }  
  Serial.println(" "); // ny linie  
  delay(1000); // vent i 1000 millisekunder = 1 sekund  
}
```

Digitale input/output

- Der er digital input og output:
- høj/sand $\sim = 5V$
- lav/falsk $\sim = 0V$
- Kan benyttes til kontakter mm

```
void setup()
{
  Serial.begin(9600);
  pinMode(12,INPUT);
  pinMode(13,OUTPUT);
}

int maaling; // oprettelse af variabel/storage/"register"
void loop()
{
  int val;

  val = digitalRead(12);
  digitalWrite(13,val);
  // eller
  digitalWrite(13,HIGH); // eller LOW
}
```

```
int maaling; // oprettelse af variabel/storage/"register"  
void loop()  
{  
  int val;  
  
  val = digitalRead(12);  
  
  if (val == HIGH) // test - husk x=3 er en tilskrivning|  
    digitalWrite(13,HIGH);  
  else  
    digitalWrite(13,LOW); // eller LOW  
  
}
```

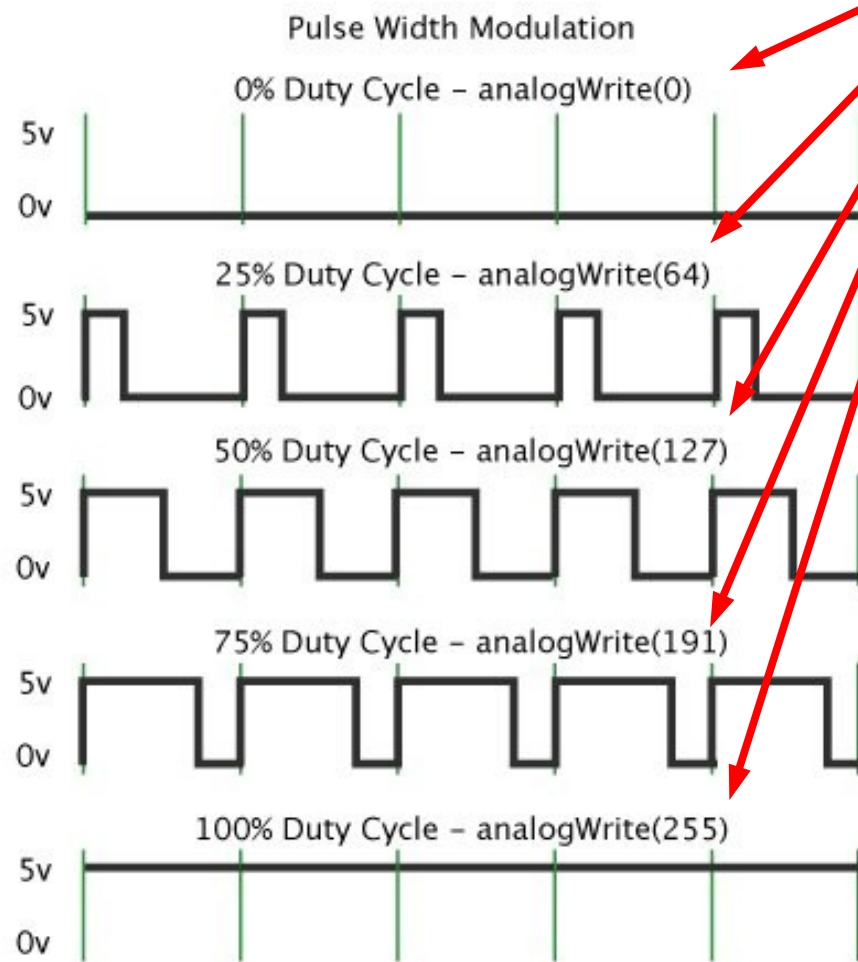

digitalt input

- Problem: hvad nu hvis der intet er tilslutte digitale indgang ?
- Så svæver den == ikke pålidelig
- Arduino kan “forbinde” digitale udgang med intern modstand til 5V.
- Ingen tilslutning på indgang så måles der høj(5V)

