# Opdracht 3.2

ADC Baby!

Deze opdracht bestaat uit twee kleinere opdrachten:

1. Maak een digitale dobbelsteen met 9 leds (zie plaatje). Als je op een knop drukt geven de leds een random waarde tussen 1 en 6 weer op de leds zoals je dat kent op een echte dobbelsteen. Hou bij het programma rekening met dender-gedrag.
2. **Maak een programma dat de waarde van een potentiometer vertaalt in een geluidssignaal tussen 100 en 10.000Hz. Laat dit signaal horen op de speaker.**



Opdracht 3.2 - ADC Baby! - Voorbeeld

## Aanpak en Uitvoering

Tijdens de les van week 3 zijn we aan de slag geweest met de potentiometer. We hebben een led van langzaam naar snel laten knipperen. Ik heb de led vervangen door de speaker en de code iets aangepast en extra functies toegevoegd:

* Ik heb een functie gemaakt die een waarde van 1 tot en met 1024 in de juiste verhouding teruggeeft tussen de 100 en 10.000. Dit zijn het aantal microseconden dat de speaker moet wachten voordat het weer stroom krijgt. Deze functie wordt ook wel map genoemd. Een groot deel van de code heb ik gevonden op [Stack Overflow](https://stackoverflow.com/questions/5731863/mapping-a-numeric-range-onto-another).
* Ook moest ik bereken hoe lang de speaker moest wachten om weer stroom te krijgen. Dit is te zien in onderstaande tabel.

|  |  |  |
| --- | --- | --- |
| Hz (hertz) | ms (milliseconden) | µs (microseconden) |
| 100 | 10 | 10.000 |
| 10.000 | 0.1 | 100 |

*Door het aanmaken van deze tabel kwam ik erachter dat de waardes geflipt werden bij het omrekenen. Door deze ontdekking heb ik de map functie laten mappen van de waarde 10.000 tot 100 in plaats van 100 tot 10.000. Hierdoor was er geen verdere berekening nodig.*

