The mini project

- Before or on October 13 team presents mini project proposal to coach

  (Location to be arranged by coach/team)

- Sessions October 18, 20, 25 no lectures, but support in and around e-labs for mini project development

- Session October 27 (LG 1.105) team presents mini project result to coach (demo + mini poster)
  (Location e-labs)

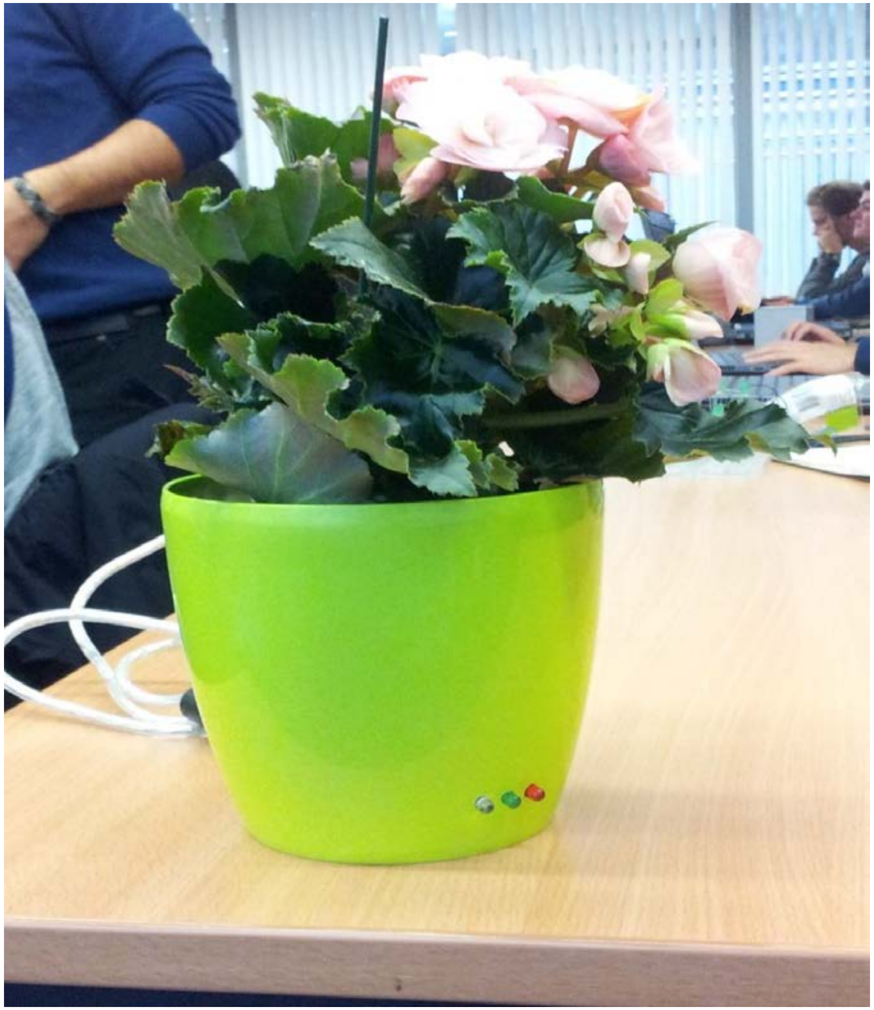
The mini project

- Proposal
  - Indicate context
  - What should it do
  - Input/output -> sensors/actuators

- You are free to choose your project, but:
  - Use Arduino if possible
  - Use at least one sensor and one actuator
  - Report on design choices (transistor, configuration, etc.)
  - Draw correct schematics
  - Present it as a prototype (user experience, packaging) !

Previous examples

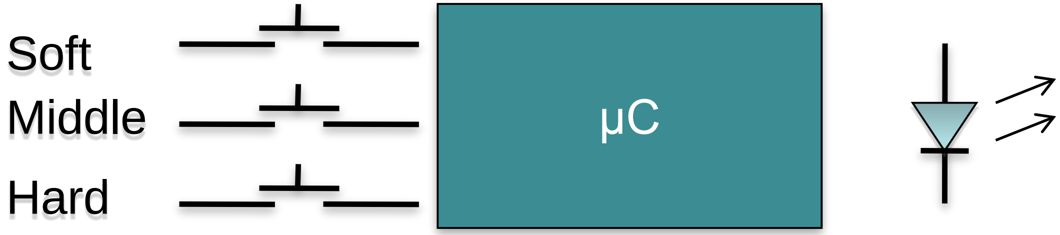
Halloween pumpkin screaming when lid
Is taken off



