

Eindverslag project 2

Multiflexmeter



Naam teamleden: Floor Helmendach, Wouter Schets, Jelmer Appelo, Jens Theunissen

Groepsnummer: Groep 12

Klassen: T3TG en V3TK

Naam opdrachtgever: Dhr. Jan van Kranenburg & dhr. Jos Goossen

Bedrijf: Waterschap Scheldestromen

Inleverdatum: 16 februari

Inhoudsopgave

Samenvatting.....	3
Voorwoord	4
inleiding	5
Programma van eisen.....	6
Het ontwerpproces	7
Conclusie	Fout! Bladwijzer niet gedefinieerd.
Aanbevelingen	14
Nawoord	15
Bronnenlijst	16
Bijlagen.....	17

Samenvatting

Deze opdracht hebben we gekregen van het Waterschap die hun ouderwetse waterpeilmeters wilden vervangen met een elektronische versie die niet alleen het waterpeil meet, maar ook nog een aantal andere dingen. We keken welke dingen de zogenaamde multiflexmeter nog miste en kozen er samen met de opdrachtgever een uit. Het werd de saliniteitssensor, of zoutgehaltemeter. We zochten uit hoe we dit konden meten, en kwamen er achter dat het handig was om te kijken naar de weerstand van het water. We deden een aantal experimenten om te kijken hoe dit in zijn werking ging en wat problemen zouden zijn als we dit zouden willen toepassen in de meter. De grootste problemen werden gevormd door het feit dat elektrolyse plaatsvond en dat we rekening moesten houden met een aantal factoren en dus het experiment opnieuw moesten doen met andere elektrodes.

Toen zochten we uit hoe we wisselstroom konden gebruiken met de Arduino en hoe we Ampère konden meten met de Arduino. Wisselstroom was nogal een probleempje en dat hebben we nog niet opgelost. Ampère kan worden gemeten met een bepaald circuit en een formule. Deze kun je vinden in de bijlagen.

Na ons experiment richtten we ons op het programma dat bij deze sensor paste. Dit bleek vrij ingewikkeld omdat het wisselstroom moest gebruiken, meerdere formules nodig had en Volts moest kunnen weten. Bij de tijd van dit verslag was ons programma nog niet af en we weten daarom ook niet of het wel af gaat komen. We denken wel dat met meer tijd en vaardigheid dit een goede manier is om zoutgehalte te meten en we hebben er vertrouwen in dat het Waterschap kan voortbouwen op ons werk.

Voorwoord

Wij hebben van het Waterschap Scheldestromen de opdracht gekregen om een sensor te bedenken voor de multiflexmeter. Deze meter is bedoeld om van een grote afstand metingen te nemen over het water. Zo hoef je niet eerst naar een sloot te gaan maar kan je de metingen vanaf kantoor meten. We zijn door Waterschap Scheldestromen ingeleid door dhr. Jan van Kranenburg en dhr. Jos Goossen. Hij legde de opdracht uit en vertelde ons welke sensor ze nog nodig hadden. Ons leek dit een leuke uitdaging.

Inleiding

De aanleiding van dit project is dat het checken van het waterpeil en andere toestanden in sloten te veel tijd in beslag neemt. De methode die nu gebruikt wordt is dat medewerkers van Waterschap Scheldestromen zelf eens in de paar weken naar een bepaalde sloot gaan en zelf de saliniteit, de hoeveelheid neerslag en de aanslibbing meten. Wij waren gevraagd om dit op te lossen. Hiervoor probeerden wij werkende sensoren te ontwerpen die toegevoegd kunnen worden in de multiflexmeter. Een van de eisen waren dat onze meters een mediaangemiddelde van 21 metingen per dag zou moeten aangeven. Iedereen mag de huidige opzet en eventuele andere opzetten gebruiken en/of verbeteren in Arduino.

Programma van eisen

Ook al is dit een aardig vrij project, er zijn natuurlijk een aantal eisen. Een aantal kregen we bij het waterschap te weten en sommige moesten we er zelf bij verzinnen. Dit zijn de eisen:

Makkelijk in gebruik.

Max. 150 euro.

In Arduino gemaakt.

Niet groter dan het doosje waar het in moet komen. (20x10x10)

Het moet lang houdbaar zijn. (5 tot 10 jaar)

Meerdere sensoren in 1 apparaat.

Je moet het kunnen aflezen terwijl je op kantoor zit.

