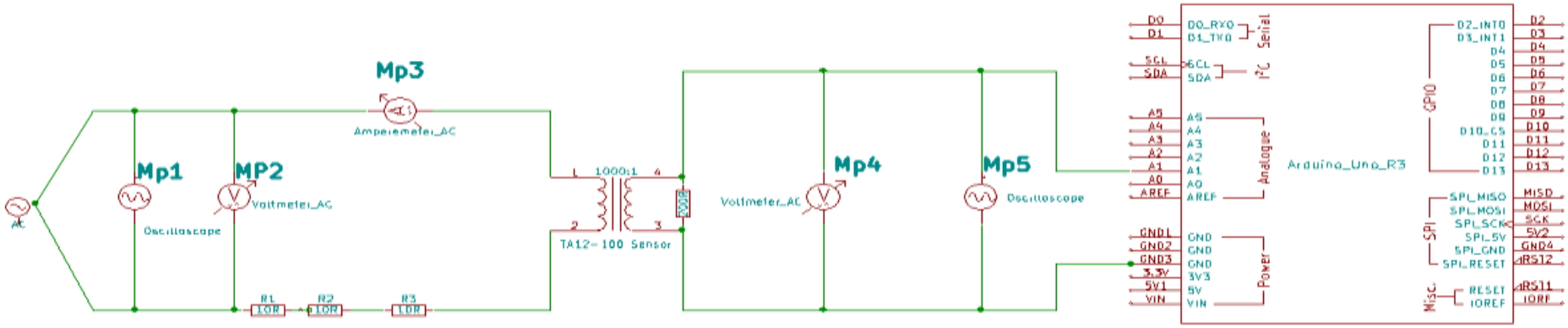
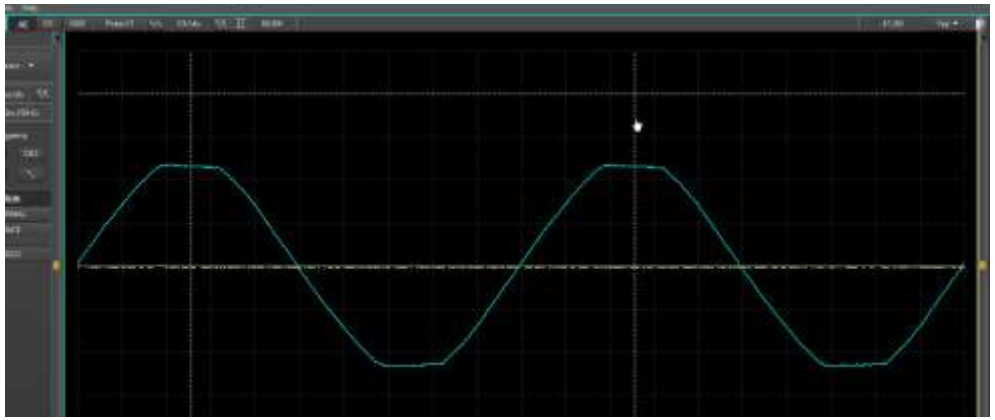
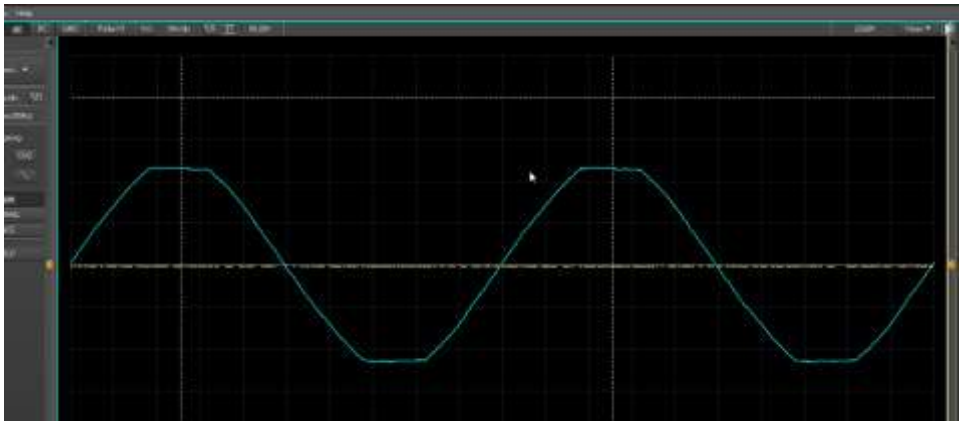
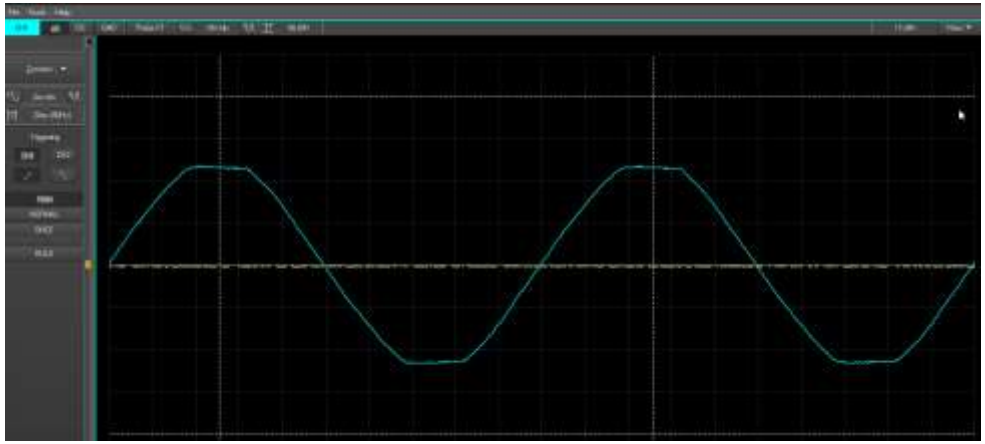
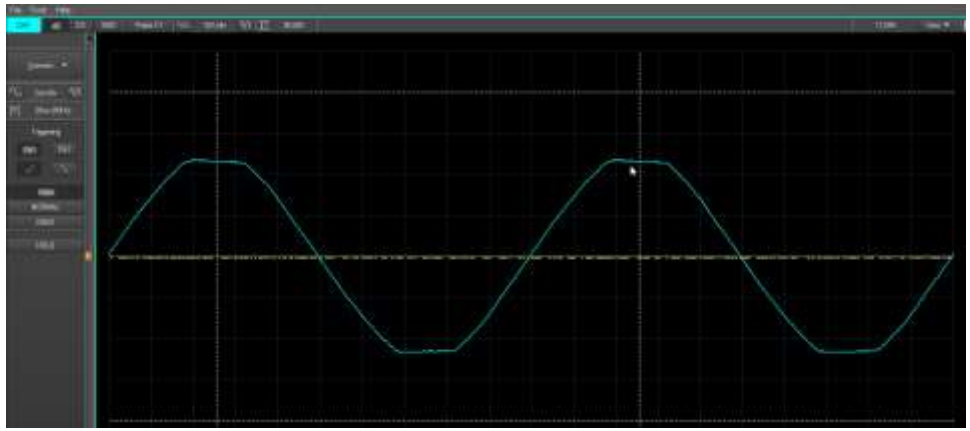
1. Testdata TA12-100 sensor met Ohmse belasting