Plant soil moisture indicator

Example
(Don’t use this one…)

- Egg timer
  - 3 simple sensors, 1 simple actuator



Spaghetti in code, schematics and circuits

Code:

- Use comments /* like this */ or // this
- Use functional variable names
- Use functions

Schematics:

- Draw the complete circuit (correctly!!) (component names, values)

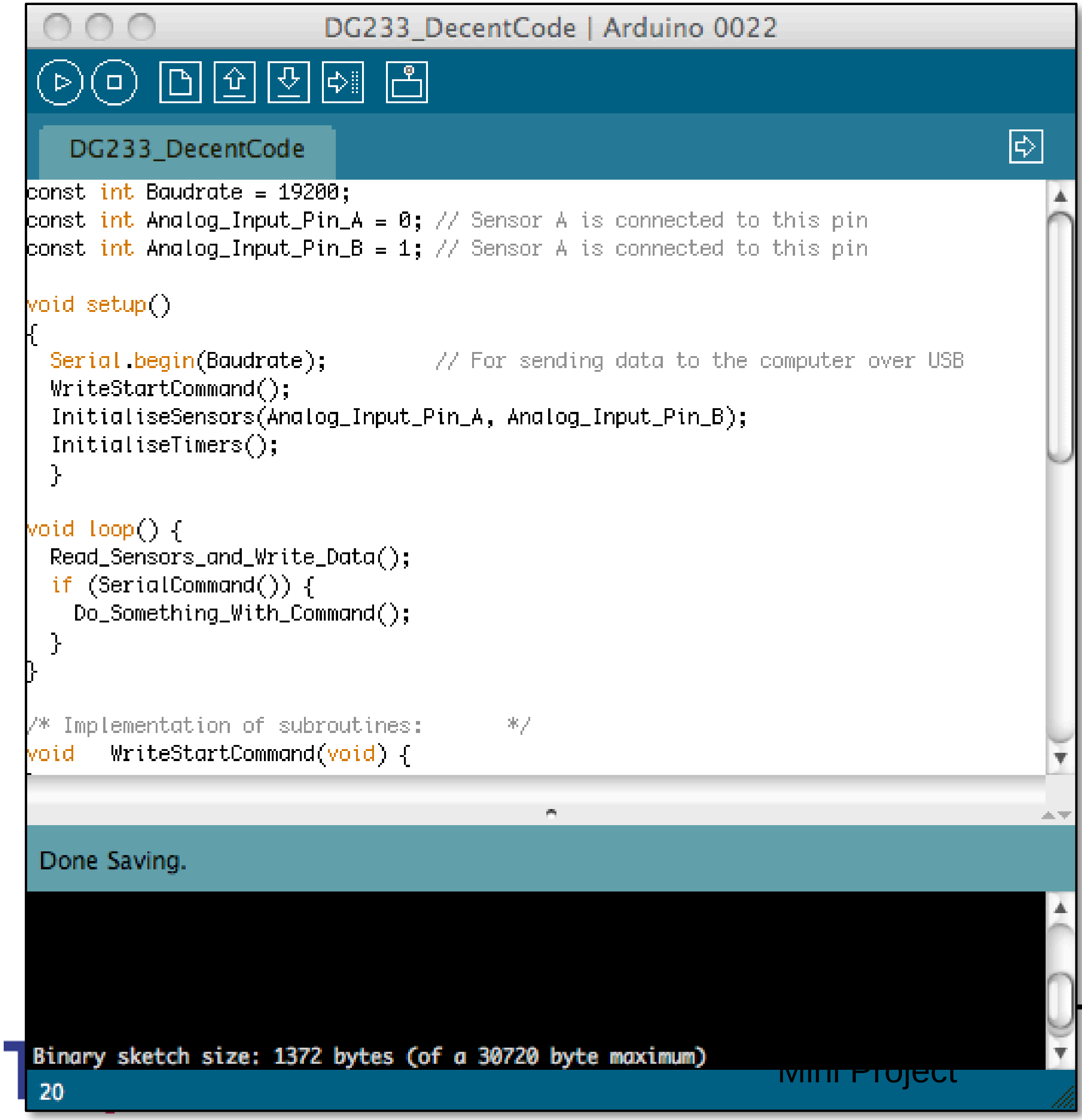
Circuit (soldered or breadboard)

