

# Intelligent kruispunt

## Probleemstelling

Iedereen kent het probleem wel: we dienen te stoppen voor een verkeerslicht en uit de zijstraten is geen verkeer te bespeuren. De wachttijden aan kruispunten zijn vaak langer dan nodig. De verkeerslichten aan het huidige en volgende kruispunt op dezelfde baan zijn niet op elkaar afgestemd. Hierdoor moet men steeds terug stoppen. Dit zijn factoren die de doorstroming verlagen en de mogelijkheid tot filevorming vergroten.

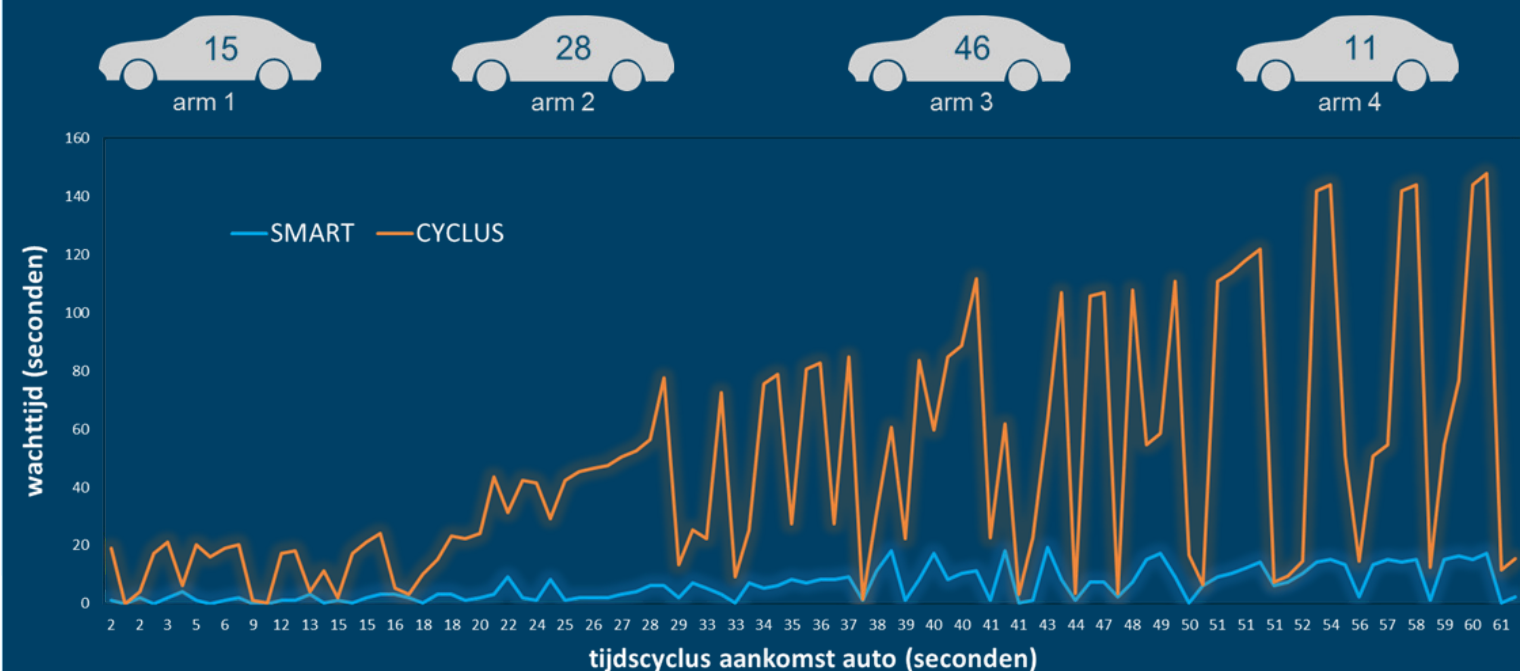
De uitdaging van dit project is om zelf een algoritme te genereren waarmee dit probleem aangepakt kan worden. Ook zullen verschillende systemen tegen elkaar worden afgewogen om bijvoorbeeld de drukte in elke arm van het kruispunt te detecteren.