Data van primaire meetkring met ohmse belasting

Primaire Kring analyse													
Load (Ω)	Mp1 (VC165) VTrms	Mp2 (DVM98) ITrms	Mp3 (WFS210) Vpp	Mp3 (WFS210) VMax	Mp3 (WFS210) VTrms	Mp3 (WFS210) Vrms		Mp1 (DVM98) VTrms	Mp1 (VC165) ITrms		Calc Gem VTrms	Calc Gem ITrms	Calc Gem Load
30 Ω	16,50 V	0,565 A	46,80 V	23,60 V	17,04 V	17,09 V		16,34 V	0,543 A		16,63 V	0,554 A	30,02 Ω
23,5 Ω	16,57 V	0,726 A	46,80 V	23,60 V	17,04 V	17,09 V		16,18 V	0,701 A		16,61 V	0,713 A	23,28 Ω

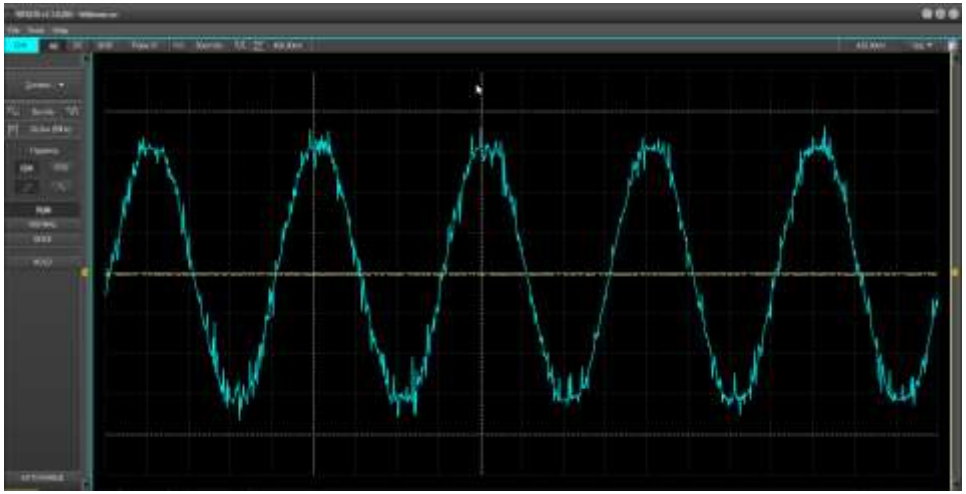
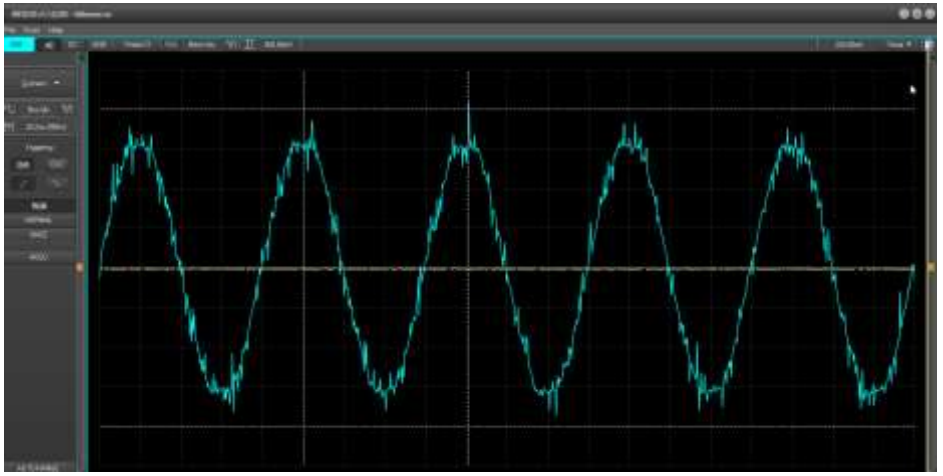
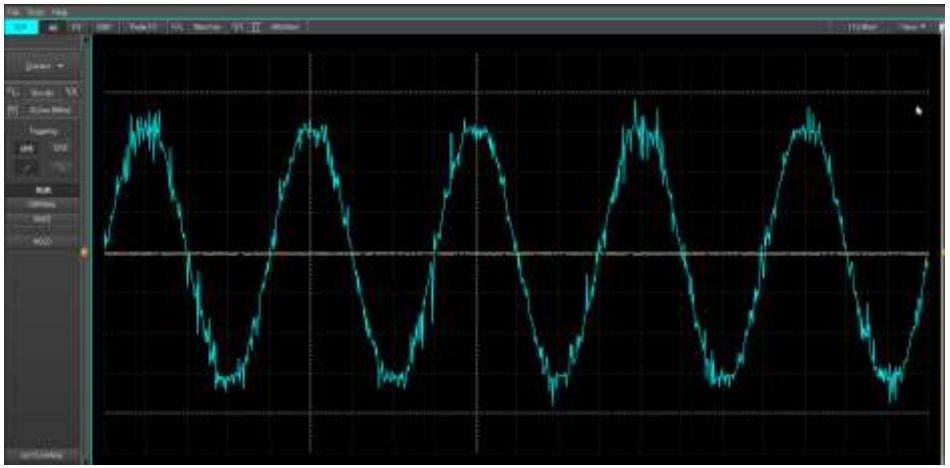
Bijhorende Scopeplaatjes van primaire meetkring met ohmse belasting