- Should be structured
- Use color coding in wires
- Think about component placement

Otherwise:

- Debugging becomes difficult
- You lack evidence in your report

Code



```
const int Baudrate = 19200;
const int Analog_Input_Pin_A = 0; // Sensor A is connected to this pin
const int Analog_Input_Pin_B = 1; // Sensor A is connected to this pin

void setup()
{
  Serial.begin(Baudrate);         // For sending data to the computer over USB
  WriteStartCommand();
  InitialiseSensors(Analog_Input_Pin_A, Analog_Input_Pin_B);
  InitialiseTimers();
  }

void loop() {
  Read_Sensors_and_Write_Data();
  if (SerialCommand()) {
    Do_Something_With_Command();
  }
}

/* Implementation of subroutines:       */
void   WriteStartCommand(void) {
```

Done Saving.

```
Binary sketch size: 1372 bytes (of a 30720 byte maximum)
```

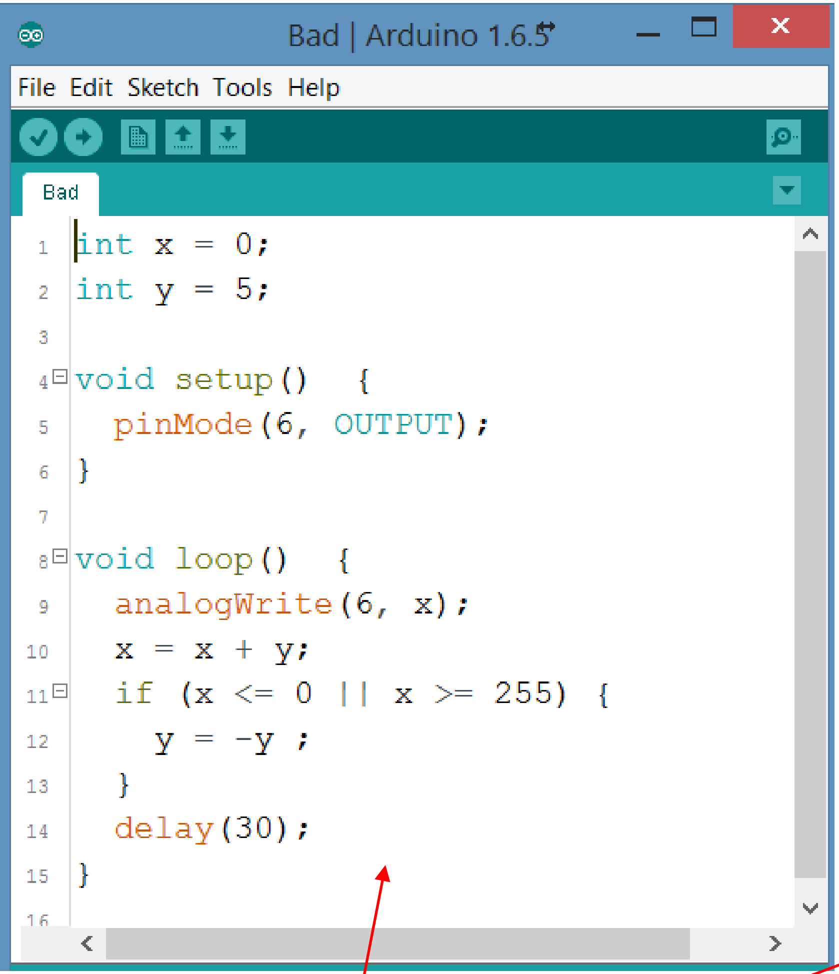
20

Code:

- Clear variable names
- Proper use of local and global variables
- Functions `setup()` and `loop()` are self-explaining
- Proper function names
- Comments (Header)