We hebben getracht om de opstelling zo realistisch mogelijk te maken, zodat dit systeem ook in de realiteit kan worden gebruikt. Niet een Arduino, zoals in de maquette voorkomt, maar wel een andere soort microcontroller zal in de praktijk worden toegepast.

Dit project zal laten zien dat er slechts een kleine ingreep nodig is om van een 'dom' kruispunt een 'intelligent' kruispunt te maken.

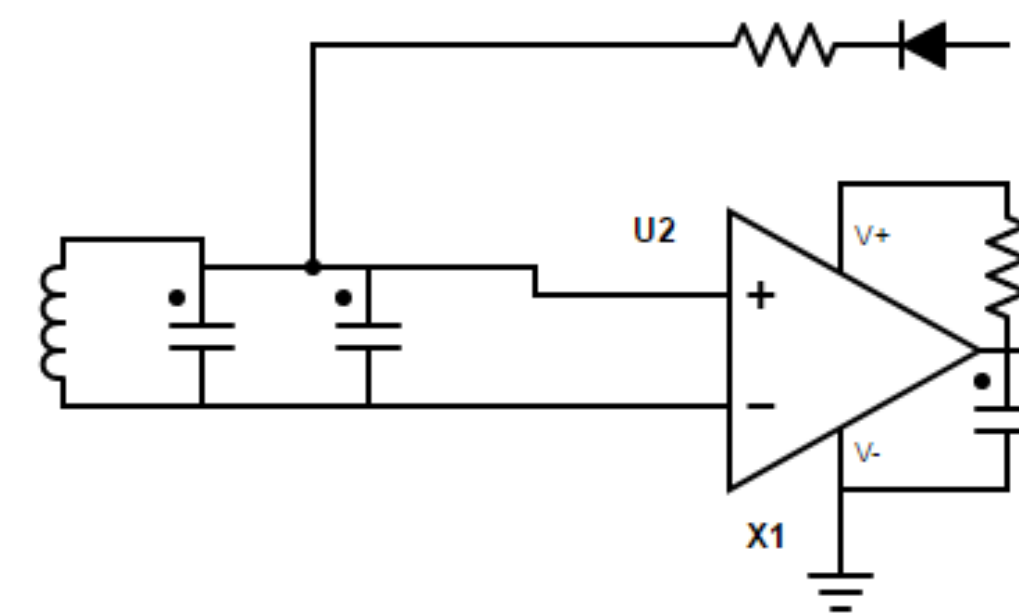
## Besluit

Voor de uiteindelijke analyse en de optimalisatie van ons project hebben we een optimalisatie software geschreven die twee keer eenzelfde bestand inleest. De grafiek hieronder is een voorstelling van de wachttijd van de auto ten opzichte van wanneer hij aangekomen is. De oranje curve in de grafiek toont de wachttijd voor onze ingreep en de blauwe curve toont de wachttijd na de ingreep. Het mag duidelijk zijn dat het verschil in wachttijd tussen een statisch kruispunt en een intelligent kruispunt aanzienlijk is.



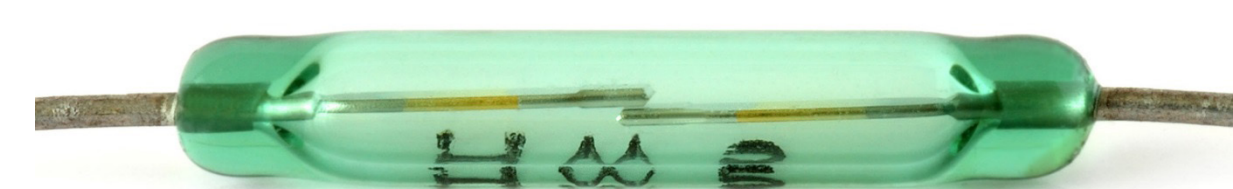
## Sensoren

We hebben eerst zelf een inductiesensor gemaakt. Deze bestaat uit een LC-kring, comparator en enkele weerstanden. De Arduino stuurt een pulssignaal naar de LC-kring op een frequentie. Die wordt bepaald door de resonante frequentie van de LC-kring. Wanneer er metaal in de buurt komt, zal de inductie van de spoel variëren. De resonante frequentie varieert ook en wordt gedetecteerd door de comparator.