Mp2 : Vpp : 	Mp2 :Vmax 
Mp2 :Vtrms 	Mp2 :Vrms 

Data van secundaire meetkring met ohmse belasting 30 ohm

Secundaire Kring analyse + Arduino Output testprogramma								
Load (Ω)	Mp4 (VC165) VTrms	Mp5 (WFS210) Vpp	Mp5 (WFS210) VMax	Mp5 (WFS210) VTrms		Arduino Progr SensorMax	Arduino Progr Efectieve Stroom	Afwijking
30 Ω	0,109 V	395 mV	200 mV	113 mV		29	509 mA	5 %

Bijhorende Scopeplaatjes van secundaire meetkring met ohmse belasting 30 ohm

Mp5 : Vpp 	Mp5 : Vmax 
Mp5 : Vtrms 	

Arduino Prog en Log van secundaire meetkring met ohmse belasting 30 ohm

```
/*****************************************
  FILE:   DEMO_Electrity Sensor.pde
  PURPOSE:  Electrity sensor sketch for Arduino
  Created by Stan Lee from Iteadstuduino
  E-mail:   Lizq@iteadstudio.com
  DATE:   2013/4/20
*****************************************/

#define ELECTRICITY_SENSOR A1 // Analog input pin that sensor is attached to

float amplitude_current;              //amplitude current
float effective_value;       //effective current

void setup()
{
  Serial.begin(9600);
  pins_init();
}
void loop()
{
  int sensor_max;
  sensor_max = getMaxValue();
  Serial.print("sensor_max = ");
  Serial.println(sensor_max);
  //the VCC on the Grove interface of the sensor is 5v
  amplitude_current=(float)sensor_max/1024*5/200*1000000;
  effective_value=amplitude_current/1.414;
  //minimum_current=1/1024*5/200*1000000/1.414=24.4(mA)
  //Only for sinusoidal alternating current
  Serial.println("The amplitude of the current is(in mA)");
  Serial.println(amplitude_current,1);//Only one number after the decimal point
  Serial.println("The effective value of the current is(in mA)");
  Serial.println(effective_value,1);

  delay(10000);
}
void pins_init()
{
  pinMode(ELECTRICITY_SENSOR, INPUT);
}
/*Function: Sample for 1000ms and get the maximum value from the SIG pin*/
int getMaxValue()
{
  int sensorValue;            //value read from the sensor
  int sensorMax = 0;
  uint32_t start_time = millis();
  while((millis()-start_time) < 1000)//sample for 1000ms
  {
```

```
sensor_max = 30

The amplitude of the current is(in mA)

732.4

The effective value of the current is(in mA)

518.0

sensor_max = 30

The amplitude of the current is(in mA)

732.4

The effective value of the current is(in mA)

518.0

sensor_max = 29

The amplitude of the current is(in mA)

708.0

The effective value of the current is(in mA)

500.7

sensor_max = 30

The amplitude of the current is(in mA)

732.4

The effective value of the current is(in mA)

518.0

sensor_max = 29

The amplitude of the current is(in mA)

708.0
```

```
    sensorValue = analogRead(ELECTRICITY_SENSOR);
//    Serial.println("");
//    Serial.print("functie : sensorValue = ");
//    Serial.print(sensorValue);
    if (sensorValue > sensorMax)
    {
      /*record the maximum sensor value*/
      sensorMax = sensorValue;
    }
//    Serial.print("   -->  : sensorMax = ");
//    Serial.println(sensorMax);
  }
  return sensorMax;
}
```

```
The effective value of the current is(in mA)

500.7

sensor_max = 29

The amplitude of the current is(in mA)

708.0

The effective value of the current is(in mA)

500.7

sensor_max = 30

The amplitude of the current is(in mA)

732.4

The effective value of the current is(in mA)

518.0

sensor_max = 30

The amplitude of the current is(in mA)

732.4
```

Vergelijking met andere TA12-100 sensor en andere Arduino en Ohmse belasting 30 ohm.