- Examples
  - Bad example
  - Good example

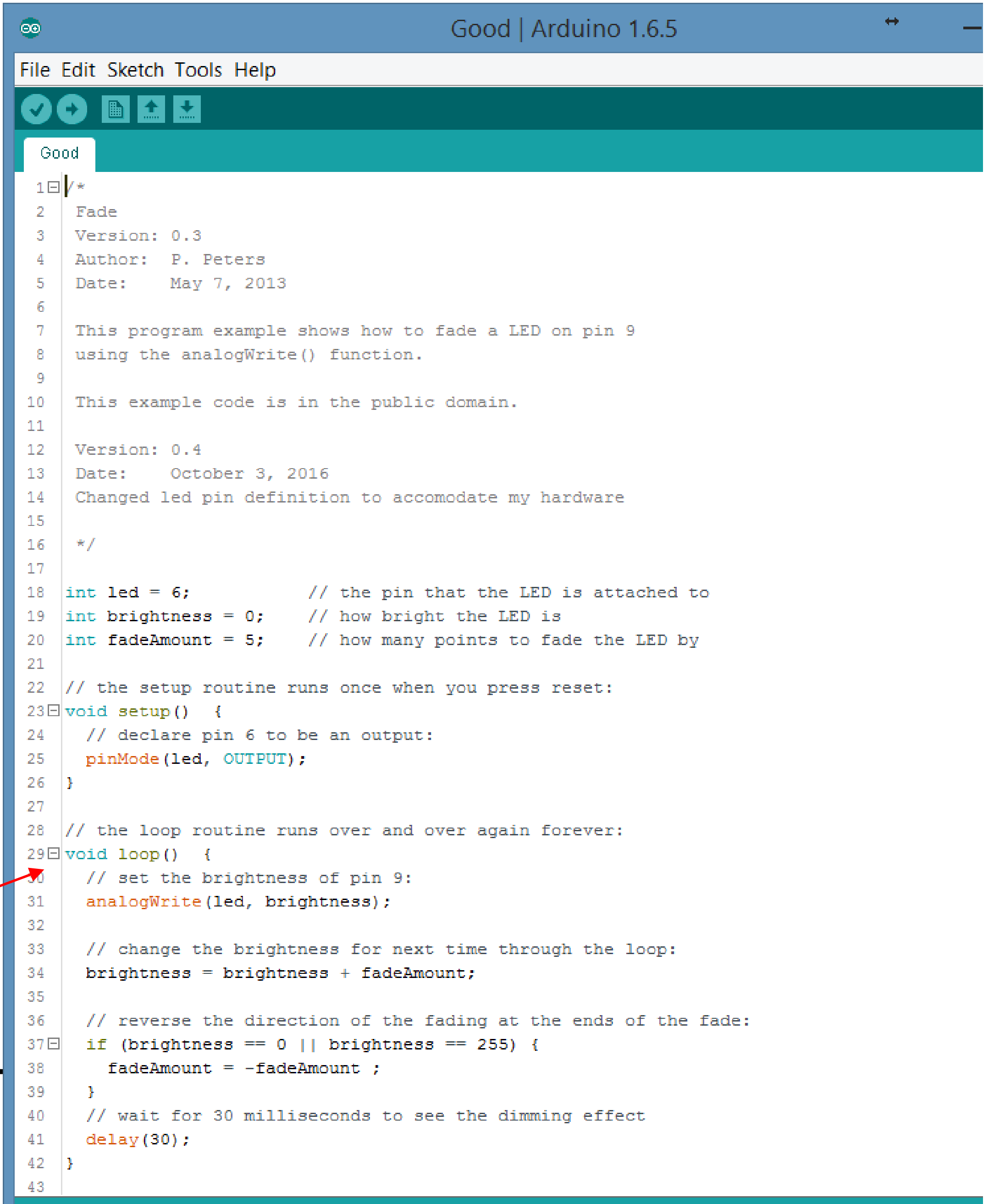
Code



```
int x = 0;
int y = 5;

void setup()  {
  pinMode(6, OUTPUT);
}

void loop()  {
  analogWrite(6, x);
  x = x + y;
  if (x <= 0 || x >= 255) {
    y = -y ;
  }
  delay(30);
}
```



```
/*
 Fade
 Version: 0.3
 Author:  P. Peters
 Date:    May 7, 2013

 This program example shows how to fade a LED on pin 9
 using the analogWrite() function.

 This example code is in the public domain.

 Version: 0.4
 Date:    October 3, 2016
 Changed led pin definition to accomodate my hardware

 */

int led = 6;           // the pin that the LED is attached to
int brightness = 0;    // how bright the LED is
int fadeAmount = 5;    // how many points to fade the LED by

// the setup routine runs once when you press reset:
void setup()  {
  // declare pin 6 to be an output:
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop()  {
  // set the brightness of pin 9:
  analogWrite(led, brightness);

  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;

  // reverse the direction of the fading at the ends of the fade:
  if (brightness == 0 || brightness == 255) {
    fadeAmount = -fadeAmount ;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(30);
}
```

