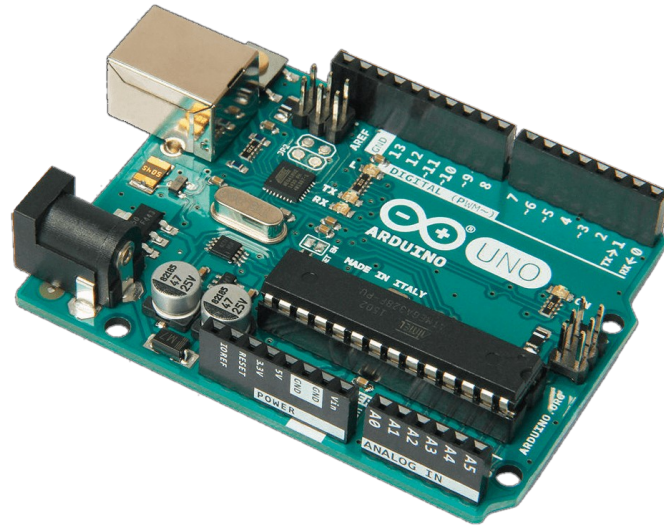
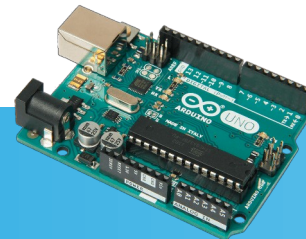


Arduino workshop #3



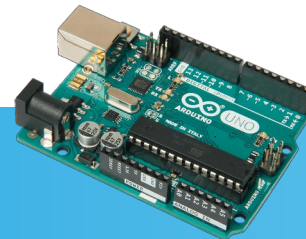
Quiz

- Hvor store tal kan repræsenteres med en int på Arduino?
- Nævn to typer løkker i C
- Hvor mange elementer har dette array: `int arr[10];`
- Hvilke værdier er lovlige index i ovenstående array?
- Hvilke pins på Arduino kan bruges til at aflæse et potentiometer (variable modstand)?
- Hvad er PWM?



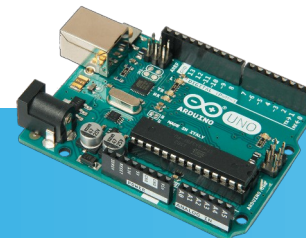
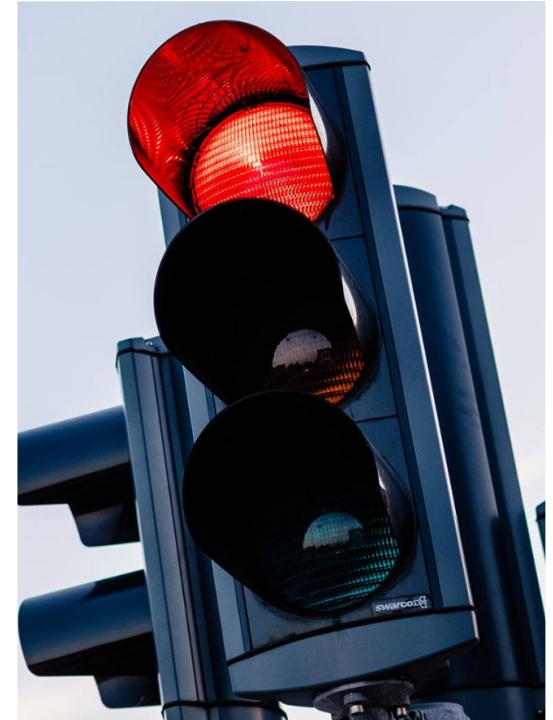
Svar

- int på Arduino er 16-bit, dvs. -32.768 til +32.767
- while-løkke, for-løkke og do-while-løkke
- Array'et har plads til 10 elementer af typen int
- Der 0-indexeres, dvs. index 0 til 9
- Analog input; A0-A5
- Pulse-Width-Modulation eller pulsbreddemodulering er at man tænder og slukker for en udgang meget hurtigt, mens man varierer hvor lang tid der er tændt og slukket.