Gebruikte Arduino	Gebruikte TA12-100	Mp3 Itrms	Mp4 Vtrms	SensorMax (programma)	Mp5 Vpp	Mp5 VMax	Mp5 VTrms	LoadStroom (programma)	Verschil Stroom(%)	Opmerking
RobDyn 1	1	0,565 A	0,109 V	29	0,395 V	0,200 V	0,109 V	0,509 A	5 %	Zie details boven
RobDyn 1	2	0,569 A	0,107 V	30	0,396 V	0,206 V	0,113 V	0,535 A	3 %	
AdvTech 1	2	0,565 A	0,108 V	29	0,360 V	0,201 V	0,112 V	0,509 A	5 %	

Conclusie van testing met Ohmse belast!ing

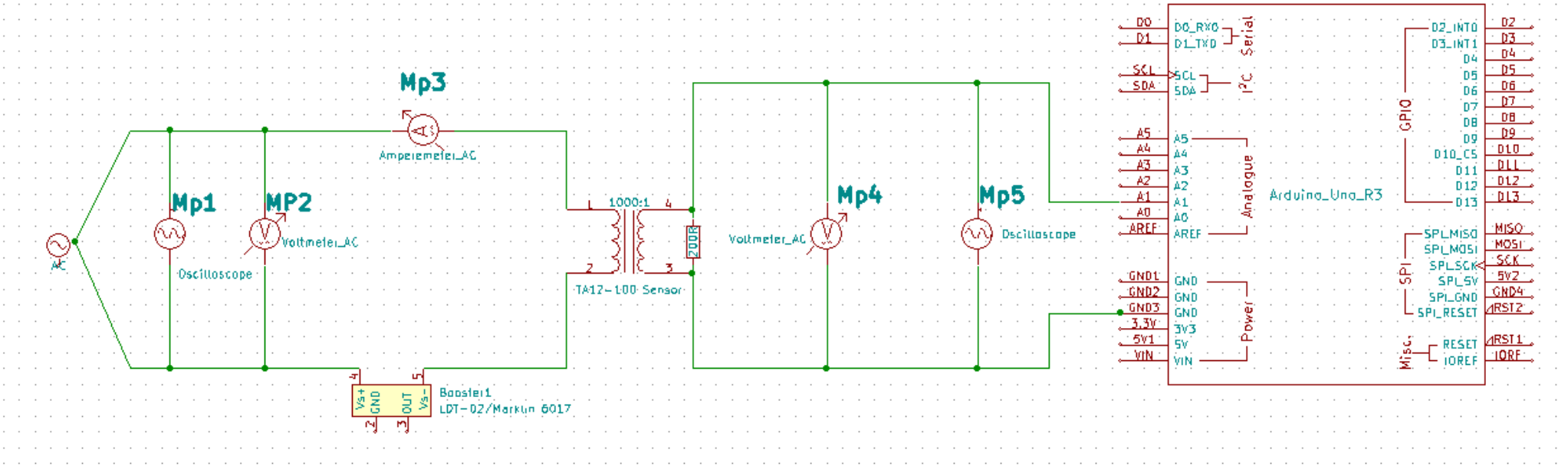
- Afwijking Arduino Uno met TA12-100 gebaseerde sensor Itlead : ongeveer 5 % afwijking

- Geen significante verschillen tussen de geteste sensors (2) en arduinos (2 van verschillende fabrikant)

- Gebruikte demo code van ItLead is goed voor zuiver sinus signaal en de door ons gebruikte omgeving (ringtransfo 230/17V 50 Hz) met ohmse belasting.

- Multimeters geven onderling afwijkingen wanneer ze zelfde grootheid meetheid meten. Ook waarde Scope kan verschillend zijn. Misschien best om gemiddelde te nemen tussen de toestellen.