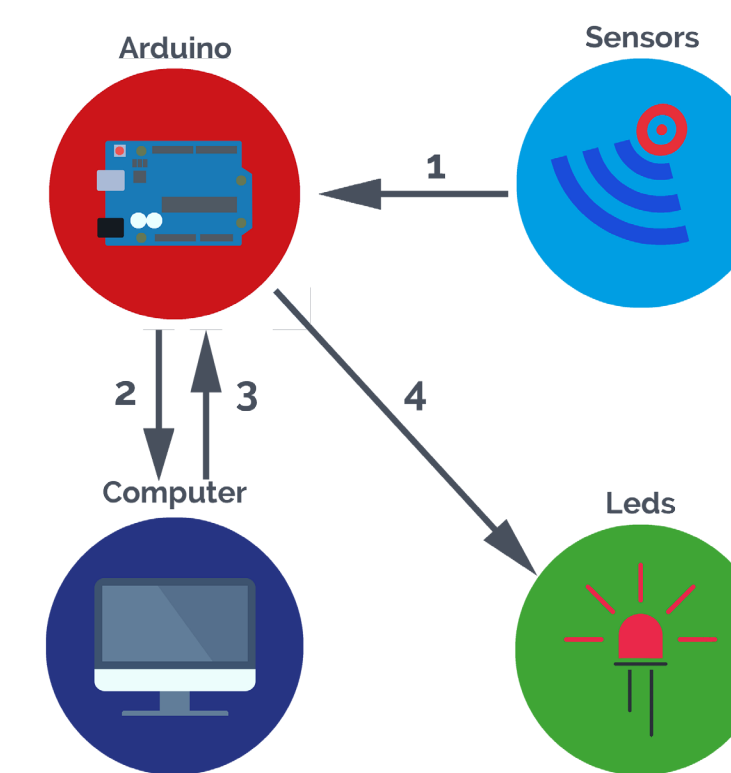
$$F_R = \frac{1}{2\pi\sqrt{LC}}$$

Door de grootte bleek uiteindelijk dat deze sensor niet geschikt is. Daarom wordt er nu gebruik gemaakt van reed switches. Dit zijn schakelaars die open en toe gaan, afhankelijk of er een magnetisch veld in de buurt is. In het glazen buisje bevindt zich een inert gas. Wanneer het glas breekt wordt de reed switch onbruikbaar.