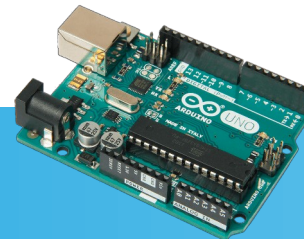
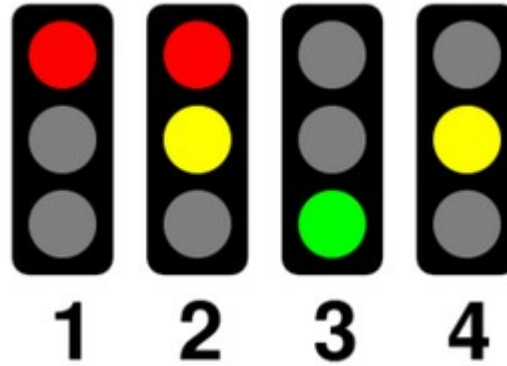
Lyssignal

- En almindelig lyskurv skifter imellem 4 forskellige tilstande
 - 1. tilstand: rødt lys
 - 2. tilstand: ???
 - 3. tilstand: ???
 - 4. tilstand: ???
- Opret en ny sketch og sæt de fire linjer ovenfor ind som kommentarer i loop-funktionen



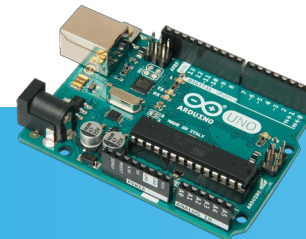
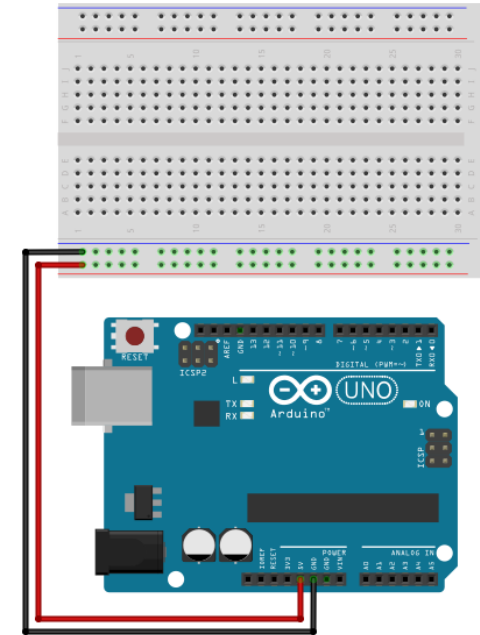
De fire tilstande

- 1. Rød
- 2. Rød + Gul
- 3. Grøn
- 4. Gul



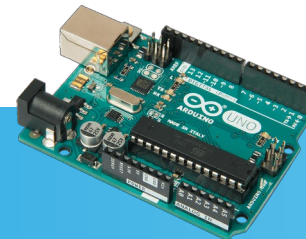
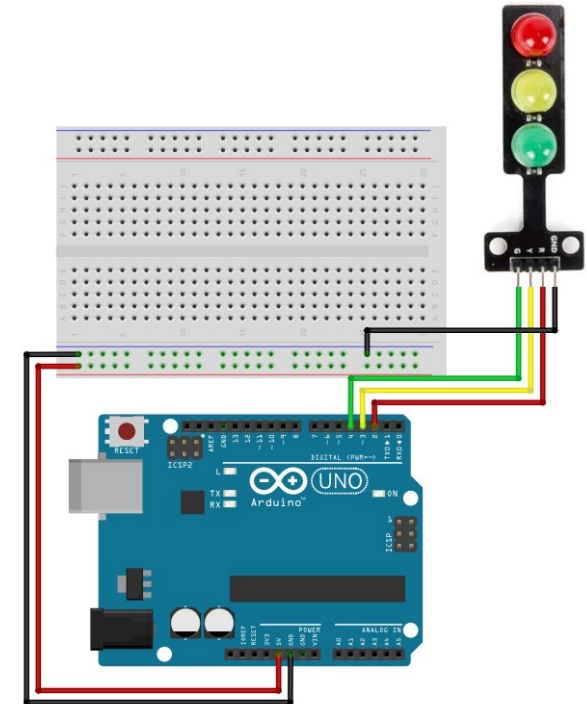
Forbind Arduino og breadboard power