Het ontwerpproces

We begonnen dit project met onderzoek naar verschillende aspecten van de multiflexmeter zoals de kosten en gebruikte sensoren. Ook keken we welke sensoren de meter miste en hoe we dit konden oplossen. We besloten samen met de opdrachtgever om verder te gaan met de saliniteitssensor. We deden onderzoek naar hoe we dit het best konden meten en besloten dit te doen door te meten hoe goed het water geleid, omdat de weerstand direct proportioneel is aan het zoutgehalte. We begonnen met een experiment om te kijken hoe dit in zijn werking ging en hoe we een formule konden maken die het aantal Ampère omrekenen naar het zoutgehalte. Het experiment ging niet helemaal goed omdat er elektrolyse plaatsvond en dit beïnvloedde de metingen nogal veel. Ook waren we de elektrodes die we gebruikten te groot om in de multiflexmeter te gebruiken. Het oppervlakte heeft heel veel invloed op de metingen en daarom moesten we het experiment dus in de toekomst opnieuw doen.

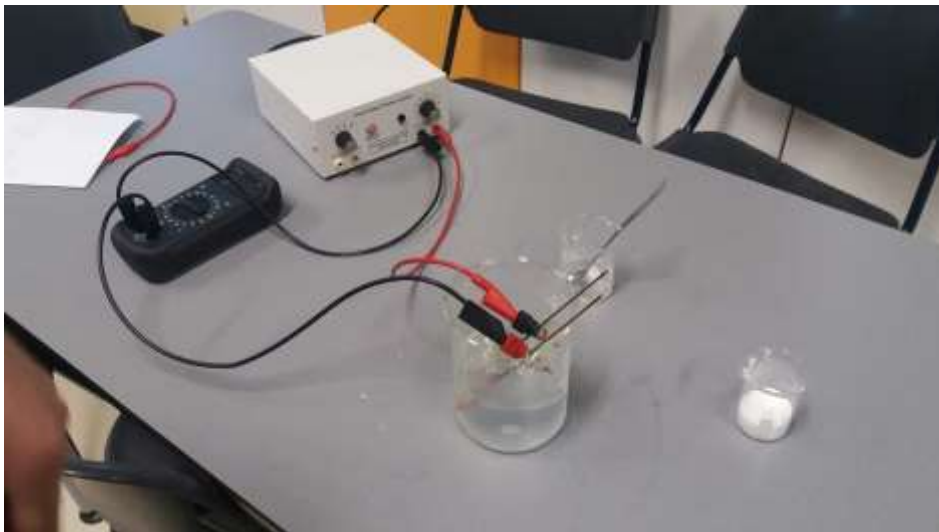


Foto: De opstelling van het 1^e experiment.

We zochten een paar geschiktere elektrodes en maakten ze met een geschikte afstand van elkaar vast. We zorgden ook dat het waterdicht was. We herhaalden het experiment zoals we het eerder hadden gedaan, maar deze keer gebruikten we wisselstroom om elektrolyse te voorkomen. Het laatste wat toen nodig was was een manier om Ampère te meten met de Arduino. Onze docenten hielpen ons hierbij en legden uit dat dit kon door een extra weerstand toe te voegen en volt te meten tussen het water en de extra weerstand. Toen begonnen we met een poging om een programma hiervoor te schrijven, wat vrij ingewikkeld was. We moesten wisselstroom gebruiken, meerdere formules en een manier om Volts te meten. Toen dit eindverslag werd geschreven was het programma nog niet af en we kunnen niks zeggen over of het überhaupt gaat lukken.

Natuulijk hebben we ook een werkend prototype gemaakt van de Multiflexmeter. Hierin zat een manier om het waterpeil en de temperatuur te meten. Maar dit vind u in de sketch

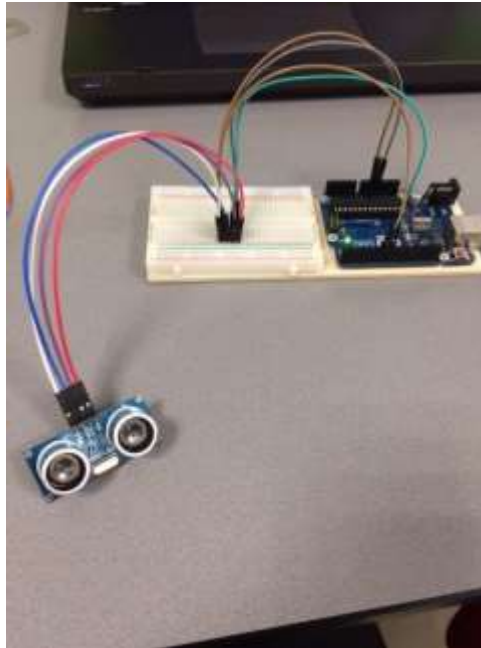
Sketch

Eén van de opdrachten was om een werkend prototype te maken met Arduino van de multiflexmeter. We hebben ons gericht op het waterpeil, de watertemperatuur en saliniteit.

