Week 11: Output devices

15.4. 2015

This week we will learn about output devices.

The agenda: http://academy.cba.mit.edu/classes/output_devices/index.html

```
electrical safety
   ~1 mA: OK
   ~10 mA: shock, contraction
   ~100 mA: fibrillation
   body: M ohm external, k ohm internal
   skin depth: ~1/f^(1/2)
   dielectric breakdown: ~kV/mm
   supply capacitors
   inductive flyback
   protection diodes
   connector polarization, orientation

power supplies

RGB LED
   PWM
   hello.RGB.45.cad board components traces interior C makefile video

LED array
   Charlieplexing
   hello.array.44.cad board components traces interior C makefile
video
   hello.array.44.2.cad board components bottom vias top interior C
makefile video

LCD
   44780
   hello.LCD.44.cad board components traces interior C makefile video

video
   RS170, NTSC, PAL, SECAM
   75 ohm impedance
   hello.video.44.cad board components traces interior C makefile
video
   asm
   SPI TVout thinner client

speaker
   MOSFET
      hello.speaker.45.cad board components traces interior C
makefile video
      wavetable
   gate resistors
   low-pass filters
   class D amplifiers
   parallel FETs

DC motor
   H-bridge
   hello.H-bridge.44.cad board components traces interior C makefile
video
   torque, power, efficiency
   gearhead, outrunner, brushless, vibration
```

servo
   hello.servo.44.cad board components traces interior
   C (hardware PWM) makefile C (two-channel, software PWM) makefile
video

stepper motor
   unipolar
      hello.stepper.44.cad board components traces interior
      wave makefile full makefile half makefile video
   bipolar
      hello.stepper.bipolar.44.cad board components traces interior
      full makefile video
   electro-permanent
   wire identification
   step PWM, duration

motor control
   Rohm
   Freescale
   Allegro
   Pololu
   TinyG ChiliPeppr

shape memory

piezo

The assignment for this week: to add an output device to a microcontroller board you've designed and program it to do something

Class:

Power supply depot

RGB / LED – changing colors
http://www.digikey.com/product-detail/en/CLV1A-FKB-CJ1M1F1BB7R4S3/CLV1A-FKB-CJ1M1F1BB7R4S3CT-ND/1987488

CREE
3 LEDs – red green blue
No standard for orientation

C code – macros that talk to the pin
Pins from the AVR can souce up to ? mamp

Pulse with modulation

Term the LED on and off

LED – more time bright

LED Arrays
http://www.digikey.com/product-detail/en/LTST-C150CKT/160-1167-1-ND/269239

Charleplexing – if you have an nsqLED, this lets you get by with the square root of that
http://www.maximintegrated.com/en/app-notes/index.mvp/id/1880

Every pin goes to both rows and columns

Each pin simulataneously is driving a row and a column

LCD
http://www.digikey.com/product-search/en?WT.z_header=search_go&lang=en&site=us&keywords=67-1781-ND&x=0&y=0

The datasheet tells you nothing
LCDs use a nasty chip HD44... Hitachi chip
http://academy.cba.mit.edu/classes/output_devices/44780.pdf

board:
http://academy.cba.mit.edu/classes/output_devices/LCD/hello.LCD.44.png

code:
http://academy.cba.mit.edu/classes/output_devices/LCD/hello.LCD.44.cad

components
http://academy.cba.mit.edu/classes/output_devices/LCD/hello.LCD.44.jpg

Handshaking – then it works
The routines to talk to it are very fussy

Video
http://www.digikey.com/product-search/en?lang=en&site=US&WT.z_homepage_link=hp_go_button&KeyWords=CP-1453-ND&x=0&y=0

Makes a video test signal on an LCD

Video game example – James Fletcher:
http://fabacademy.org/archives/2014/students/fletcher.james/week12.html -

Arduino TV output
Analogue video ... RS170:
http://academy.cba.mit.edu/classes/output_devices/RS-170.jpg
Lines – need to be correct

Video:
http://academy.cba.mit.edu/classes/output_devices/video/hello.video.44.mp4

Board:
http://academy.cba.mit.edu/classes/output_devices/video/hello.video.44.jpg

Speaker
http://www.digikey.com/product-detail/en/PSR-23F08S-JQ/458-1124-ND

2types of mosfets
http://www.digikey.com/product-detail/en/PSR-23F08S-JQ/458-1124-ND

Pulse with modeling – variying the on versus off
Kiloherz – Megaherz
Average time, on – off
Varying the time that it goes to the speakers

2 ways to pulse

Video:
http://academy.cba.mit.edu/classes/output_devices/speaker/hello.speaker.45.mp4

Vary the pw8 to make the sound

Motors
http://www.jameco.com/webapp/wcs/stores/servlet/Product_10001_10001_1810099_-1

Hbridge
http://www.digikey.com/product-detail/en/A4953ELJTR-T/620-1428-1-ND/2765622

Mosfets...

