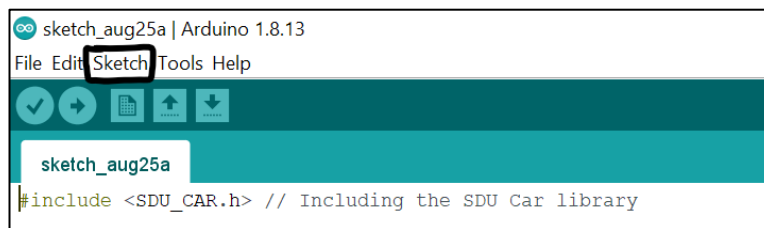


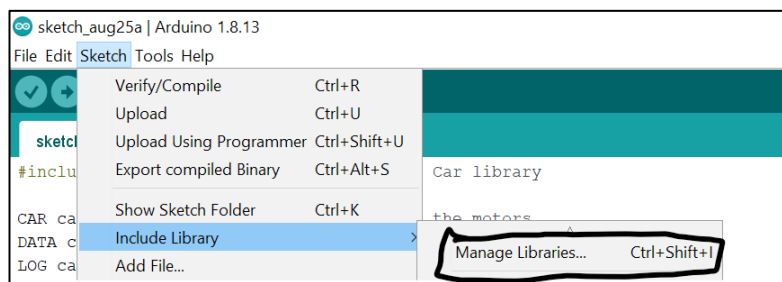
Brug af SDU CAR Biblioteket

Inkluder SDU_CAR library:

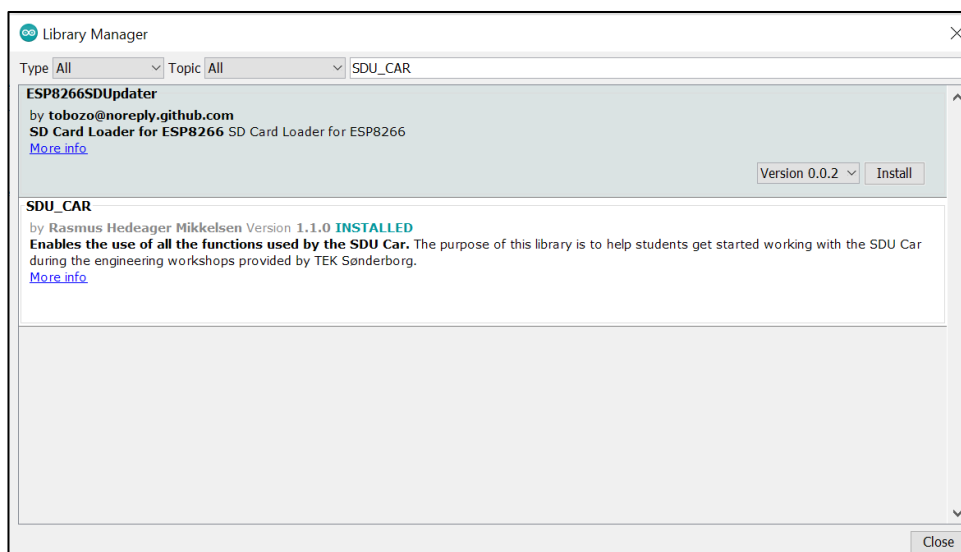
Klik på sketch:



Klik på Manage libraries:

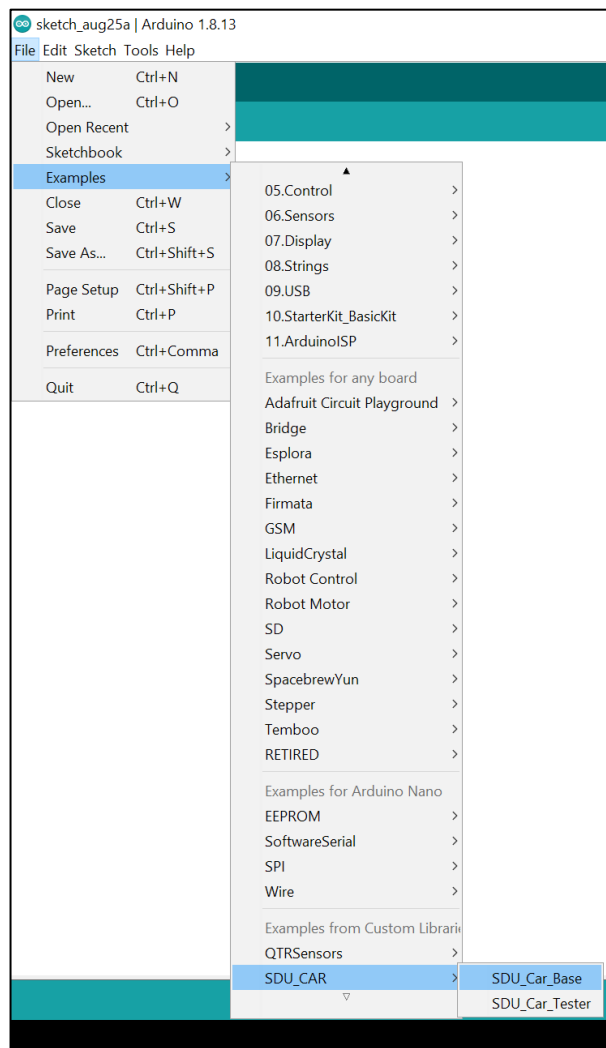


Søg efter SDU_CAR og tryk installer:



Åben en opstartskode:

Dette sikrer at biblioteket bliver indlæst korrekt. Gå nu ind i filer, examples og rul helt ned i bunden hvor du finder Examples from custom Library og vælg SDU_car_Base.



Hvordan bruger man koden?

Indstil hastigheden på hvert hjul

- Man kan indstille hastigheden fra -100 til 100
- Til venstre for komma er det venstre hjul
- Til højre for komma er det højre hjul

Eksempel på brug:

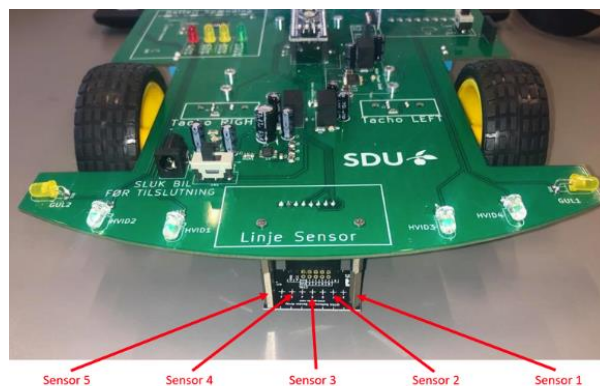
```
car.setCarSpeed(75, -75);
```

Dette sætter hastigheden på det hjulet til venstre til 75 og det højre til -75 (Bemærk, motor kører kun når batteri kontakten står på "ON")

Aflæs linje sensor

Når man kigger på bilen hvor den vil køre væk fra dig, hvis den kørte fremad, er numrene på sensoren i læse rækkefølge. Det vil sige helt ude til venstre er sensor 1 så 2 osv.

På billedet under kører bilen mod dig og derfor er det omvendt rækkefølge:



For så at aflæse sensoren og gemme den i en variabel kan følgende kode bruges:

```
cardata.readLineSensor();  
int sensor4 = cardata.getLineSensor(4);
```

Her bliver sensor 4 aflæst, hvis man vil aflæse sensor 1 ændre du blot 4-tallet til et 1-tal. Du vil modtage en værdi der skal kalibreres så der kan kendes forskel på hvidt gulv og sort baggrund.

Kalibrering af sensor:

```
1 #include <SDU_CAR.h>
2
3 // Sets up the SDU Car library.
4 CAR car;
5 DATA cardata;
6 LOG carlog;
7
8
9 void setup() {
10     Serial.begin(9600); // Communication with the computer.
11
12     cardata.begin(); // Enables reading of sensors!
13 }
14
15
16 void loop() {
17
18     // Reads the line follower sensor.
19     cardata.readLineSensor();
20
21     int sensor1 = cardata.getLineSensor(1); // Gets the value from sensor 1.
22
23     Serial.println(sensor1);
24
25 }
```