- Forbind GND på Arduino til GND-bus på Breadboard (sort ledning)
- Forbind 5V på Arduino til + bus på breadboard (rød ledning)
- NB: brug M-M dupont-kabler



Forbind LED lyssignal

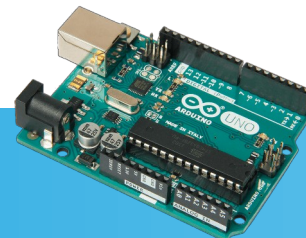
- LED lyssignalet har indbyggede formodstande til at begrænse strømmen
- Bruges der diskrete lysdioder **skal** man huske selv at tilføje en formodstand
- Tilslut lyssignalet sådan (M-F dupont):
 - Lyssignal GND til Breadboard GND
 - Lyssignal R til Arduino pin 2
 - Lyssignal Y til Arduino pin 3
 - Lyssignal G til Arduino pin 4



Initialisér udgange

- Først defineres navne til de 3 pins (før setup)
- I setup-funktionen sættes de 3 pins til udgange

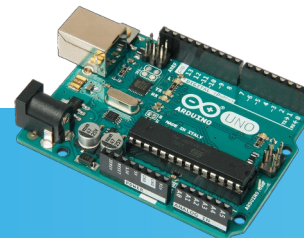
```
1  const byte traffic_lightA_red_pin = 2;  
2  const byte traffic_lightA_yellow_pin = 3;  
3  const byte traffic_lightA_green_pin = 4;  
4  
5  void setup() {  
6      pinMode(traffic_lightA_red_pin, OUTPUT);  
7      pinMode(traffic_lightA_yellow_pin, OUTPUT);  
8      pinMode(traffic_lightA_green_pin, OUTPUT);  
9  }
```



Tilstand 1

- Under kommentaren for tilstand 1 tilføjes kode, der:
 - Tænder den røde lysdiode – `digitalWrite(yyy, HIGH);`
 - Venter 7 sekunder – `delay(7000);`
- Prøv at uploade koden til Arduino; det røde lys tænder

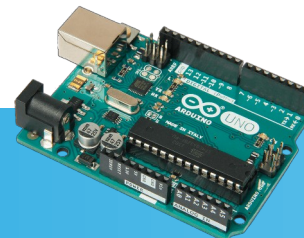
```
11 void loop() {  
12     // 1. tilstand: rødt lys  
13     digitalWrite(traffic_lightA_red_pin, HIGH); // Tænd rødt  
14     delay(7000);  
15  
16     // 2. tilstand: rødt + gult lys
```



Tilstand 2

- I tilstand 2 skal vi have tændt den gule lysdiode
- Vi venter 1 sekund
- NB: Den røde lysdiode er stadig tændt fra tilstand 1

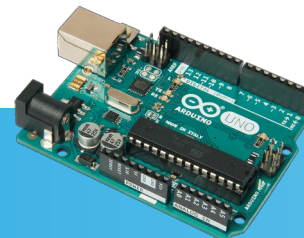
```
16 // 2. tilstand: rødt + gult lys
17 digitalWrite(traffic_lightA_yellow_pin, HIGH); // Tænd gult
18 delay(1000);
19
20 // 3. tilstand: grønt lys
```