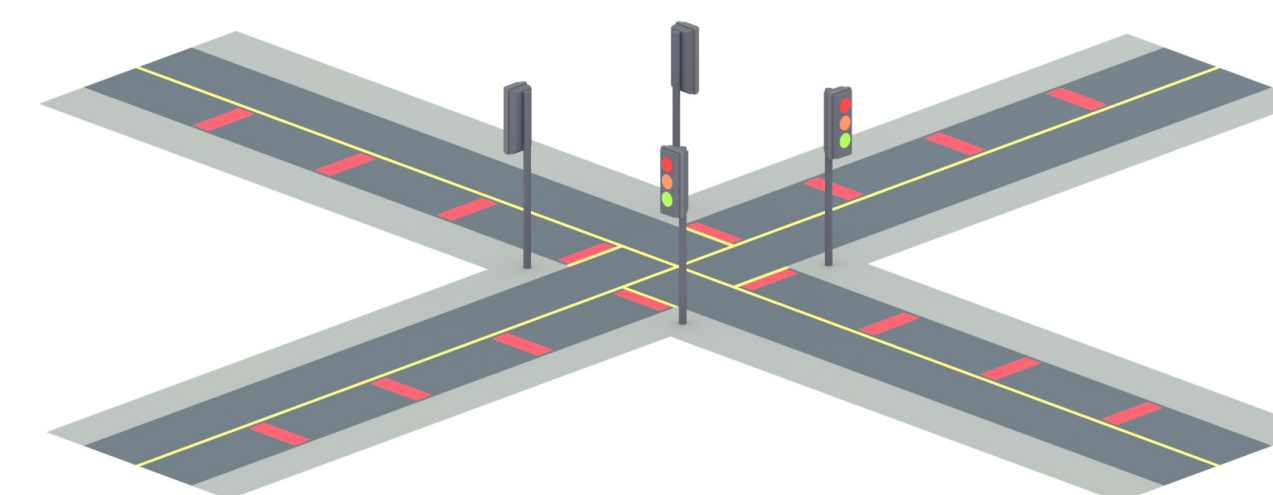
## Communicatie

Het algoritme wordt op de computer uitgevoerd en de Arduino zal enkel worden gebruikt om de sensors en lichten aan te sturen. Een wederzijdse communicatie tussen de Arduino en computer is noodzakelijk. Om deze seriële communicatie tussen de Java-applicatie en de Arduino tot stand te brengen, maakten we gebruik van de library RXTX. Vanaf dat deze communicatie mogelijk was, was er alleen nog een afgesproken protocol nodig om de doelen te bereiken.



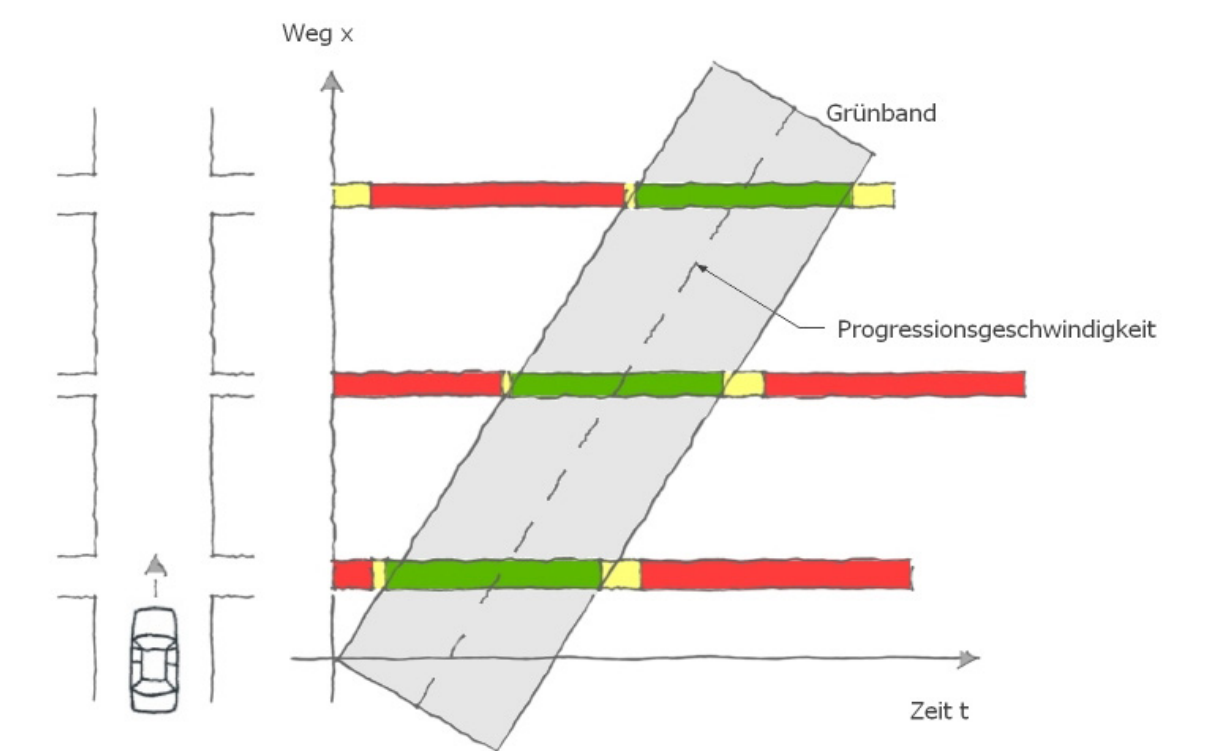
## Algoritme

We zochten naar een algoritme dat ervoor kan zorgen dat een kruispunt veel efficiënter kan worden gebruikt. We bespreken een kruispunt met 4 armen. Elke arm heeft 4 sensoren verspreid over de lengte. Deze sensoren zullen de auto's tellen die op het kruispunt rijden. Het algoritme kan dan het kruispunt opdelen in zones en daarmee bepalen hoe druk het is op elke arm. Het verkeerslicht blijft dan op groen staan voor twee tegenoverliggende armen totdat er een aantal auto's het kruispunt verlaten.

