FOTOMIND

## Assignment:

**NE**: Op een bepaald event sturen we verscheidene camera’s tegelijkertijd aan, om op een gestructureerde manier foto’s nemen. De camera sturen we aan met een Arduino, iedere camera heeft één Arduino module. We sturen alle modules tegelijkertijd & draadloos aan vanuit één Master Arduino. Dit hoort op een dynamische manier mogelijk te zijn zodat we meer of minder camera’s kunnen gebruiken indien nodig. Voor een continue werking van deze opstelling halen we de foto’s van de geheugen kaart op met behulp van een RaspberryPi & sturen deze rechtstreeks door naar een server, zo kan de camera foto’s blijven nemen zonder onderbreking. Een verdere uitwerking van het project draait rond het visueel voorstellen van de foto’s.

**EN**: We take via a structured way a couple of photo’s by controlling several camera’s at the same time trough an Arduino. Every camera has it’s own Arduino module. Every module connected to the camera is a slave & is wireless supervised by one master Arduino, so we can dynamically add & remove camera’s. Every camera has it’s own RaspberryPi to collect the photo’s on the camera and send them to the server. This function allows us to keep taking pictures without removing the SD card.

1. Aansturen camera met een Arduino
2. Communicatie tussen Arduino’s
   1. Bedraad aansturen
   2. Draadloos aanturen
3. Uitlezen van de foto’s met een RaspberryPi
4. Data op de RaspberryPi naar een server sturen
5. Control the camera trough an Arduino
6. Communicate between several Arduino’s
   1. Wirelles connection
   2. Wired connection
7. Read the photo’s from the camera
8. Send the photo’s to the server