```
pinMode(pin, INPUT); // set pin to input  
digitalWrite(pin, HIGH); // turn on pullupresistor(20kohm)
```

Analog output

0-255 i tællertal

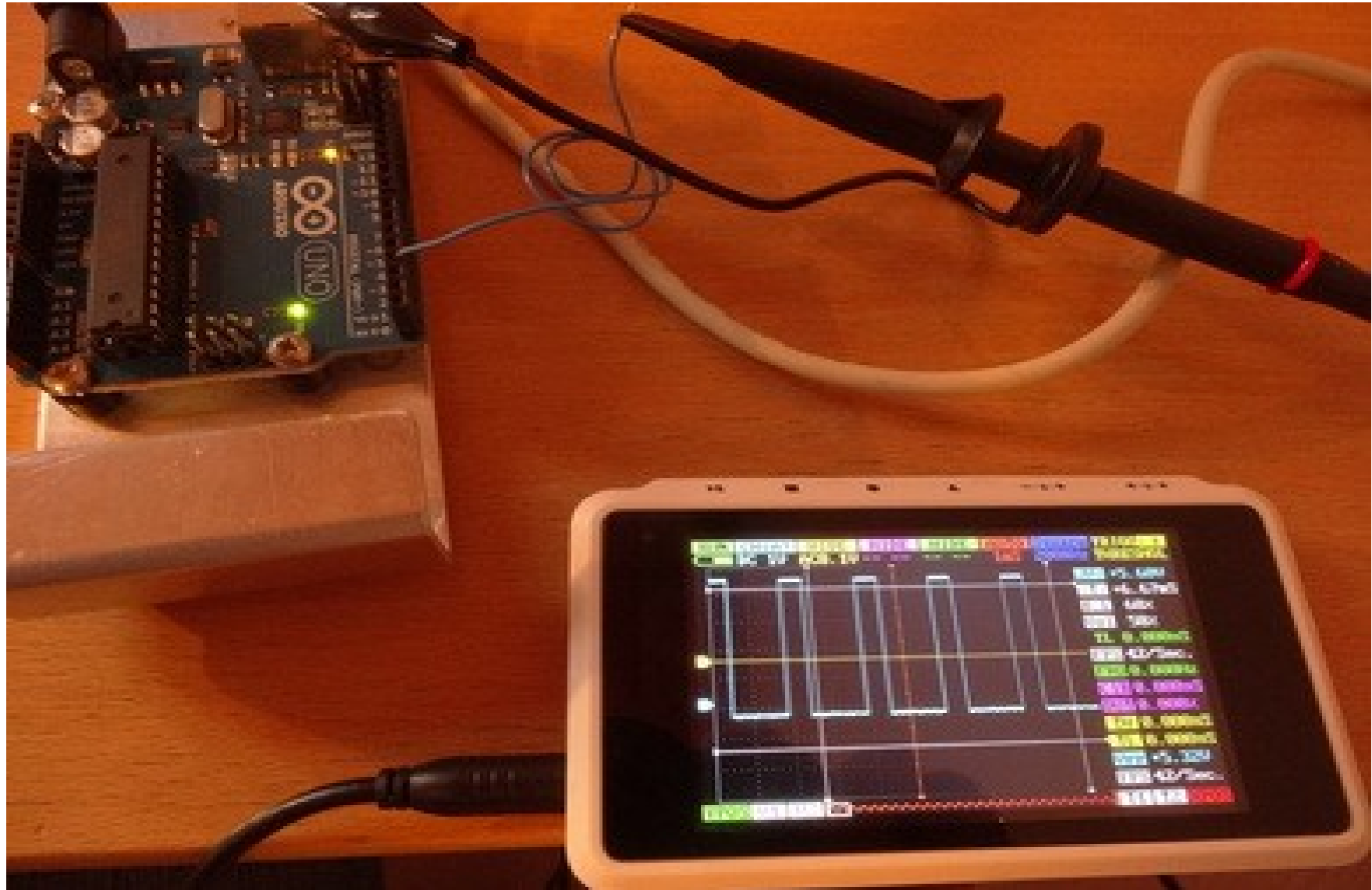


frekvens ca 490 Hz dvs ca 2 msec pr periode

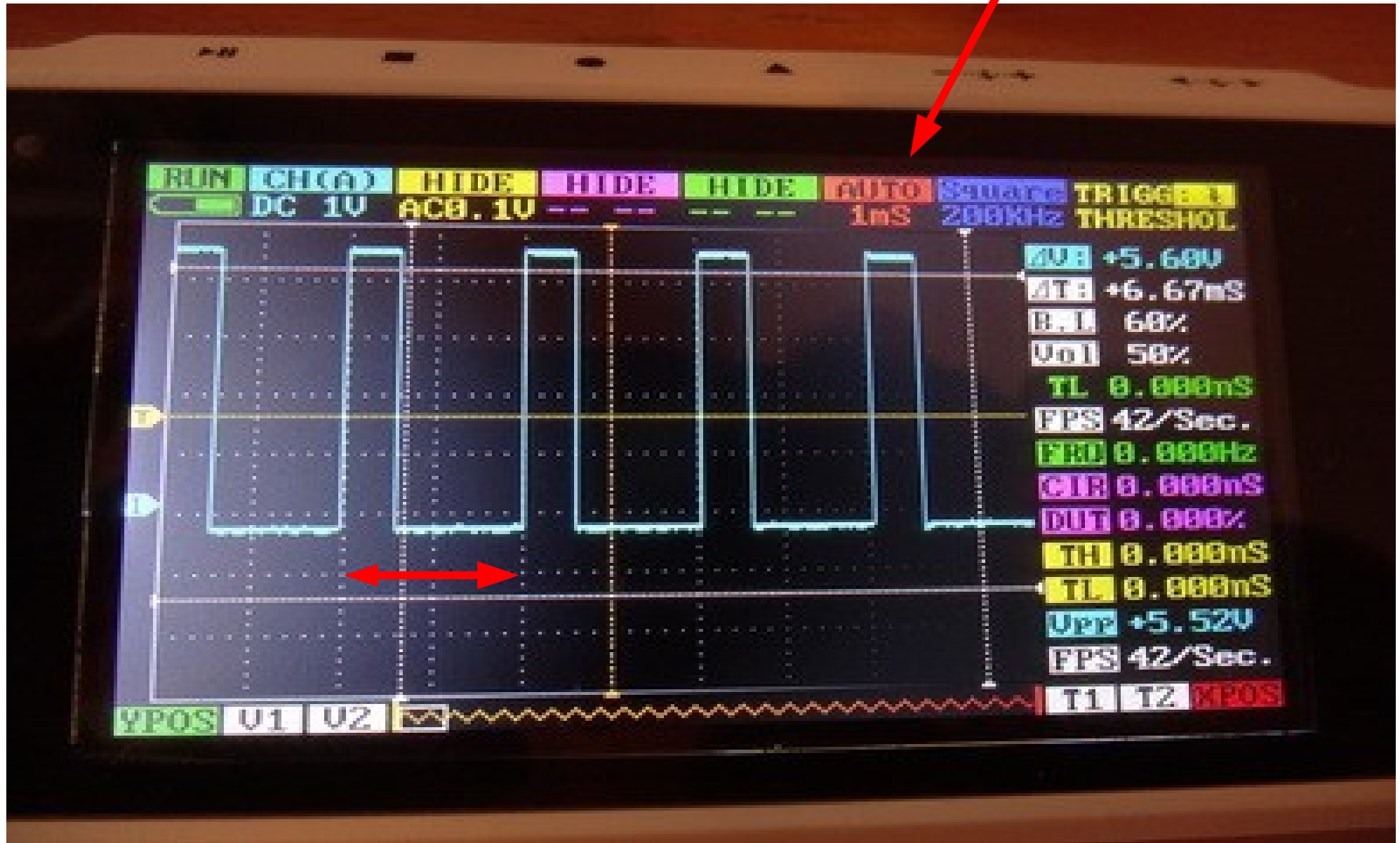
```
void setup()
{
  Serial.begin(9600);
}