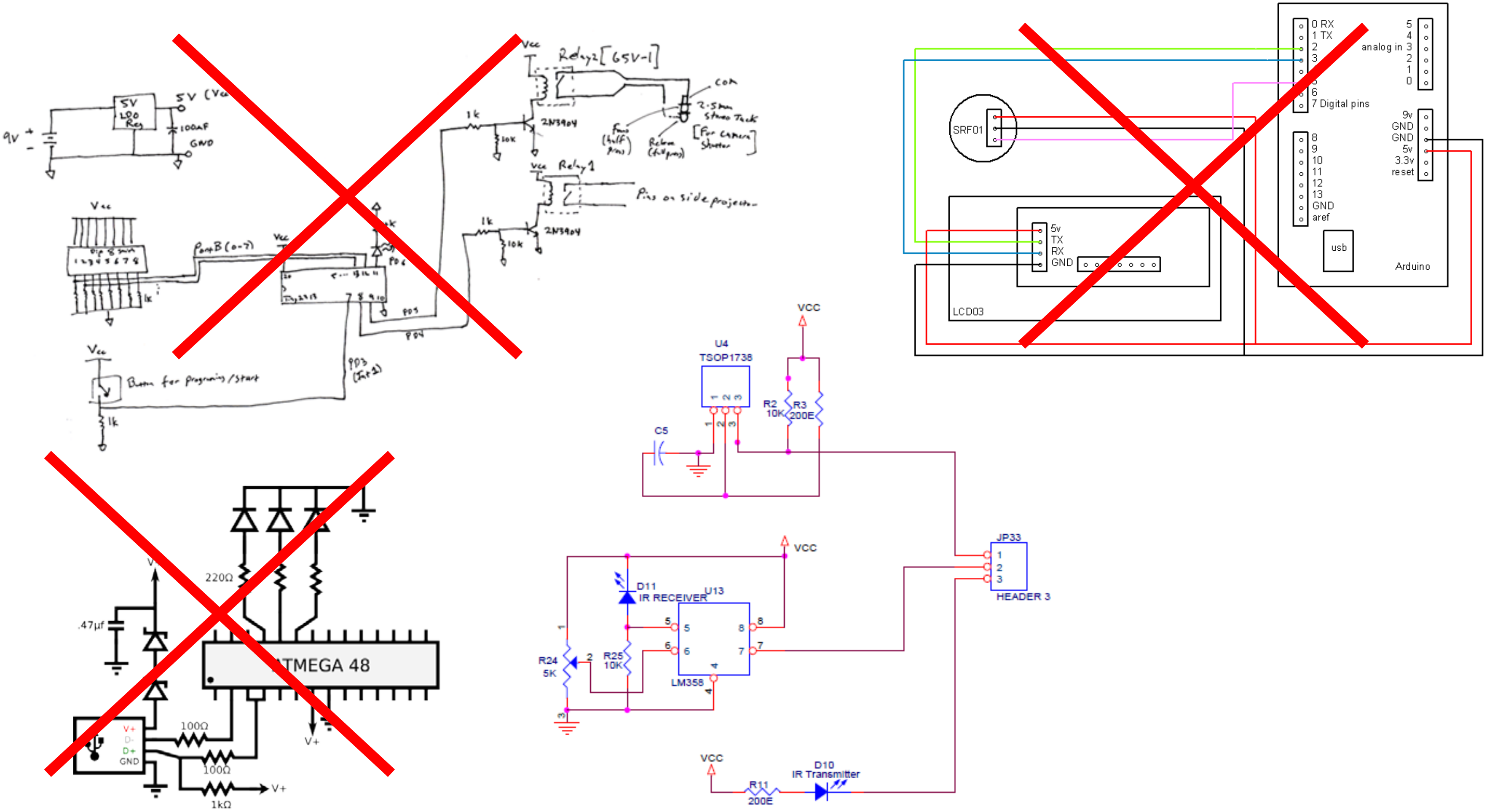
Bad

Good

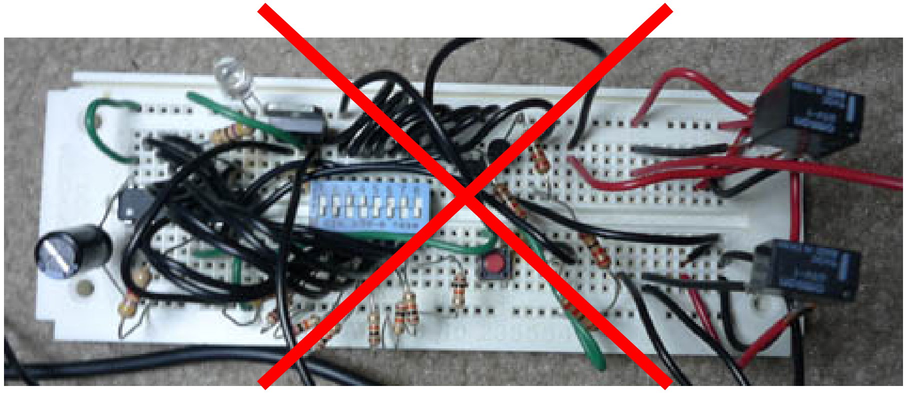
Technische Universiteit
Eindhoven
University of Technology

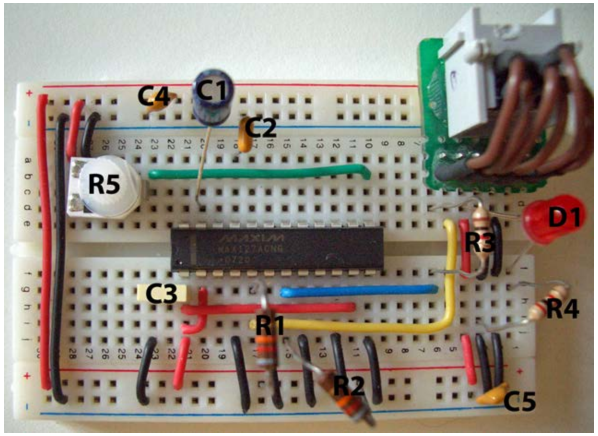
Designed
Intelligence
Group

Schematics


9V
5V LDO Reg
5V (Vcc
100nF
GND
Vcc
Dip 8 switch
1 2 3 4 5 6 7 8
1k
PortB (0-7)
Vcc
Tiny2313
PD6
PD5
PD4
PD3 (Int 1)
Vcc
Button for programing / start
1k
Vcc
Relay2 [G5V-1]
Com
2.5mm Stereo Jack
[For Camera Shutter]
Focus (half press)
Release (full press)
1k
10k
2N3904
Vcc Relay 1
Pins on side projector
1k
10k
2N3904
220Ω
.47µf
ATMEGA 48
V+
D-
D+
GND
100Ω
100Ω
1kΩ
V+
V+
SRF01
0 RX
1 TX
2
3
6
7 Digital pins
8
9
10
11
12
13
GND
aref
5
4
analog in 3
2
1
0
9v
GND
GND
5v
3.3v
reset
usb
Arduino
5v
TX
RX
GND
LCD03
VCC
U4
TSOP1738
R2 10K
R3 200E
C5
VCC
D11 IR RECEIVER
U13
LM358
R24 5K
R25 10K
JP33
1
2
3
HEADER 3
VCC
R11 200E
D10 IR Transmitter