Tilstand 3

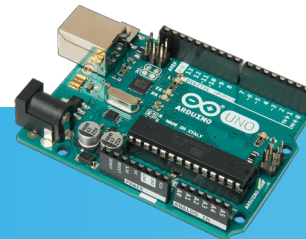
- I tilstand 3 skal vi have grønt lys
- Vi skal også have slukket rødt og gult
 - `digitalWrite(xxx, LOW);`
- Tilstanden skal vare 5 sekunder

```
20 // 3. tilstand: grønt lys
21 digitalWrite(traffic_lightA_red_pin, LOW); // Sluk rødt
22 digitalWrite(traffic_lightA_yellow_pin, LOW); // Sluk gult
23 digitalWrite(traffic_lightA_green_pin, HIGH); // Tænd grønt
24 delay(5000);
25
26 // 4. tilstand: gult lys
```



Tilstand 4

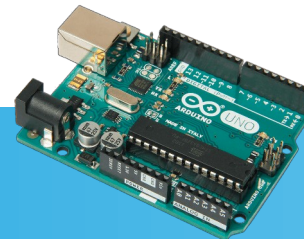
- Vi skal have gult lys
- Og huske at slukke det grønne
- Skriv selv koden
- Upload til Arduino



Huskede du?

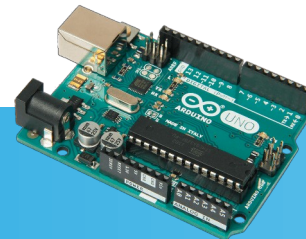
- Huskede du at slukke det gule lys?
 - Det kan enten gøres efter delay i 4. tilstand
 - Eller gøres i toppen (1. tilstand)
 - NB: Der sker ikke noget ved at slukke den gule LED når der allerede er slukket

```
26 // 4. tilstand: gult lys
27 digitalWrite(traffic_lightA_green_pin, LOW); // Sluk grønt
28 digitalWrite(traffic_lightA_yellow_pin, HIGH); // Tænd gult
29 delay(1000);
30 digitalWrite(traffic_lightA_yellow_pin, LOW); // Sluk gult
31 }
```



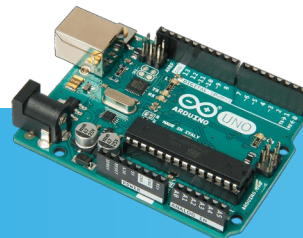
Anden retning

- Vi tilføjer nu et lyssignal for den anden retning
 - LyssignalB GND til Breadboard GND
 - LyssignalB R til Arduino pin 5
 - LyssignalB Y til Arduino pin 6
 - LyssignalB G til Arduino pin 7



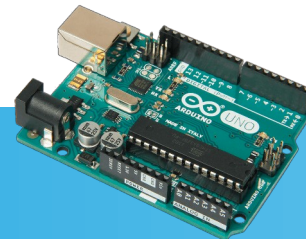
Initialisering

- Definér nye konstanter for de 3 nye pins
 - Kald dem `traffic_lightB_*_pin`
- Tilføj initialisering i `setup`-funktionen



Tilstande

- Der tilføjes 4 nye tilstande B1, B2, B3, B4
 - B1 = rødt lys for lyssignal B, osv.
- De 4 nye tilstande skal flettes ind i den eksisterende kode, men hvor?
 - Hint: der må kun være gult eller grønt i én retning ad gangen
- Husk at der er et delay, der skal tilpasses
- Upload sketch'en til Arduino og se om det virker

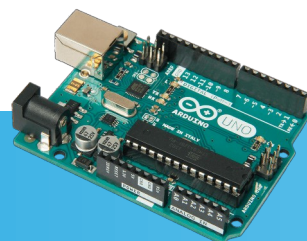


Kodestumper

```
4  const byte traffic_lightB_red_pin = 5;  
5  const byte traffic_lightB_yellow_pin = 6;  
6  const byte traffic_lightB_green_pin = 7;
```