2. Testdata TA12-100 sensor met boosterkring + IB

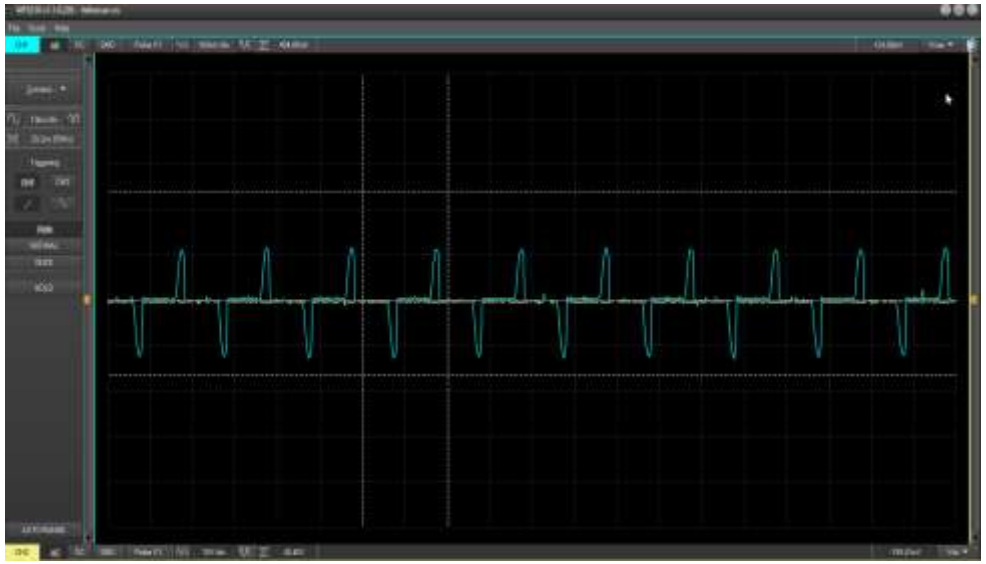


Data van secundaire meetkring met onbelaste boosterkring B1 + Intellibox

Secundaire Kring analyse + Arduino Output testprogramma										
Booster	Mp3 (DVM98) A Trms	Mp4 (VC165) VTrms	Mp5 (WFS210) Vpp	Mp5 (WFS210) VMax	Mp5 (WFS210) VTrms	Arduino Progr SensorMax	Arduino Progr Max Stroom Load	Arduino Progr Eff Stroom Load	Afwijkings-factor Op Prim Load	Opmerking
B1+ IB	0,231 A	0,043 V	300 mV	140 mV	45 mV	23	561 mA	397 mA	1,71	De Scopebeelden tonen aan dat we helemaal geen sinusoidaal signaal meer hebben op de boosterbelasting. De gebruikte faktor van 0,707 mag dus niet gebruikt worden !
B1+IB+ ohm belasting	0,446 A	0,085 V	420 mV	224 mV	85 mV	46	1123 mA	794 mA	1,78	
B2+ IB	0,241 A	0,045 V	280 mV	128 mV	44 mV	23	561 mA	397 mA	1,64	
B2+IB+ ohm belasting	0,736 A	0,136 V	650 mV	424 mV	134 mV	82	2002 mA	1415 mA	1,92	
B2+IB+ ohm belastin	0,750 A	0,138 V				86	2098 mA	1483 mA	1,97	
B2+IB+ ohm belastin	0,845 A	0,149 V				97	2368 mA	1674 mA	1,98	
B2+IB+ ohm belastin	1,720 A	0,277 V				229	5876 mA	4154 mA	2,41	

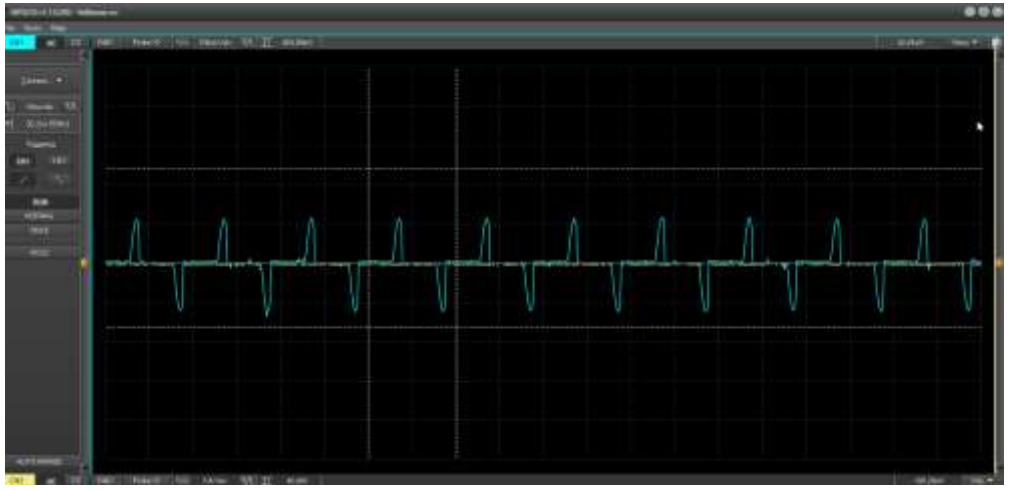
Bijhorende Scopeplaatjes van secundaire meetkring met booster belasting.