## Afbeelding

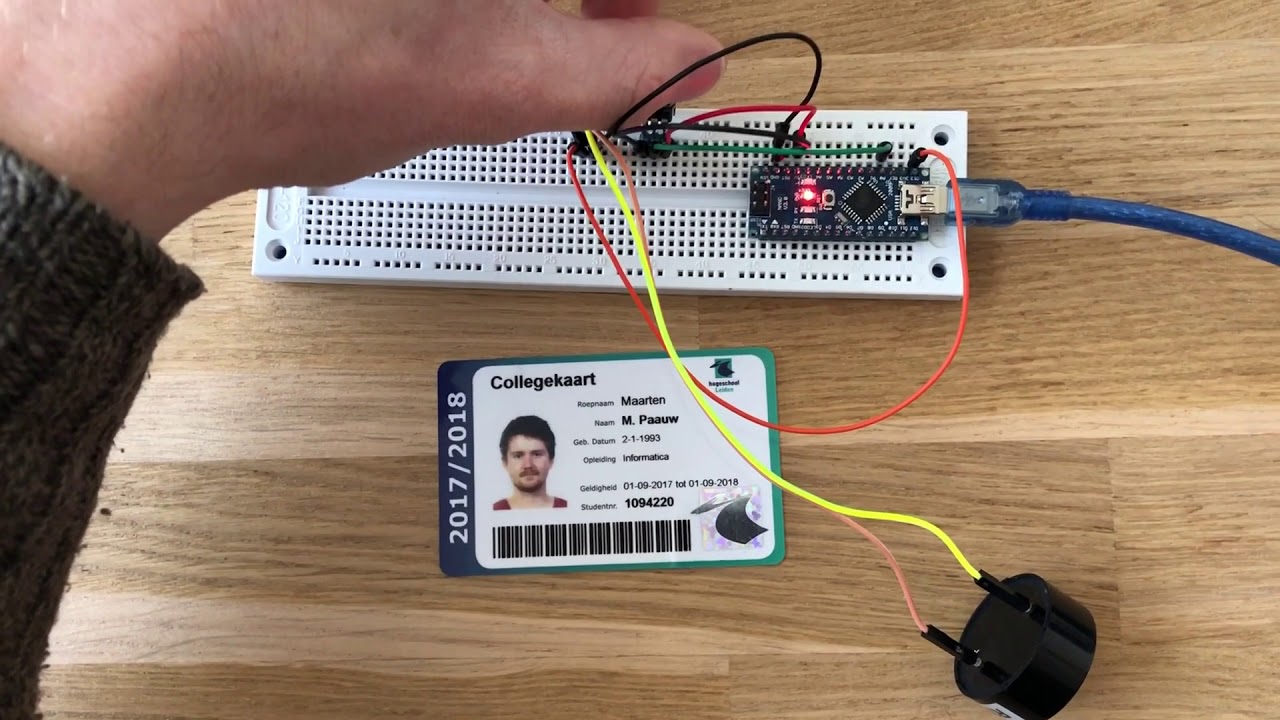


Opdracht 3.2 - ADC Baby! - Afbeelding

De afbeelding van de setup kan ook gedownload worden via de volgende link:

<https://raw.githubusercontent.com/maartenpaauw/IMTHE1/master/O3_2/assets/setup.jpg>