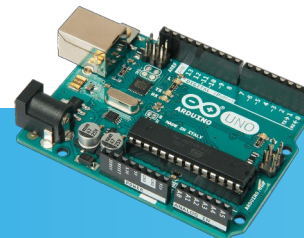
```
13  pinMode(traffic_lightB_red_pin, OUTPUT);  
14  pinMode(traffic_lightB_yellow_pin, OUTPUT);  
15  pinMode(traffic_lightB_green_pin, OUTPUT);
```

```
22  // B2. rød + gul  
23  digitalWrite(traffic_lightB_red_pin, HIGH); // Tænd rødt  
24  digitalWrite(traffic_lightB_yellow_pin, HIGH); // Tænd gult  
25  delay(1000);  
26  
27  // B3. grøn  
28  digitalWrite(traffic_lightB_red_pin, LOW); // Sluk rødt  
29  digitalWrite(traffic_lightB_yellow_pin, LOW); // Sluk gult  
30  digitalWrite(traffic_lightB_green_pin, HIGH); // Tænd grønt  
31  delay(5000);
```



Fodgængersignal pins

- Vi tilføjer nu et fodgængersignal til retning B
 - Fodgængersignal GND til Breadboard GND
 - Fodgængersignal R til Arduino pin 8
 - Fodgængersignal G til Arduino pin 9

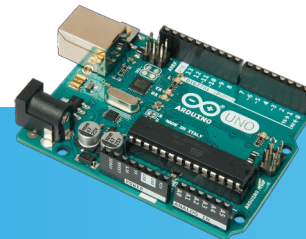


Initialisering af pins

- Opret konstanter til de to nye pins (lightF for fodgænger)
- Konfigurer som udgange i setup

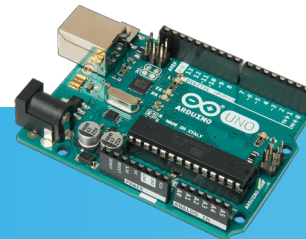
```
7  const byte traffic_lightF_red_pin = 8;  
8  const byte traffic_lightF_green_pin = 9;
```

```
17  pinMode(traffic_lightF_red_pin, OUTPUT);  
18  pinMode(traffic_lightF_green_pin, OUTPUT);
```