Problem – a significant one. For the end mosteps...

Video: http://academy.cba.mit.edu/classes/output_devices/DC/hello.H-bridge.44.DC.mp4

Board: http://academy.cba.mit.edu/classes/output_devices/H-bridge/hello.H-bridge.44.png

Components: http://academy.cba.mit.edu/classes/output_devices/H-bridge/hello.H-bridge.44.jpg

The regulator has the capacitor

PWM software

Varying the current to the motors

DC motors has gears that can slow it down

Servo:
http://www.hobbyking.com/hobbyking/store/__84__189__Servos_and_parts-All_Servos.html
Video:
http://academy.cba.mit.edu/classes/output_devices/servo/hello.servo.44.mp4

Stepper motor:
http://www.jameco.com/webapp/wcs/stores/servlet/Product_10001_10001_2138812_-1
unipolar:
http://academy.cba.mit.edu/classes/output_devices/stepper/hello.stepper.44.mp4
bipolar:
http://academy.cba.mit.edu/classes/output_devices/stepper/hello.stepper.bipolar.44.mp4

Motor actuator drivers: http://www.rohm.com/web/global/groups/-/group/groupname/Motor%20~%20Actuator%20Drivers

Number of vendors

Rhom: http://www.rohm.com/web/global/groups/-/group/groupname/Motor%20~%20Actuator%20Drivers

Freescale: http://www.rohm.com/web/global/groups/-/group/groupname/Motor%20~%20Actuator%20Drivers

Pololu: https://www.pololu.com/category/9/motion-control-modules

Allegro: http://www.allegromicro.com/Products/Motor-Driver-And-Interface-ICs.aspx

TinyG: http://synthetos.myshopify.com/products/tinyg

ChiliPepper: http://chilipeppr.com/tinyg

Shape memory: http://fab.cba.mit.edu/classes/MIT/863.10/people/jie.qi/jieweek10.html

Piezo: http://www.jameco.com/webapp/wcs/stores/servlet/Product_10001_10001_1956784_-1

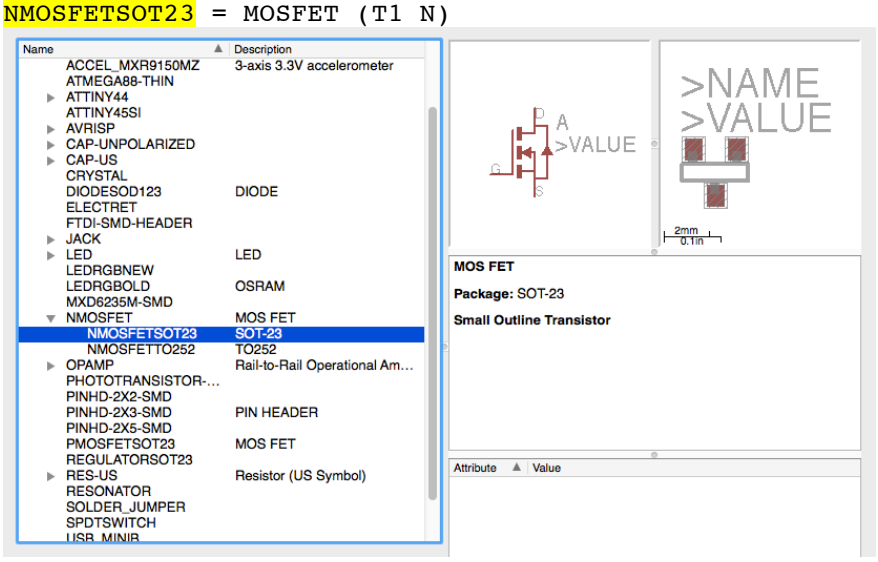
Assignment:

A board that can sense changes in natural daylight. When the light increases a sound can be produced, that increases in intensity with the light exposure.