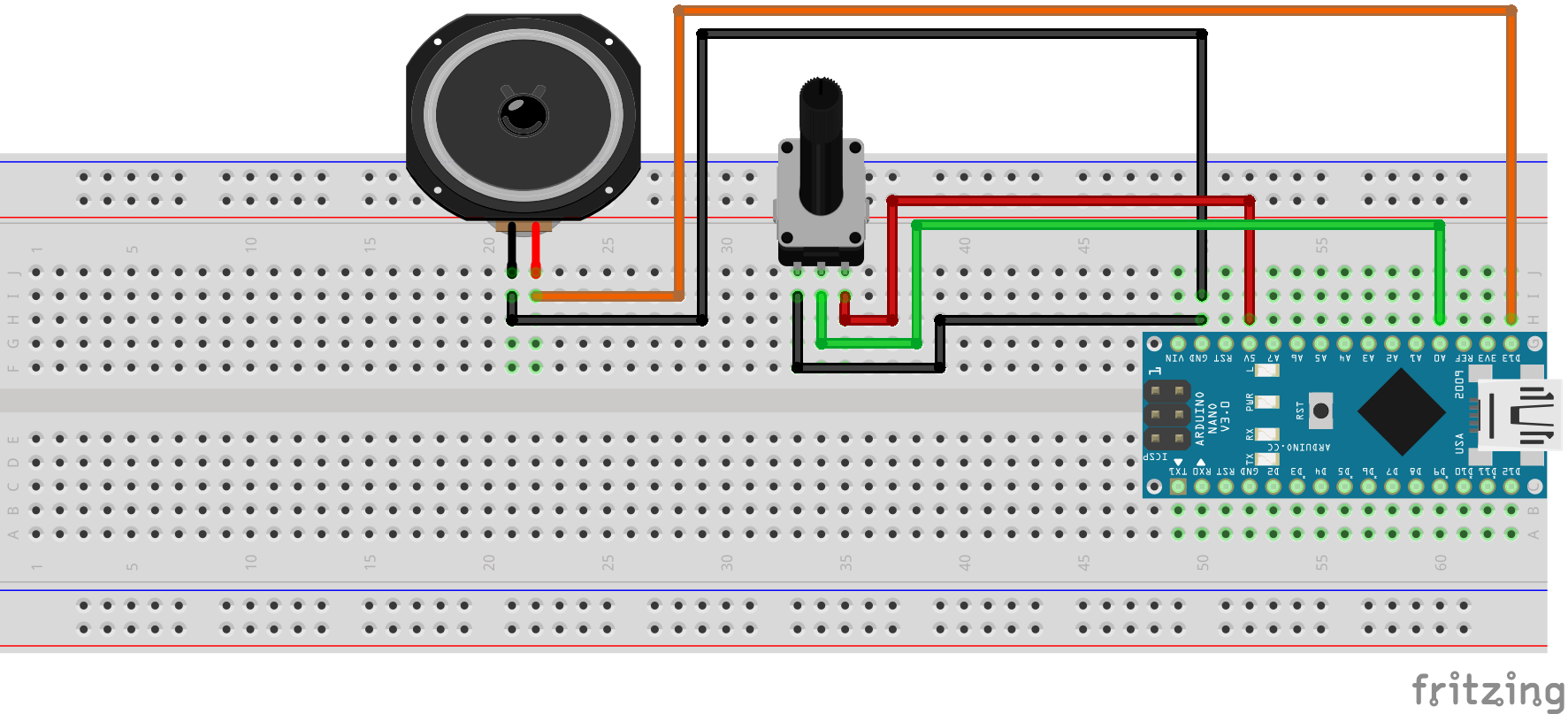
## Video

[](https://www.youtube.com/watch?v=gUwsOV7sM_M)

Deze video is ook te vinden op **Youtube**:

<https://www.youtube.com/watch?v=gUwsOV7sM_M>

## Breadboard Schema

[](https://raw.githubusercontent.com/maartenpaauw/IMTHE1/master/O3_2/assets/fritzing/schema.png)

Het **Fritzing** schema kan ook gedownload worden via de volgende link:

<https://github.com/maartenpaauw/IMTHE1/raw/master/O3_2/assets/fritzing/schema.fzz>

### Hardware

|  |
| --- |
| Onderdelen |
| Arduino Nano (v3.0) - 1× |
| Potentiometer - 1× |
| Speaker - 1× |
| Dupont kabel - 5× |