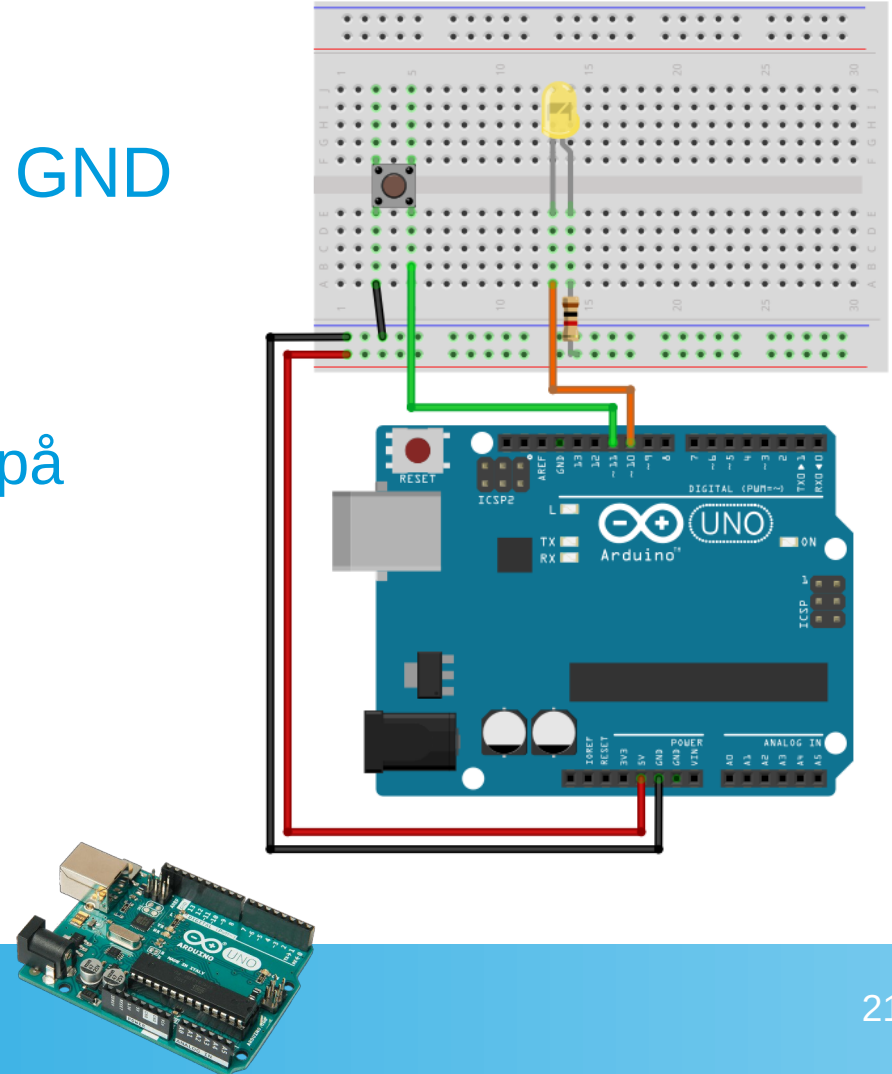
Fodgængersignal logik

- Fodgængersignalet skal kun være grønt når signal B er grønt
- Husk også at fodgængersignalet skal være rødt når sketch'en starter (fx i setup-funktionen)



Trykknop og lysdiode

- På breadboard tilføjes:
- En knap der forbinder pin 11 til GND
- En lysdiode med formodstand på $1k\Omega$
 - Formodstand forbindes fra +5V på breadboard til lysdiodens anode (lange ben)
 - Lysdiodens katode (korte ben) forbindes til Arduino pin 10.



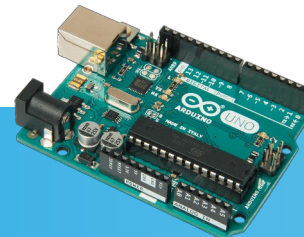
Konfigurere input og output

- Først defineres konstanter for de to nye pins:

```
9  const byte traffic_lightF_indicator_pin = 10;  
10 const byte traffic_lightF_button_pin = 11;
```

- Lysdioden er forbundet så den tændes ved at sætte pin 10 lav. Vi sætter derfor pin 10 høj for at slukke den
- Fordi knappen er forbundet til GND kan vi udnytte at mikroprocessoren har en intern pull-up modstand

```
22  pinMode(traffic_lightF_indicator_pin, OUTPUT);  
23  digitalWrite(traffic_lightF_indicator_pin, HIGH); // Sluk indikator  
24  pinMode(traffic_lightF_button_pin, INPUT_PULLUP);
```

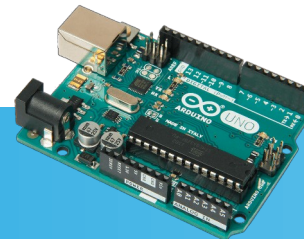


Simpel løsning

- Vi starter med en simpel løsning: Det grønne fodgængersignal tænder hvis knappen et tryk ned på det tidspunkt hvor signal B skifter til grønt:

```
36   if (digitalRead(traffic_lightF_button_pin) == LOW) {  
37       // Fodgænger grønt  
38       digitalWrite(traffic_lightF_red_pin, LOW); // Sluk rødt  
39       digitalWrite(traffic_lightF_green_pin, HIGH); // Tænd grønt  
40   }
```