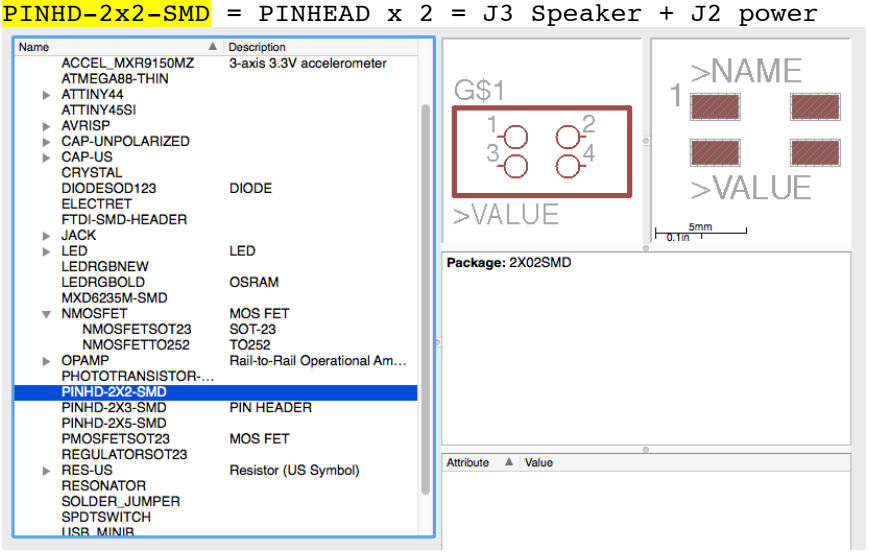
- measuring the light (photosensor)
- driving the speaker (mosfet)
- energy (battery – 9v)
- driving the activities (microcontroller)

Speaker board

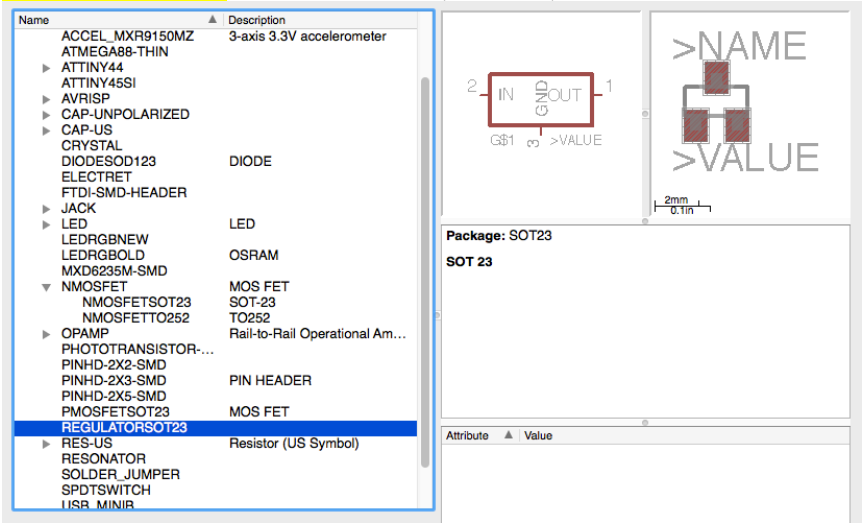
NMOSFETSOT23 = MOSFET (T1 N)



PINHD-2x2-SMD = PINHEAD x 2 = J3 Speaker + J2 power



REGULATORSOT23 = Regulator (IC2 5V)