## Code

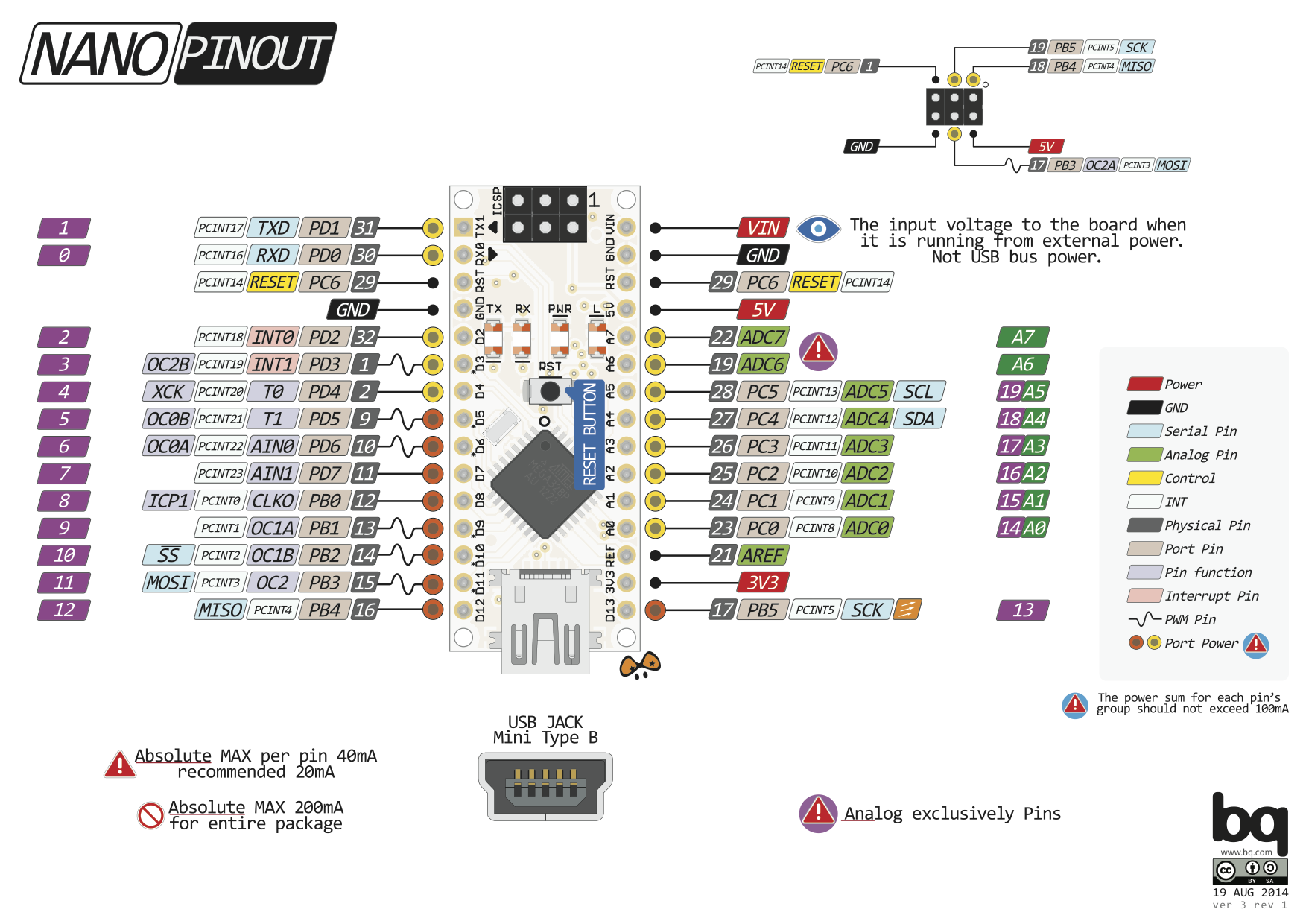
/\*  
 \* Opdracht 3.2 - ADC Baby!  
 \*   
 \* Eigenaar:  
 \* Maarten Paauw <s1094220@student.hsleiden.nl>  
 \* s1094220  
 \* INF3C  
 \*   
 \* Versie: 1  
 \* Aangemaakt: 23 februari 2018  
 \* Gewijzigd: 27 maart 2018  
 \*/  
  
#include <avr/io.h>  
#include <util/delay.h>  
#include <math.h>  
  
// Initialiseer de potential meter.  
void initADC()  
{  
 // Zet op 5 volt.  
 ADMUX |= (1 << REFS0);  
  
 // Divider op 128 voor 10 bit precisie.  
 ADCSRA |= ((1 << ADPS0) | (1 << ADPS1) | (1 << ADPS2));  
  
 // AD enable.  
 ADCSRA |= (1 << ADEN);  
}  
  
// Lees de potential meter.  
uint16\_t readADC()  
{  
 // Starten met lezen.  
 ADCSRA |= (1 << ADSC);  
  
 // Loop totdat er een waarde is.  
 loop\_until\_bit\_is\_clear(ADCSRA, ADSC);  
  
 // Geef de waarde terug.  
 return ADC;  
}  
  
// Dynamische timeout.  
void delay(uint16\_t time)  
{  
 // For loop.  
 for (uint16\_t i = 0; i < time; i++)  
 {  
 // Wacht voor 1 ms.  
 \_delay\_us(1);  
 }  
}  
  
// Preciezere afronding.  
// https://stackoverflow.com/questions/5731863/mapping-a-numeric-range-onto-another  
double round(double d)  
{  
 // Tel er 0.5 bij op en rond het getal af.  
 return floor(d + 0.5);  
}  
  