int val = 0;
void loop()
{
  analogWrite(6, val);
  val = val+1; // eller val++;
  if (255 < val)
    val = 0; // sav tak generator
  delay(100); // vent 0.1 sekund
}
```

setup – simpel output

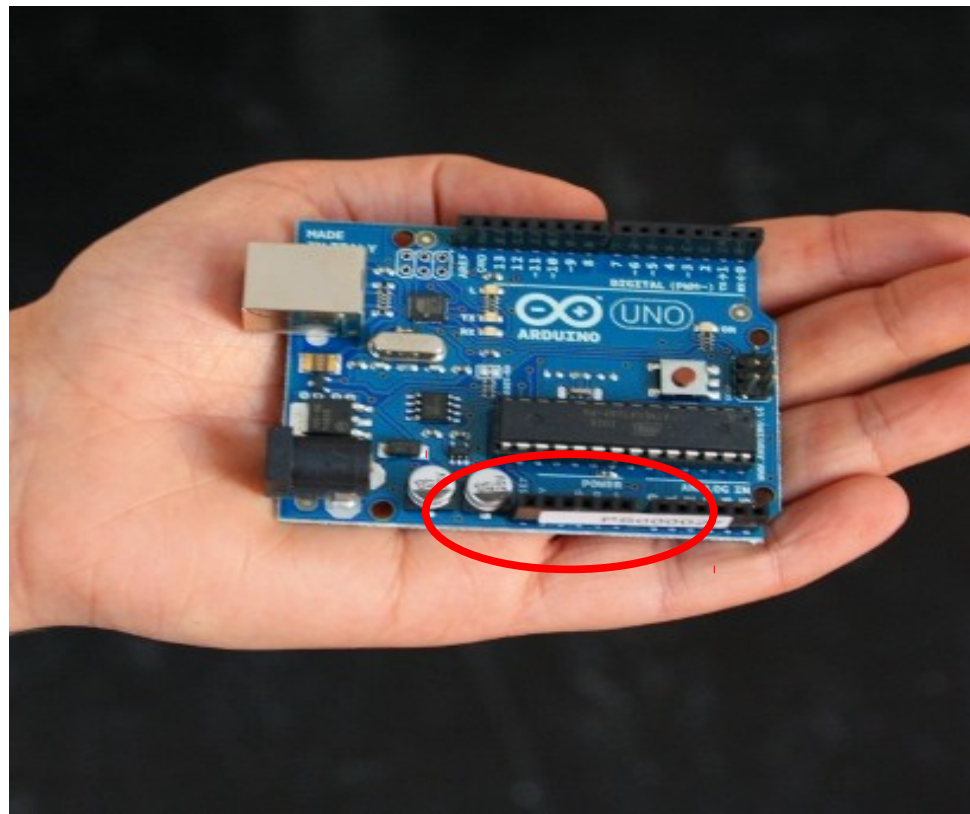


Rigtig måling



øvelse 1

- mål spænding på analog 3
- skriv ud
- sæt analog 3 med ledning fra hhv 3.3V og 5V



next step – modular coding

- Formål: at kunne forberede et forsøg
- Ide: At man – i forvejen – kan lave noget funktionalitet.
- Metode:
 - arduino projekt
 - en “skjult” fil

File Edit **Sketch** Tools Help

Verify / Compile Ctrl+R

Show Sketch Folder Ctrl+K

Add File...

Import Library... >

modulae

void se

{

Serial.begin(9600);

}

int value;

void loop()

{

delay(1000);

value = maal_temperatur(0); // på analog ch 0

}



- målt temperat
- målt tid
- målt analog

- målt
- målt
- målt



modulaer tempLib