Mp5 : Vpp

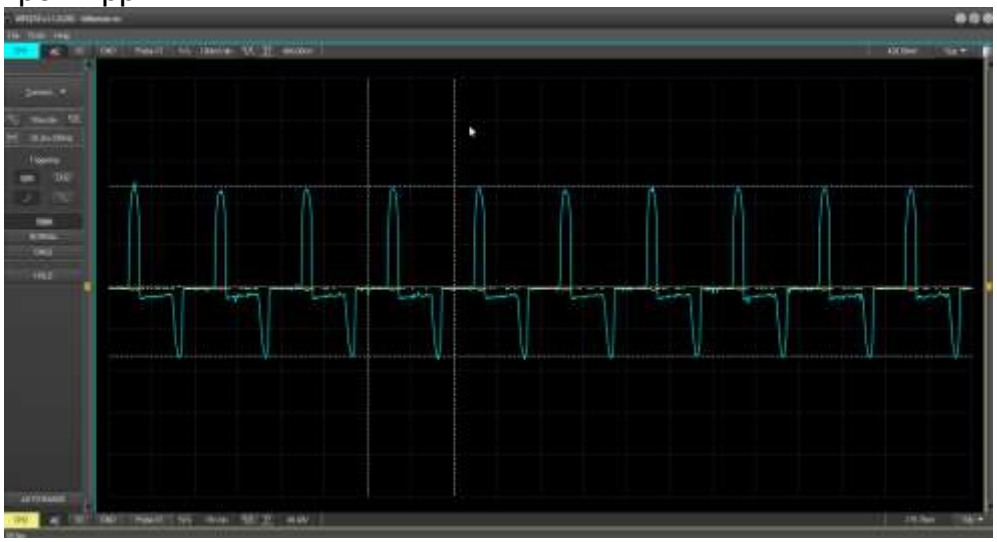
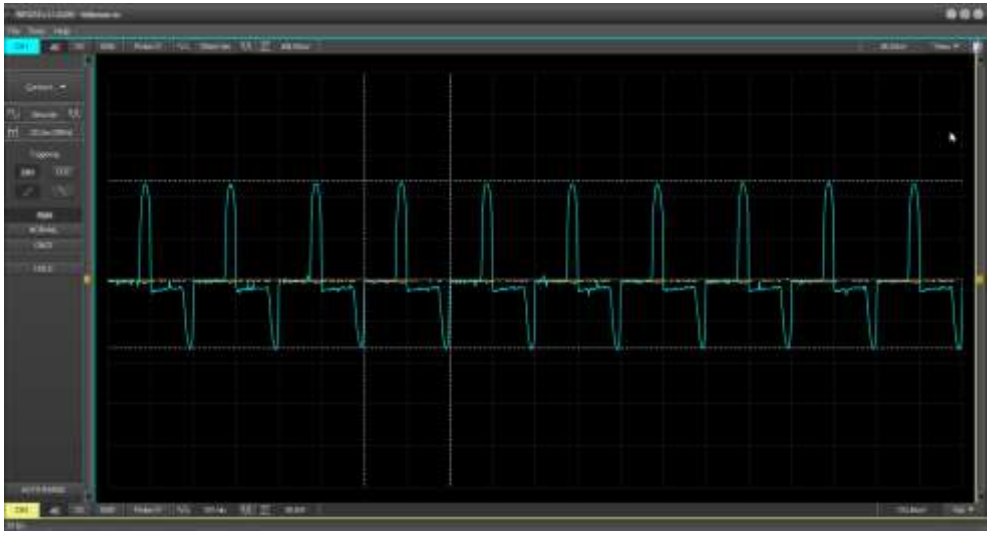


Mp5: Max

Mp5 : Vtrms

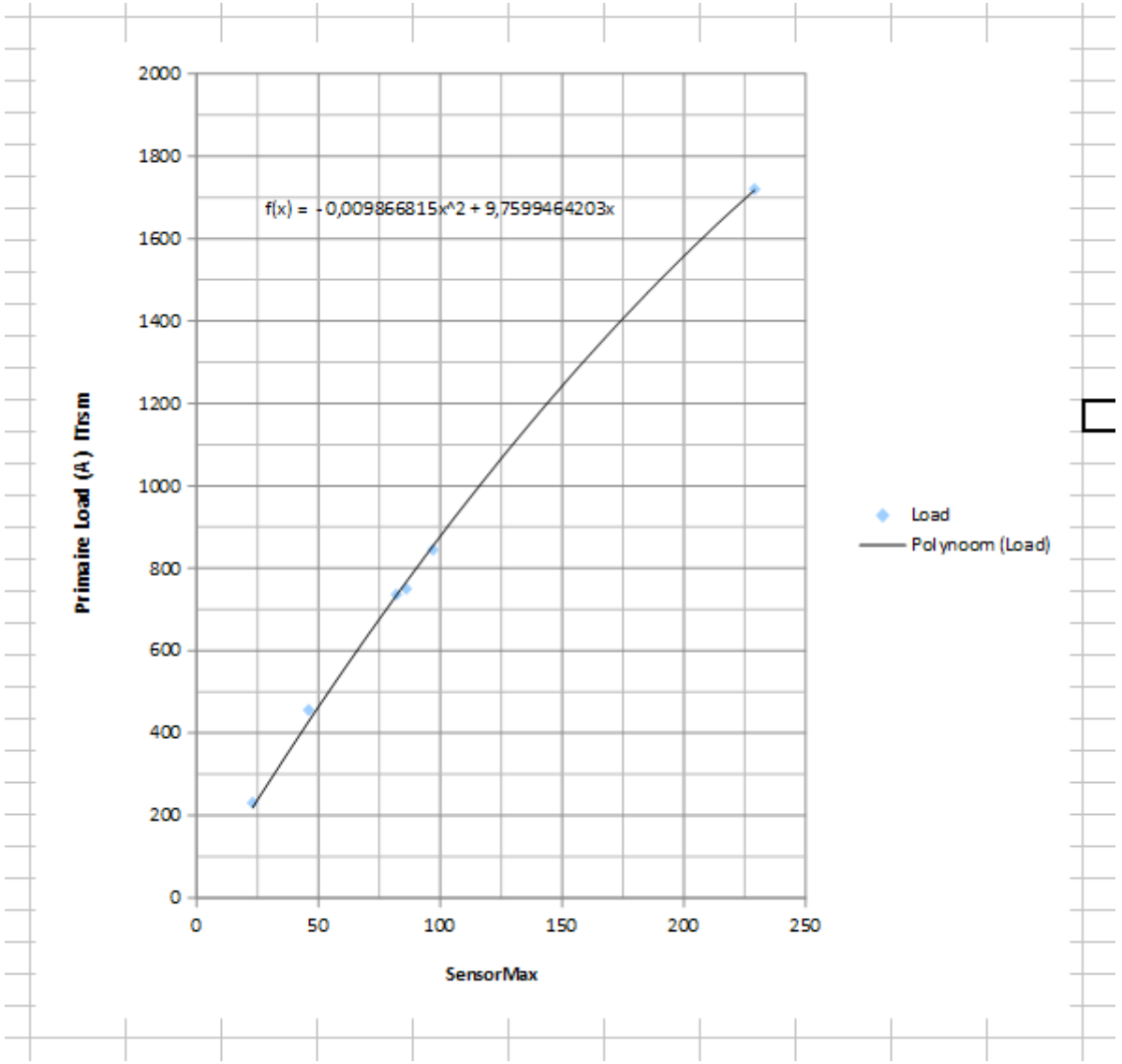


Bijhorende Scopeplaatjes van secundaire meetkring met booster belasting B1 met extra ohmse belasting.