We zijn begonnen door een ultrasone sensor in de Arduino te verwerken. Met deze sensor kan je het waterpeil meten. Hieronder staat een foto met de sketch. Zodat dit wellicht gebruikt kan worden.

```
#include <Dhcp.h>
#include <Dns.h>
#include <Ethernet.h>
#include <EthernetClient.h>
#include <EthernetServer.h>
#include <EthernetUdp.h>

#define echoPin 7 // Echo Pin
#define trigPin 8 // Trigger Pin
#define LEDPin 13 // Onboard LED
```



```
int maximumRange = 200; // Maximum range needed
int minimumRange = 0; // Minimum range needed
long duration, distance; // Duration used to calculate distance
```

```
void setup() {
  Serial.begin (9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(LEDPin, OUTPUT); // Use LED indicator (if required)
}
```

```
#include <RunningMedian.h>
```



```

void loop() {
  /* The following trigPin/echoPin cycle is used to determine the
  distance of the nearest object by bouncing soundwaves off of it. */
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);

  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);

  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);

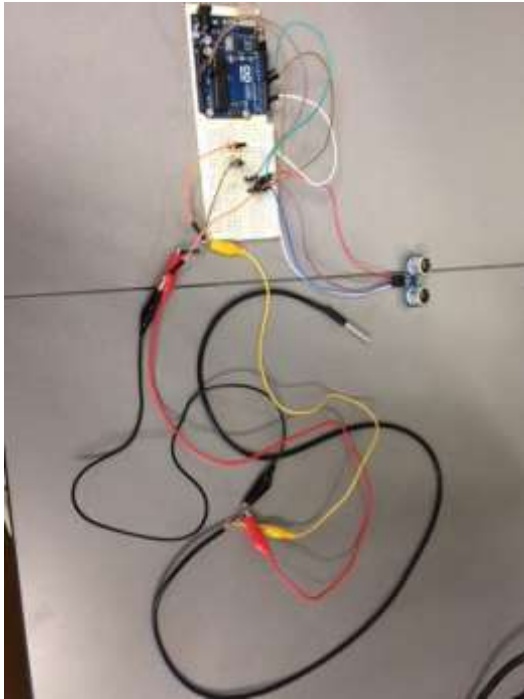
  //Calculate the distance (in cm) based on the speed of sound.
  distance = duration/58.2;

  if (distance >= maximumRange || distance <= minimumRange){
    /* Send a negative number to computer and Turn LED ON
    to indicate "out of range" */
    Serial.println("-1");
    digitalWrite(LEDPin, HIGH);
  }
  else {
    /* Send the distance to the computer using Serial protocol, and
    turn LED OFF to indicate successful reading. */
    Serial.println(distance);
    digitalWrite(LEDPin, LOW);
  }

  delay(2000)
}

```

Toen deze werkte zijn we verder gegaan om er ook een waterdichte thermometer aan toe te voegen. Hiermee kan je de watertemperatuur meten. We wouden ook de codes samenvoegen zodat het op 1 breadboard kan.



```
#include <OneWire.h>
```

```
#include <DallasTemperature.h>
```

```
#include <Dhcp.h>
```

```
#include <Dns.h>
```

```
#include <Ethernet.h>
```

```
#include <EthernetClient.h>
```

```
#include <EthernetServer.h>
```

```
#include <EthernetUdp.h>
```

```
#include <RunningMedian.h>
```

```
#define DS18B20#define echoPin 7 // Echo Pin
```

```
#define trigPin 8 // Trigger Pin
```

```
#define LEDPin 13 // Onboard LED
```

```
int maximumRange = 200; // Maximum range needed
```

```
int minimumRange = 0; // Minimum range needed
```

```
long duration, distance; // Duration used to calculate distance
```

```

OneWire ourWire(DS18B20);
DallasTemperature sensors(&ourWire);

void setup() {
  Serial.begin(9600);
  delay(1000);
  //start reading
  sensors.begin();

  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(LEDPin, OUTPUT); // Use LED indicator (if required)
}

void loop() {

  sensors.requestTemperatures();
  Serial.print(sensors.getTempCByIndex(0));
  Serial.println(" degrees C");

  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);

  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);

  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);

  //Calculate the distance (in cm) based on the speed of sound.
  distance = duration / 58.2;

  if (distance >= maximumRange || distance <= minimumRange) {
    /* Send a negative number to computer and Turn LED ON