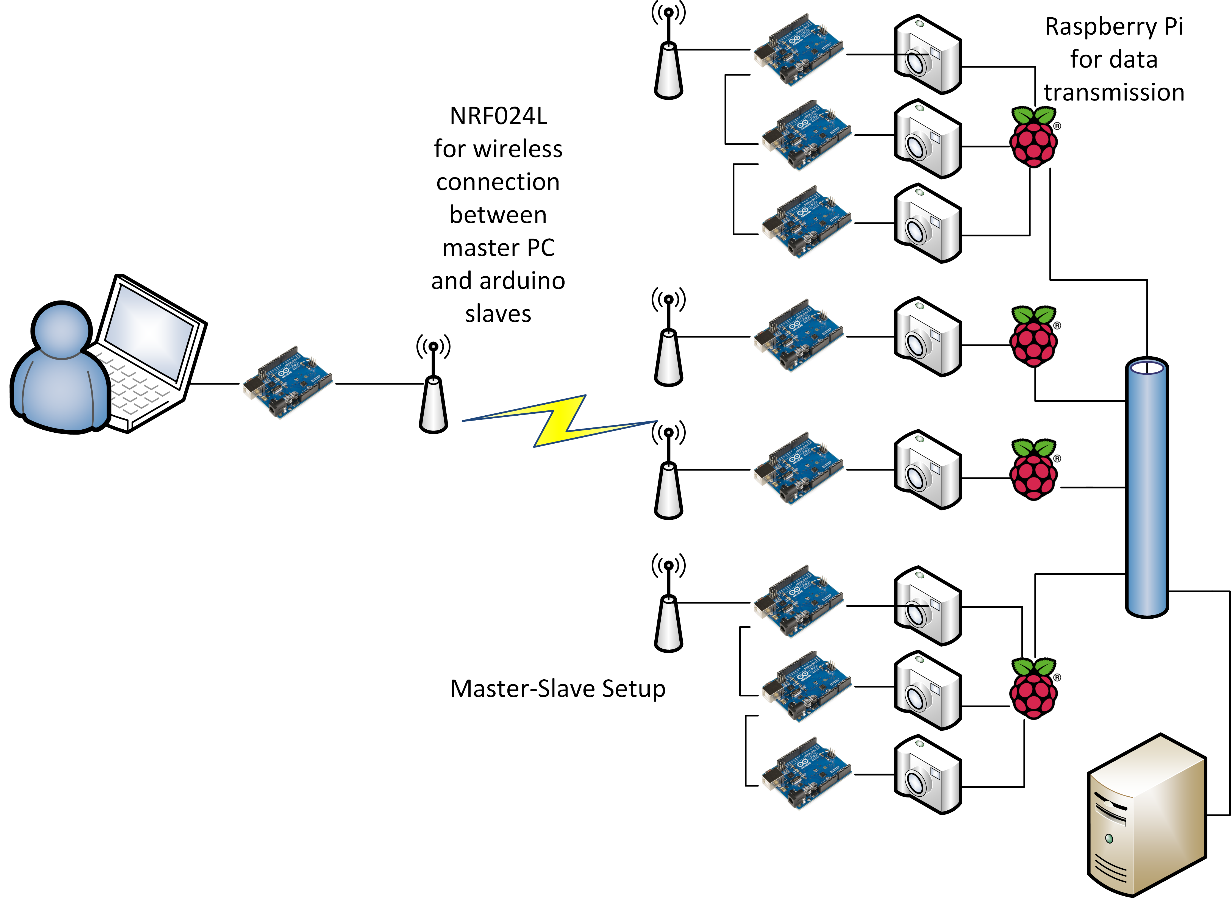
# 20/11/2014

**NE:** Christian legde ons de eerste stappen uit van het project, door een eerste opstelling op te zetten voor highspeed foto’s. De opstelling bestond uit een Arduino, een laser & een sensor die op elkaar gericht werden. Indien de laser straal onderbroken werd namen we een foto. Hierin werd al meteen duidelijk dat de afstelling van de camera & het inschatten van de val van het onderwerp van cruciaal belang was.

**EN:** Christian explained the first steps from the project to us, by making an alignment for highspeed photography. The alignment existed from an Arduino, a camera, a laser & a sensor which are all connected. If the laser beam was broken by a solid object the shutter realease was triggered. It got clear on how sensitive the alignement is & how the estimation when the camera had to be trigger was crucial.

**NE:** Verder werden de project doelstellingen uitgediept en hebben we een visueel schema opgebouwd:

**EN:** Next we drew up a visual connection schema:



**NE:** Hierop volgend zijn we opzoek gegaan om een manier om foto’s van de camera te lezen zonder dat de geheugenkaart uit de camera verwijdert wordt. De uitwerking van dit deel van het project kan mogelijk gemaakt worden met de bibliotheek: gphoto2 en de applicatie die hierop gebaseerd is: gtkam. Dit programma kunnen we draaien op een RaspberryPi. Dit is niet in de praktijk getest omdat we het nodige materiaal niet ter beschikking hadden.

**EN:** Here after we started researching on the possiblities for reading the picture from the camera without removing the SDcard from the slot. We found a library which gives us the possibilities for this: gphoto2. We could succesfully run this on a RaspberryPi.

**NE:** Achteraf zijn we gestart met onderzoek hoe we de Arduino’s met elkaar kunnen laten communiceren. Hierbij konden we twee Arduino’s met elkaar laten communiceren waar één Arduino een signaal doorgaf zodat de andere Arduino de foto nam.

**EN:** Afterwards we started investigating on how we need to communicate between several Arduino’s. We were able to succesfull connect one Arduino to another, send & receive data trough a wire.

**File 1 – receiving data**

int analogpin = 0;

int val = 0;

void setup()

{

 Serial.begin(9600);

 pinMode(12,OUTPUT);

}

void loop()

{

 val = analogRead(analogpin);

 Serial.println(val);

 if(val == 0)

 {

   digitalWrite(12,LOW);

 }

 else if(val ==1023)

 {

   digitalWrite(12,HIGH);

 }

}

**File 2 – transmitting signals**

int delayMS = 1000;

int incomingByte = 0;

// the setup function runs once when you press reset or power the board

void setup()

{

 Serial.begin(9600);

 pinMode(3, OUTPUT);

}

// the loop function runs over and over again forever

void loop()

{

 // send data only when you receive data:

 if (Serial.available() > 0) {

         // read the incoming byte:

         incomingByte = Serial.read();

         // say what you got:

         Serial.print("I received: ");

         Serial.println(incomingByte, DEC);

         if(incomingByte == 49)

         {

           digitalWrite(3, HIGH);

           delay(delayMS);

           digitalWrite(3, LOW);

           incomingByte = -10;

         }

 }

 /\*

 Serial.println("High");

 digitalWrite(3, HIGH);

 delay(delayMS);

 Serial.println("Low");

 digitalWrite(3, LOW);

 delay(delayMS);\*/

}