Mp5 : Vpp	Mp5: Trms
	

Bepalen van de meest gunstige faktor :

We doen een analyse tussen sensorMax (bit) en de gemeten load in de primare kring (A Rms)



De gekozen methode is een polynoom welke een beter gemiddeld resultaat gaf dan andere regressievormen.

Resultaat van de testing met de nieuwe formule als faktor . Gewijzigde testprogramma , zie volgende blad

(we hebben hier de multimeters ook eens van plaats verwisseld om een extra vergelijking te doen)

Booster	SensorMax	SensorSpanning Mp4 (VC195) mVTrms	SensorSpanning Mp4 (DVM98) mVTrms	Primaire Stroom Mp3 (DVM98) mA Trms	Primaire Stroom Mp3 (VC195) mA Trms	Lin Reg Calc SensorSpanning mV Trms	Lin Reg Calc Prim Stroom mA Trms	Poly Reg Calc SensorSpanning mV Trms	Poly Reg Calc Prim Stroom mA Trms	Afwijkings-Factor Op Prim Load
B2 + IB	25	45	45	229	222	60	302	47	237	1.05
B2 + IB + Ohm Load1	84	136	133	737	693	114	720	150	749	1.04
B2 + IB + Ohm Load2	97	151	148	839	795	162	812	170	853	1.04
B2 + IB + Ohm Load3	211	264	256	1719	1622	333	1667	330	1657	0.99