```
void setup()  
{  
  Serial.begin(9600);  
}
```

```
int value;
```

```
void loop()  
{  
  delay(1000);  
  value = maal_temperatur(0); // på analog ch 0  
}
```

modulaer tempLib §

```
// value = maal_temperatur(0); // på analog ch 0
```

```
int maal_temperatur(int kanal)  
{  
  return analogRead(kanal);  
}
```

Funktioner - modularisering

- Det talte sprog ;-)
- setup
- loop
 - maal
 - sendData
 - SovLidt
 - gåTilTop

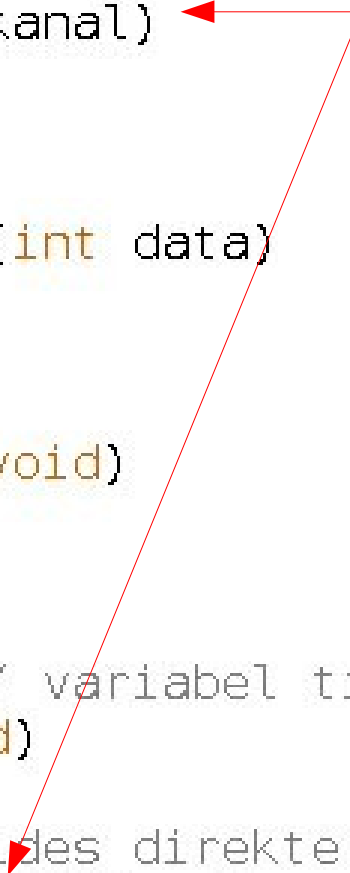
```
void setup()
{
}

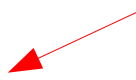
int maal(int kanal) ←
{
}

void sendData(int data)
{
}

void sovLidt(void)
{
}

int vaerdi; // variabel til at holde maaling
void loop(void)
{
    // setup kaldes direkte uden jeg skal gøre noget
    vaerdi = maal(3); // maal kanal 3
    sendData(vaerdi);
    sovLidt();
} // herfra op til start igen :-)
```





```
int maal(int kanal)
{
    int lokalVaerdi;

    lokalVaerdi = analogRead(kanal);
    return vaerdi;
}
```

til tasterne