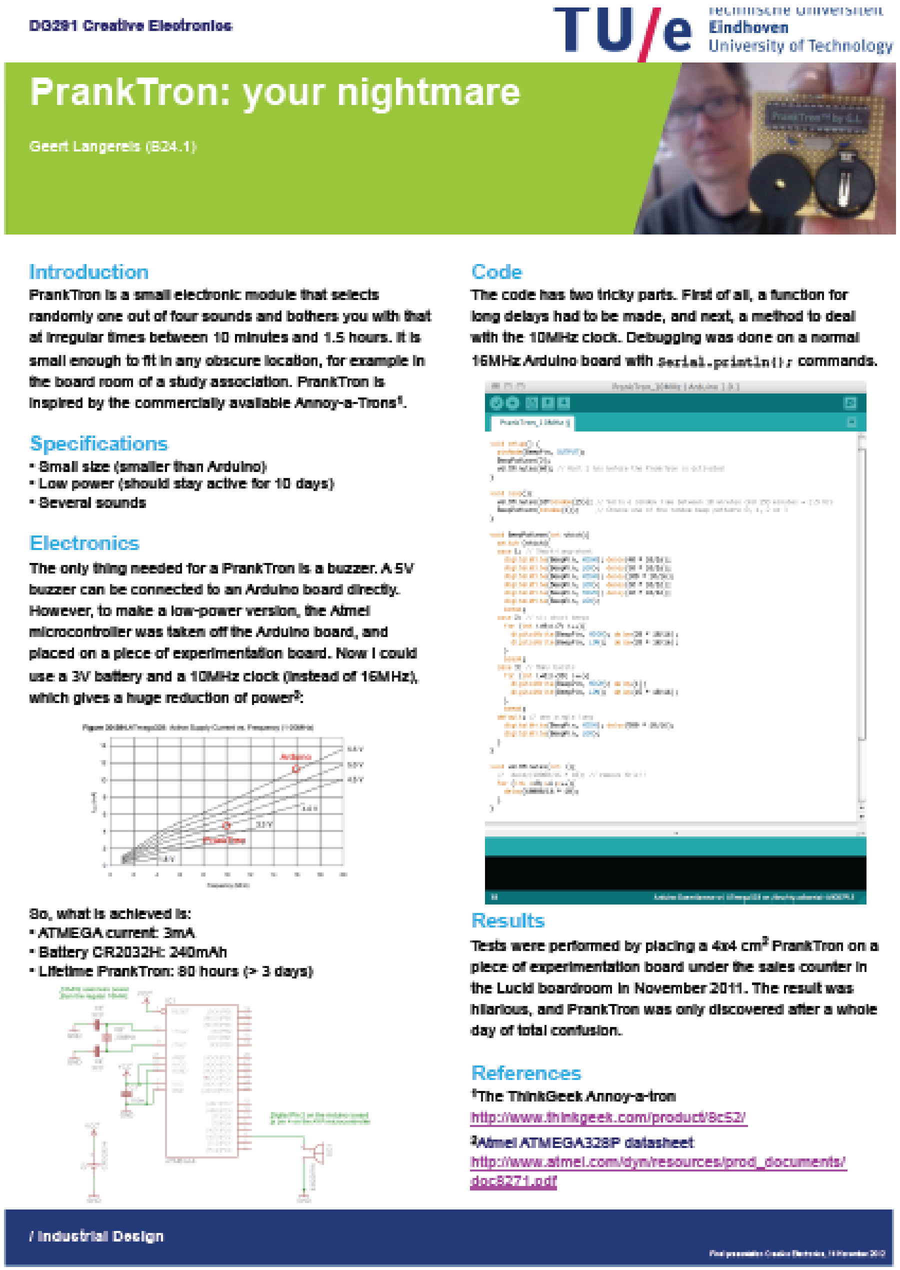
Circuits





Poster

- Explanation
- Schematic
- Code
- Pictures



DG281 Creative Electronics

PrankTron: your nightmare

Geert Langerels (B24.1)

Introduction

PrankTron is a small electronic module that selects randomly one out of four sounds and bothers you with that at irregular times between 10 minutes and 1.5 hours. It is small enough to fit in any obscure location, for example in the board room of a study association. PrankTron is inspired by the commercially available Annoy-a-Trons[1].

Specifications

- Small size (smaller than Arduino)
- Low power (should stay active for 10 days)
- Several sounds

Electronics

The only thing needed for a PrankTron is a buzzer. A 5V buzzer can be connected to an Arduino board directly. However, to make a low-power version, the Atmel microcontroller was taken off the Arduino board, and placed on a piece of experimentation board. Now I could use a 3V battery and a 10MHz clock (instead of 16MHz), which gives a huge reduction of power[2]:

So, what is achieved is:

- ATMEGA current: 3mA
- Battery CR2032H: 240mAh
- Lifetime PrankTron: 80 hours (> 3 days)

Code

The code has two tricky parts. First of all, a function for long delays had to be made, and next, a method to deal with the 10MHz clock. Debugging was done on a normal 16MHz Arduino board with `Serial.println();` commands.

Results

Tests were performed by placing a 4x4 cm^2 PrankTron on a piece of experimentation board under the sales counter in the Lucid boardroom in November 2011. The result was hilarious, and PrankTron was only discovered after a whole day of total confusion.

References

[1]The ThinkGeek Annoy-a-tron
http://www.thinkgeek.com/product/8c52/

[2]Atmel ATMEGA328P datasheet
http://www.atmel.com/dyn/resources/prod_documents/doc8271.pdf

/ Industrial Design

Final presentation Creative Electronics, 14 November 2012