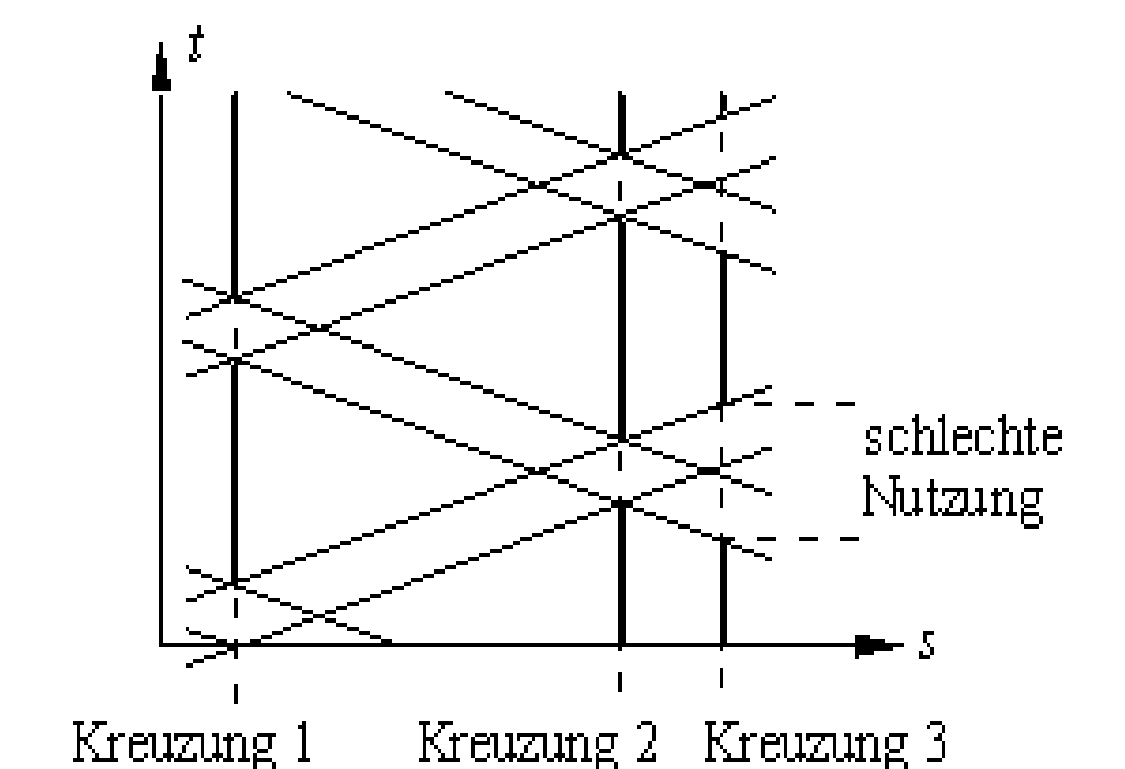


## Groene golf

De groene golf zorgt ervoor dat de verkeerslichten op elkaar zijn afgestemd. Dit voorkomt dat men steeds bij elk kruispunt dat men tegenkomt moet stoppen. Bij een vooraf opgelegde snelheid zal de weggebruiker gegarandeerd steeds groen licht krijgen. De groene golven, zoals ze nu bestaan, zijn statisch. Ze zijn niet afgestemd op de drukte op een bepaald moment. Het doel is om een dynamische golf te maken. Dit kan door de drukte in de zijstraten op te meten.



De berekening kan het beste op een grafische manier gebeuren. De breedte van de band is gebaseerd op het licht dat het kortst groen is.



De groene golf kan maar in één richting worden toegepast als de kruispunten niet even ver van elkaar liggen.