// Genereer de slope.  
// https://stackoverflow.com/questions/5731863/mapping-a-numeric-range-onto-another  
double slope (int output\_start, int output\_end, int input\_start, int input\_end)  
{  
 // Geef de slope terug.  
 return 1.0 \* (output\_end - output\_start) / (input\_end - input\_start);  
}  
  
// Map het getal van een range in een nieuwe range.  
// https://stackoverflow.com/questions/5731863/mapping-a-numeric-range-onto-another  
double map(int input, int input\_start, int output\_start, double static\_slope)  
{  
 // Geef het nieuwe getal terug.  
 return output\_start + round(static\_slope \* (input - input\_start));  
}  
  
// De main functie.  
int main(void)  
{  
 // B Bank initialiseren.  
 DDRB = (1 << PB5);  
  
 // Initialiseer de potential meter.  
 initADC();  
  
 // Defineer de slope.  
 double static\_slope = slope(10000, 100, 1, 1024);  
  
 // Loop voor altijd.  
 while (1)  
 {  
 // Lees de waarde van de potential meter uit.  
 uint16\_t pwm = readADC();  
  
 // Zet de B pin aan.  
 PORTB = (1 << PB5);  
  
 // Bereken de delay.  
 double mapped = map(pwm, 1, 10000, static\_slope) / 2;  
  
 // Wacht voor een X aantal µs.  
 delay(mapped);  
  
 // Zet de B pin uit.  
 PORTB = 0;  
  
 // Wacht voor een X aantal µs.  
 delay(mapped);  
 }  
  
 // Geef een 0 terug.  
 return 0;  
}

De code kan ook gevonden worden in mijn **GitHub** *repository* via de volgende link:

<https://github.com/maartenpaauw/IMTHE1/blob/master/O3_2/src/main.c>

## Datasheets

### Arduino Nano



Arduino Nano Pinout

De [Arduino Nano Pinout](https://raw.githubusercontent.com/maartenpaauw/IMTHE1/master/O3_2/assets/data_sheets/nano.png) is uitgedeeld tijdens de eerste les en terug te vinden via het volgende forum post:

<https://forum.arduino.cc/index.php?topic=147582.0>

### RS 7243119

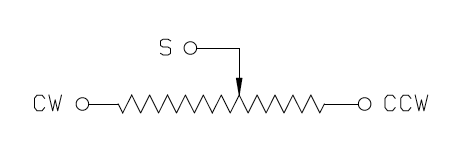


RS 7243119 Pinout

De [RS 7243119 Pinout](https://raw.githubusercontent.com/maartenpaauw/IMTHE1/master/O3_2/assets/data_sheets/rs-7243119.png) heb ik gevonden in de datasheet van de RS 7243119. Deze datasheet heb ik gevonden in de specificaties bij een webshop en is terug te vinden via de volgende link:

<https://www.alliedelec.com/m/d/69ca74a806a82d744141c552ce43700a.pdf>

### Potentiometer



RS Potentiometer Pinout

De [Potentiometer Pinout](https://raw.githubusercontent.com/maartenpaauw/IMTHE1/master/O3_2/assets/data_sheets/potentiometer.png) heb ik gevonden in de datasheet van de Potentiometer. Deze datasheet is terug te vinden via de volgende link:

<https://components101.com/sites/default/files/component_datasheet/potentiometer%20datasheet.pdf>

## Bronvermelding

* <https://forum.arduino.cc/index.php?topic=147582.0> (Arduino Nano Pinout)
* <https://www.alliedelec.com/m/d/69ca74a806a82d744141c552ce43700a.pdf> (RS 7243119 Datasheet)
* <https://stackoverflow.com/questions/5731863/mapping-a-numeric-range-onto-another> (Mapping a Numeric Range Onto Another)
* [https://cdn-learn.adafruit.com/assets/assets/000/027/586/original/light\_control-pot.png](https://components101.com/sites/default/files/component_datasheet/potentiometer%20datasheet.pdf) (Potentiometer Datasheet)