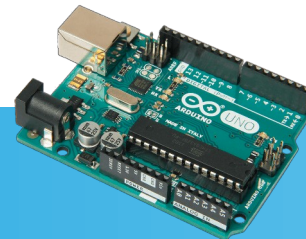
- Prøv at uploade koden til Arduino
 - Hvis knappen ikke trykkes ned forbliver fodgængersignalet rødt
 - Hvis knappen holdes nede skifter den til grønt med signal B



Den *rigtige* løsning

- Vi ændrer funktionen, så man ved at trykke på knappen tænder indikator-lysdioden, og hvis denne er tændt skifter fodgængersignalet *næste* gang signal B bliver grønt
- Start med at lave en global variabel til at huske om fodgængersignalet er aktiveret (før setup)

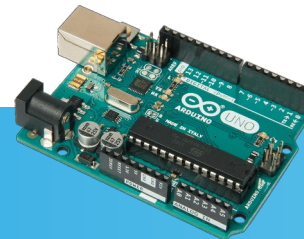
```
18  bool lightF_activated = false;|
19
20  void setup() {
```



Ændre if-sætningen

- Vi udskifter betingelsen for at tænde det grønne fodgængersignal, og samtidig slukker vi her indikatoren:

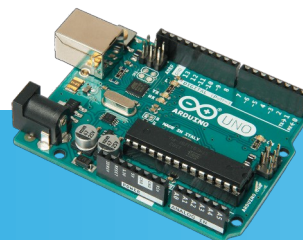
```
44 | if (lightF_activated) {  
45 |     lightF_activated = false;  
46 |     digitalWrite(traffic_lightF_indicator_pin, HIGH); // Sluk indikator  
47 |     // Fodgænger grønt  
48 |     digitalWrite(traffic_lightF_red_pin, LOW); // Sluk rødt  
49 |     digitalWrite(traffic_lightF_green_pin, HIGH); // Tænd grønt  
50 | }
```



Vores egen delay-funktion

- Nu laver vi vores egen delay-funktion, som også checker tilstanden af knappen (indsæt mellem setup og loop):

```
35 void delay_poll_button(unsigned long duration_msec) {  
36     unsigned long expire = millis() + duration_msec;  
37     while (millis() < expire) {  
38         if (digitalRead(traffic_lightF_button_pin) == LOW) {  
39             lightF_activated = true;  
40             digitalWrite(traffic_lightF_indicator_pin, LOW); // Tænd indikator  
41         }  
42     }  
43 }
```

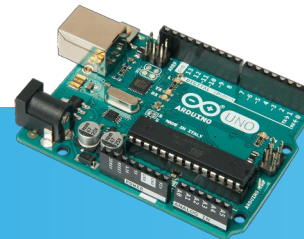


Udskift delay-funktionen

- Nu udskiftes alle kald til delay-funktionen i loop med vores egen delay_poll_button, fx:

```
81 // 2. tilstand: rødt + gult lys
82 digitalWrite(traffic_lightA_yellow_pin, HIGH); // Tænd gult
83 delay_poll_button(1000);
84
85 // 3. tilstand: grønt lys
86 digitalWrite(traffic_lightA_red_pin, LOW); // Sluk rødt
87 digitalWrite(traffic_lightA_yellow_pin, LOW); // Sluk gult
88 digitalWrite(traffic_lightA_green_pin, HIGH); // Tænd grønt
89 delay_poll_button(5000);
90
```

- Upload til Arduino og prøv!



Ekstraopgaver

- Tilføj et potentiometer til analog indgang A0
- Den analoge værdi er 0-1023, den kaldes X
- Programmér så det grønne lys i retning A er tændt $3000 + 4 * A$ millisekunder
- Det grønne lys i retning B skal være tændt i $7000 - 4 * A$ millisekunder
- NB: Det betyder at begge retninger kan indstilles til at være grønt i intervallet ca. 3-7 sekunder