```

```

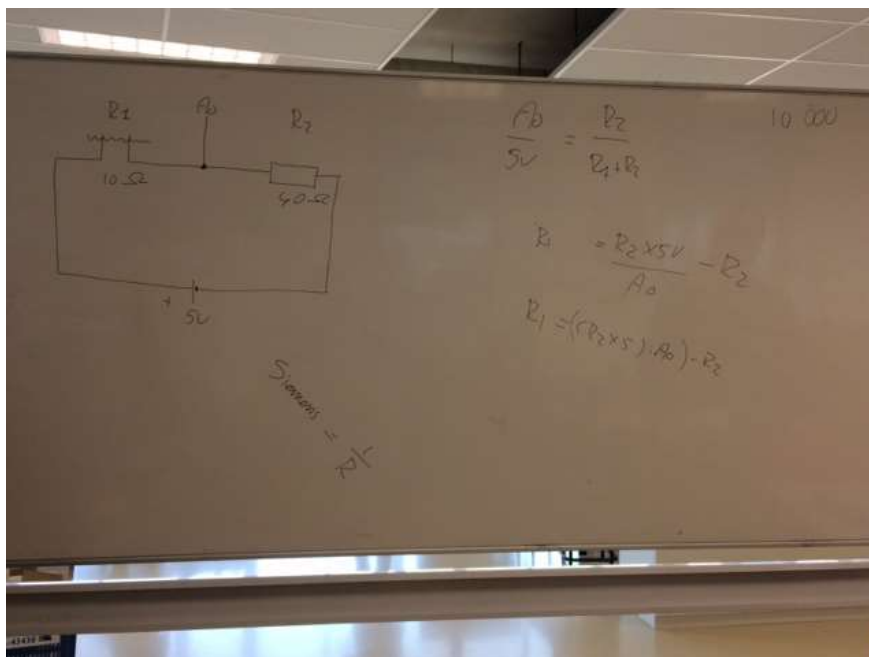
    to indicate "out of range" */
    Serial.println("-1");
    digitalWrite(LEDPin, HIGH);
}
else {
    /* Send the distance to the computer using Serial protocol, and
    turn LED OFF to indicate successful reading. */
    Serial.println(distance);
    digitalWrite(LEDPin, LOW);
}

//Delay 50ms before next reading.
delay(2000);

}

```

We hebben hierna nog een poging gewaagd om in deze Arduino een saliniteit sensor te plaatsen. We hiervoor proeven gedaan en een formule gemaakt. Hierna wilden we deze formule nog verwerken in Arduino en de sketch, maar hier hadden we helaas te weinig tijd voor. Hieronder vind u wel de foto met de formule en de schets die erbij hoort.



Conclusie

De opdracht was dus om de multiflexmeter te verbeteren en een nieuwe sensor te maken waar het waterschap iets mee kan. Onze sensor is nog niet af en het programma erbij ook niet. Daarentegen is ons idee realistisch en bruikbaar. Met iets meer tijd en moeite kan ons idee makkelijk worden toegepast in de meter. We vertrouwen erop dat het Waterschap op ons onderzoek kan voortbouwen en er wat mee kan uiteindelijk.

Voor de rest voldoet onze sensor aan het programma van eisen, zo heeft deze sensor weinig kosten en is hij, wanneer het programma af is, makkelijk in gebruik. Hij is ook niet erg groot. Wel is er het probleem dat deze in het water moet hangen, en dat de meter zelf boven het water hangt (anders kan hij niet het waterpeil meten). Hiervoor is een oplossing nodig, en dit kan eventueel ook worden gedaan door een pijpje uit de meter naar het water te laten lopen. Dit is iets waar de opdrachtgever nog even naar moet kijken.

Aanbevelingen

Er zijn veel dingen die nog moeten worden veranderd en verbeterd aan onze sensor. Het is bij lange na niet compleet. Door een tekort aan tijd en vaardigheid hebben we niet het complete programma kunnen schrijven voor de saliniteit en ook is onze elektrode gebrekkig. Corrosie heeft al veel invloed erop gehad al na 1 experiment. Ook moet de formule bij de sketch worden verwerkt tot een programma om de Ampère om te rekenen naar zoutgehalte met de Arduino.