Arduino Prog en Log van gewijzigde testprogramma voor booster belasting (Niet Sinusoidale belasting) : testing 2 regressiemethodes

```
/*****************************************
  FILE:   DEMO_Electrity Sensor.pde
  PURPOSE:  Electrity sensor sketch for Arduino
  Created by Stan Lee from Iteadstuduino
  E-mail:   Lizq@iteadstudio.com
  DATE:   2013/4/20
*****************************************/

#define ELECTRICITY_SENSOR A1 // Analog input pin that sensor is attached to

float amplitude_current;              //amplitude current
float effective_value;       //effective current


void setup()
{
  Serial.begin(9600);
  pins_init();
}
void loop()
{
  int sensor_max;



  sensor_max = getMaxValue();


 // Dirk extra check
 // Arduino AI ADC sensitivity (default 5V ref) = 4.88 mV/count


  float calculated_sensor_voltage_Vmax=0.0;
  float calculated_sensor_voltage_VTrms=0.0;
  float calculated_sensor_current_ATrms=0.0;
  float calculated_sensor_voltage_VTrms_lineair=0.0;
  float calculated_sensor_voltage_VTrms_polygoon=0.0;
  float calculated_booster_current_ATrms=0.0;
  float calculated_booster_current_ATrms_lineair=0.0;
  float calculated_booster_current_ATrms_polygoon=0.0;
  float defined_booster_factor=0.33;


  calculated_booster_current_ATrms_lineair=(float)(125.93+(7.0735*sensor_max));
  calculated_sensor_voltage_VTrms_lineair=(float)((calculated_booster_current_ATrms_lineair/1000.0)*200.0);
  Serial.println("  ");
  Serial.println("Met Regressieformule lineair ");
  Serial.print("  sensor_max  :  ");
  Serial.print(sensor_max);
  Serial.print("  Bit  ,   calculated RMS on Analog Input  = ");
```

```
Met Regressieformule lineair

 sensor_max  :  0  Bit  ,   calculated RMS on Analog Input  = 25.19 mV ,  calculated Load Booster (I Rms) A = 125.93  mA

Met Regressieformule polygoon

 sensor_max  :  0  Bit  ,   calculated RMS on Analog Input  = 0.00 mV ,  calculated Load Booster (I Rms) A = 0.00  mA


Met Regressieformule lineair

 sensor_max  :  0  Bit  ,   calculated RMS on Analog Input  = 25.19 mV ,  calculated Load Booster (I Rms) A = 125.93  mA

Met Regressieformule polygoon

 sensor_max  :  0  Bit  ,   calculated RMS on Analog Input  = 0.00 mV ,  calculated Load Booster (I Rms) A = 0.00  mA


Met Regressieformule lineair

 sensor_max  :  25  Bit  ,   calculated RMS on Analog Input  = 60.55 mV ,  calculated Load Booster (I Rms) A = 302.77  mA

Met Regressieformule polygoon

 sensor_max  :  25  Bit  ,   calculated RMS on Analog Input  = 47.56 mV ,  calculated Load Booster (I Rms) A = 237.81  mA


Met Regressieformule lineair
```

```
  Serial.print(calculated_sensor_voltage_VTrms_lineair);
  Serial.print(" mV ,  calculated Load Booster (I Rms) A = ");
  Serial.print(calculated_booster_current_ATrms_lineair);
  Serial.println("  mA");

  calculated_booster_current_ATrms_polygoon=(float)( (-0.0099*sensor_max*sensor_max) +
(9.7599*sensor_max) );

calculated_sensor_voltage_VTrms_polygoon=(float)((calculated_booster_current_ATrms_polygoon/1000.0)*200.0);
  Serial.println("Met Regressieformule polygoon ");
  Serial.print("  sensor_max  :  ");
  Serial.print(sensor_max);
  Serial.print("  Bit  ,   calculated RMS on Analog Input  = ");
  Serial.print(calculated_sensor_voltage_VTrms_polygoon);
  Serial.print(" mV ,  calculated Load Booster (I Rms) A = ");
  Serial.print(calculated_booster_current_ATrms_polygoon);
  Serial.println("  mA");
  Serial.println("  ");


  // Dirk extra check




/*  Origineel stuk niet meer doen , is niet geschikt voor boosterbelasting)
  //the VCC on the Grove interface of the sensor is 5v
  amplitude_current=(float)sensor_max/1024*5/200*1000000;
  effective_value=amplitude_current/1.414;
  //minimum_current=1/1024*5/200*1000000/1.414=24.4(mA)
  //Only for sinusoidal alternating current
  Serial.println("The amplitude of the current is(in mA)");
  Serial.println(amplitude_current,1);//Only one number after the decimal point
  Serial.println("The effective value of the current is(in mA)");
  Serial.println(effective_value,1);
 */


  delay(10000);
}
void pins_init()
{
  pinMode(ELECTRICITY_SENSOR, INPUT);
}
/*Function: Sample for 1000ms and get the maximum value from the SIG pin*/
int getMaxValue()
{
  int sensorValue;            //value read from the sensor
  int sensorMax = 0;
  uint32_t start_time = millis();
```

```
 sensor_max  :  25  Bit  ,   calculated RMS on Analog Input  = 60.55 mV ,
calculated Load Booster (I Rms) A = 302.77  mA

Met Regressieformule polygoon

 sensor_max  :  25  Bit  ,   calculated RMS on Analog Input  = 47.56 mV ,
calculated Load Booster (I Rms) A = 237.81  mA

Met Regressieformule lineair

 sensor_max  :  80  Bit  ,   calculated RMS on Analog Input  = 138.36 mV ,
calculated Load Booster (I Rms) A = 691.81  mA

Met Regressieformule polygoon

 sensor_max  :  80  Bit  ,   calculated RMS on Analog Input  = 143.49 mV ,
calculated Load Booster (I Rms) A = 717.43  mA

Met Regressieformule lineair

 sensor_max  :  83  Bit  ,   calculated RMS on Analog Input  = 142.61 mV ,
calculated Load Booster (I Rms) A = 713.03  mA

Met Regressieformule polygoon

 sensor_max  :  83  Bit  ,   calculated RMS on Analog Input  = 148.37 mV ,
calculated Load Booster (I Rms) A = 741.87  mA

Met Regressieformule lineair

 sensor_max  :  80  Bit  ,   calculated RMS on Analog Input  = 138.36 mV ,
calculated Load Booster (I Rms) A = 691.81  mA

Met Regressieformule polygoon

 sensor_max  :  80  Bit  ,   calculated RMS on Analog Input  = 143.49 mV ,
```

```
  while((millis()-start_time) < 1000)//sample for 1000ms
  {
    sensorValue = analogRead(ELECTRICITY_SENSOR);
//    Serial.println("");
//    Serial.print("functie : sensorValue = ");
//    Serial.print(sensorValue);
    if (sensorValue > sensorMax)
    {
      /*record the maximum sensor value*/
      sensorMax = sensorValue;
    }
//    Serial.print("   -->  : sensorMax = ");
//    Serial.println(sensorMax);
  }
  return sensorMax;
}
```

```
calculated Load Booster (I Rms) A = 717.43  mA




Met Regressieformule lineair

 sensor_max : 94  Bit ,  calculated RMS on Analog Input  = 158.17 mV ,
calculated Load Booster (I Rms) A = 790.84  mA

Met Regressieformule polygoon

 sensor_max : 94  Bit ,  calculated RMS on Analog Input  = 165.99 mV ,
calculated Load Booster (I Rms) A = 829.95  mA




Met Regressieformule lineair

 sensor_max : 94  Bit ,  calculated RMS on Analog Input  = 158.17 mV ,
calculated Load Booster (I Rms) A = 790.84  mA

Met Regressieformule polygoon

 sensor_max : 94  Bit ,  calculated RMS on Analog Input  = 165.99 mV ,
calculated Load Booster (I Rms) A = 829.95  mA




Met Regressieformule lineair

 sensor_max : 211  Bit ,  calculated RMS on Analog Input  = 323.69 mV ,
calculated Load Booster (I Rms) A = 1618.44  mA

Met Regressieformule polygoon

 sensor_max : 211  Bit ,  calculated RMS on Analog Input  = 323.72 mV ,
calculated Load Booster (I Rms) A = 1618.58  mA
```

	Met Regressieformule lineair sensor_max : 210 Bit , calculated RMS on Analog Input = 322.27 mV , calculated Load Booster (I Rms) A = 1611.37 mA Met Regressieformule polygoon sensor_max : 210 Bit , calculated RMS on Analog Input = 322.60 mV , calculated Load Booster (I Rms) A = 1612.99 mA