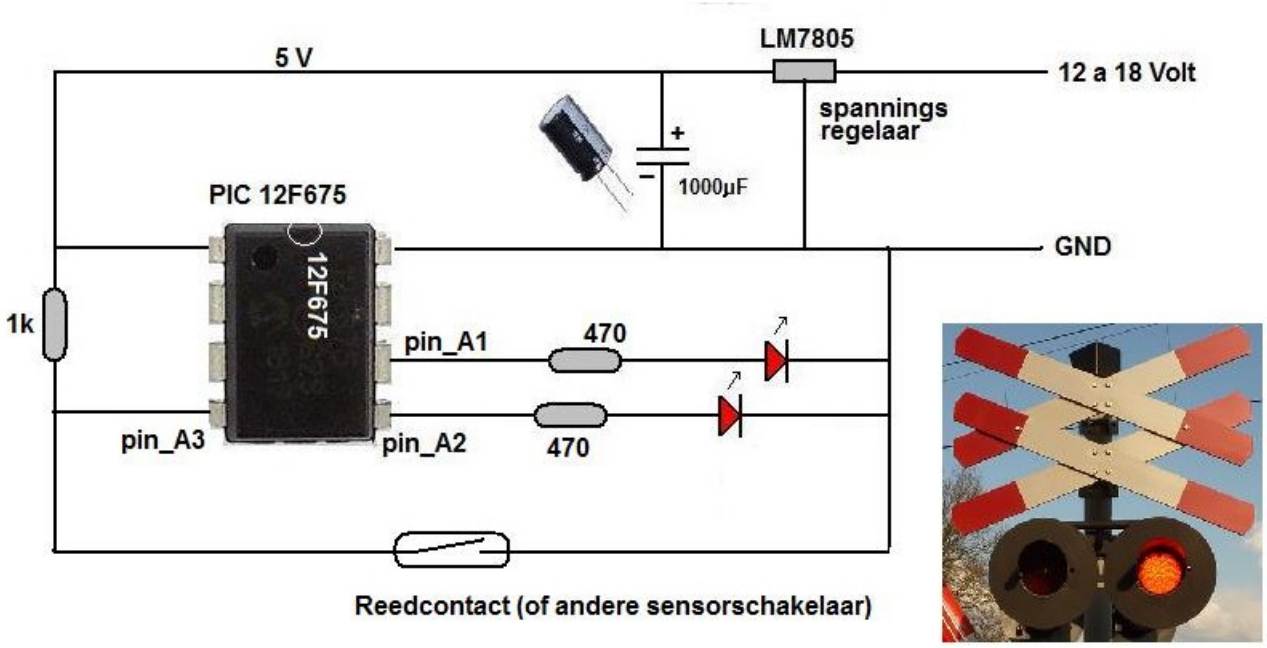
Zelf PIC's programmeren

Deel-3 Spoorwegovergang (Railroad crossing)

In dit voorbeeld wordt er met de 12F675 een knippersignaal gemaakt dat 10 aan/uit gaat.
Je zou dit kunnen gebruiken voor een overwegbeveiliging.
Als de trein over een contactpunt rijdt gaan de rode lampen om beurten aan en uit.

Je kunt het programma hiervoor op verschillende manieren maken, eenvoudig tot wat meer gecompliceerd.

We zullen eenvoudig beginnen
Rijdt de trein over een contactpunt, dan gaan de 2 rode LEDs om beurten 10 keer aan en uit. Daarna doven beide rode LEDs weer.



De LEDs staan op pin_A1 en pin_A2.
Het contact wordt hier gemaakt met een reedcontact, maar andere contacten zijn natuurlijk ook mogelijk.
Als er geen trein is is pin_A3 hoog (5 Volt).
Rijdt er een trein over het contact dan sluit het reedcontact en komt pin_A3 laag op 0 Volt.
De LED's gaan daardoor (10 keer) knipperen.

De JALEdit code is

```
include 12f675

pragma target clock    4_000_000

pragma target OSC      INTOSC_NOCLKOUT
pragma target WDT      disabled
pragma target MCLR     internal
pragma target BROWNOUT disabled

enable_digital_io()

include delay                          -- het laden van de delay's

pin_A1_direction = output     alias Rood1 is pin_A1
pin_A2_direction = output     alias Rood2 is pin_A2
pin_A3_direction = input      alias sensor is pin_A3

Var Byte teller
Rood1=low
Rood2=low
```

forever loop

While sensor==high loop end loop

Teller= 10
** Repeat**
** Rood1=high Rood2=low Delay_1s(1)**
** Rood1=low Rood2=high Delay_1s(1)**
** Teller=Teller-1**
** Until Teller==0**
Rood1=low Rood2=low

end loop

De codes nader toegelicht

include 12f675
pragma target clock 4_000_000
pragma target OSC INTOSC_NOCLKOUT
pragma target WDT disabled
pragma target MCLR internal
pragma target BROWNOUT disabled
enable_digital_io()
include delay
zijn weer de standaard codes bij dit gebruik van de 12F675.

forever loop, het begin van de eeuwigdurende loop of lus.

While sensor==high loop end loop, zolang de sensor hoog is (geen trein) blijft het programma in een 2e loop zitten. Er gebeurt hier niets anders dan wachten op een trein.

Teller= 10, er is inmiddels een een trein aangekomen, we gaan verder door de teller op 10 te zetten.

Repeat, hier start weer een lus, de repeat-lus.

Rood1=high Rood2=low Delay_1s(1), LED rood1 gaat aan en rood2 gaat uit, vervolgens wachten we 1 seconde.

Rood1=low Rood2=high Delay_1s(1), LED rood1 gaat uit en rood2 gaat aan, vervolgens wachten we weer 1 seconde.

Teller=Teller-1, de teller wordt met 1 verminderd.

Until Teller==0, als de teller op 0 gekomen is verlaten we de repeat-lus
(opmerking: bij een vergelijking wordt 2 keer het = gebruikt!)

Rood1=low Rood2=low, beide LEDs gaan nu uit.

end loop, weer terug naar het begin van de hoofd lus of loop.

Het nadeel van deze uitvoering is dat er maar een start-sensor in het geheel zit.
Er wordt nog niet gekeken of de trein op tijd de overweg heeft gepasseerd.
In deel ahob-overweg komt deze schakeling aan bod.

Resume

The train drives over a reed contact in this circuit.After this the two LEDs start to flash alternately.Both LEDs flash 10 times in total, after which they go out again.This setup does not look at whether the trainhas actually completely passed the transition.The first code is the program are the fixed standard codes.The program then waits until the sensor goes low, after which the repeat loop starts.