Een handige manier om wisselstroom te gebruiken met de Arduino hebben we ook nog niet gevonden. Mocht dit niet mogelijk zijn met de Arduino, dan moet er een andere manier worden gevonden om elektrolyse te voorkomen tijdens de meting. Een optie is een meting nemen voordat er elektrolyse plaatsvindt (bijvoorbeeld met PWM) maar dit is minder betrouwbaar.

Nawoord

Om te beginnen vonden we dit een erg leuk project. Een typische O&O opdracht waarbij je eerst veel dingen moet onderzoeken, en daarna lekker aan de slag kan met iets ontwerpen. De Arduino. Als tip zal ik wel willen meegeven op op de planning te letten. We gingen natuurlijk naar het UCR tijdens het project, waardoor we er weer even in moesten komen.

Bronnenlijst

Multiflexmeter, 2016, van het www gehaald voor informatie www.multiflexmeter.nl

Arduino, van het www gehaald voor informatie en verschillende voorbeeldcodes
www.arduino.cc

Verder hebben we op internet nog wat voorbeelden van arduino opgezocht. Helaas weten we niet meer welke websites dit waren.

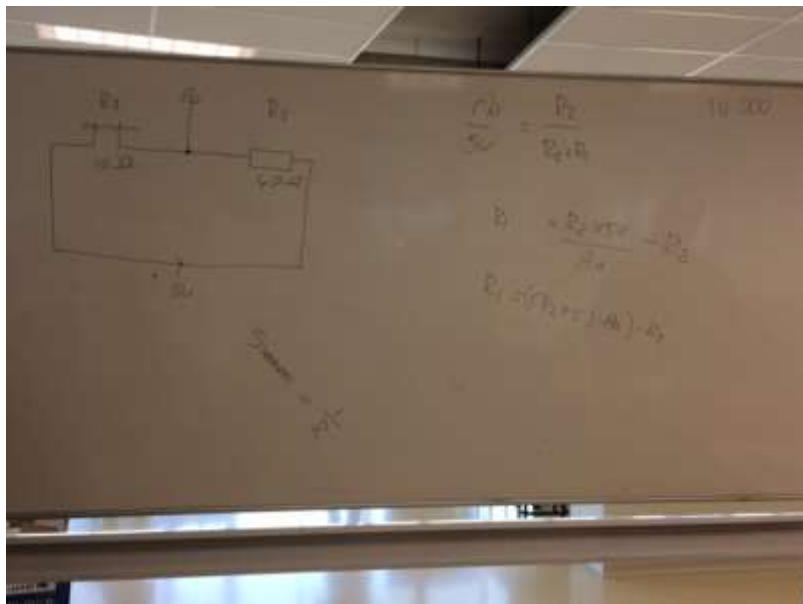
Company profile

The Waterschap is part of the government responsible for the waters in the Netherlands. It's tasks include keeping track of the water level and quality. It also makes sure there's never too much or too little water in a certain area. It takes care of trees surrounding the water and controls over 2000 km of road. Another thing they do is building and sustaining pumps which are used to pump away excess water. Their goal is to do these tasks as sustainable and easy as possible. There's many studies one could do to get job there, for example civil engineering or ecology. In the whole sector Waterschap in the Netherlands, over 12000 people have a job. The waterschap we do a project for employs 340 people. Our employer is the boss of the multiflexmeter project. He's in charge of the project and decides what features it should have. He will therefore decide which of our ideas is a good idea and should be used

Saliniteit experiment

We besloten zelf aan de hand van elektrodes een saliniteitssensor te maken, en we experimenteerden om er achter te komen welke formule we nodig hadden om het aantal ampère om te rekenen naar de hoeveelheid zout. We mengden steeds 10 gram zout in het water en keken op de Ammeter hoeveel het aantal Ampère veranderde. Dit ging fout omdat er elektrolyse plaatsvond en dit het oppervlakte van de negatieve elektrode beïnvloedde. We losten dit op door zo snel mogelijk een meting te noteren en het voltage te verlagen. Hieronder een lijst van onze metingen, die we hebben omgerekend naar weerstand in Ohm. Dit is namelijk wat word beïnvloed door de hoeveelheid zout.

Hoeveelheid zout in gram	Weerstand in Ohm
10	33.333
20	28.571
30	25
40	22.222
50	20
60	18.182
70	16.667



De foto met de formule om het aantal zout aan te geven.