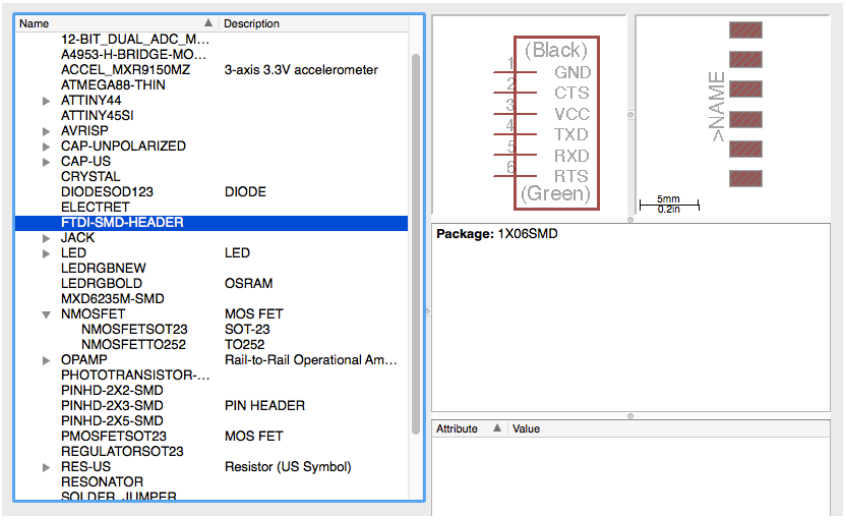


Combined with light board

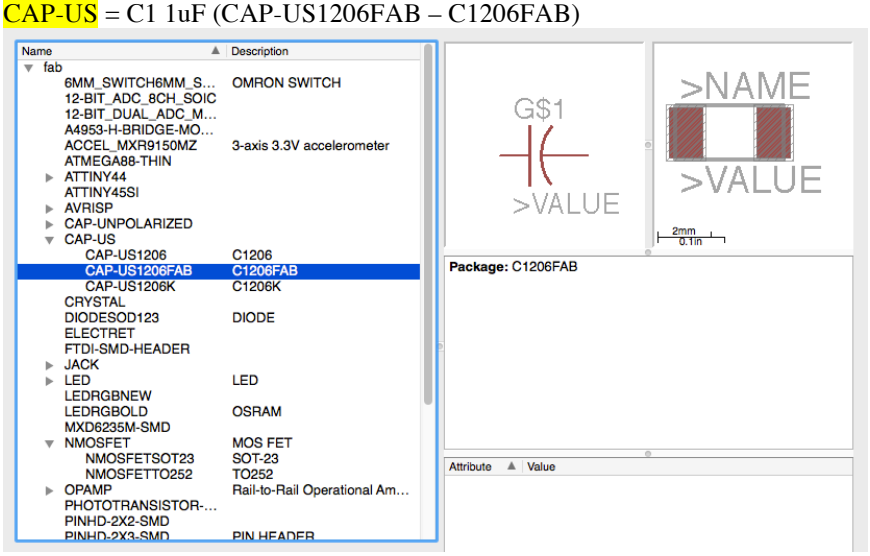
(not neccessary, as we are not communicating with the computer)

FTDI-SMD-HEADER = J2 FDTI

Allows the computer to communicate with the board, power source from the computer

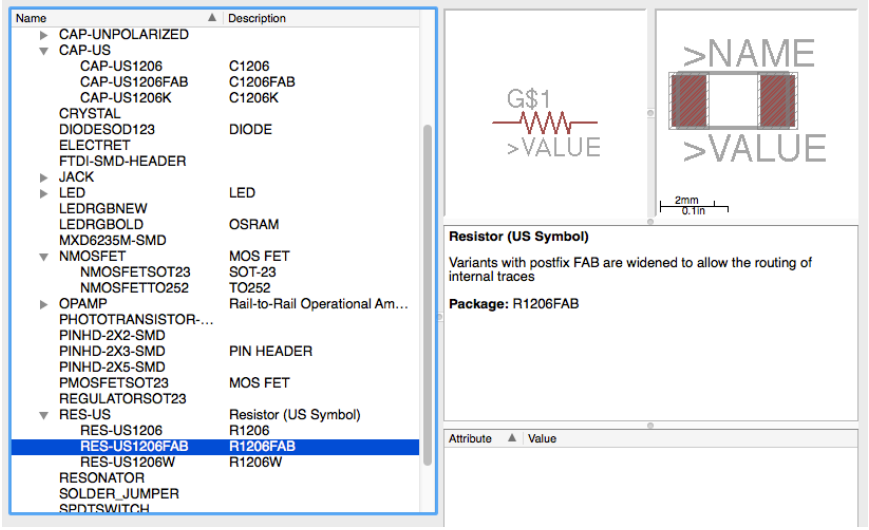


CAP-US = C1 1uF (CAP-US1206FAB – C1206FAB)



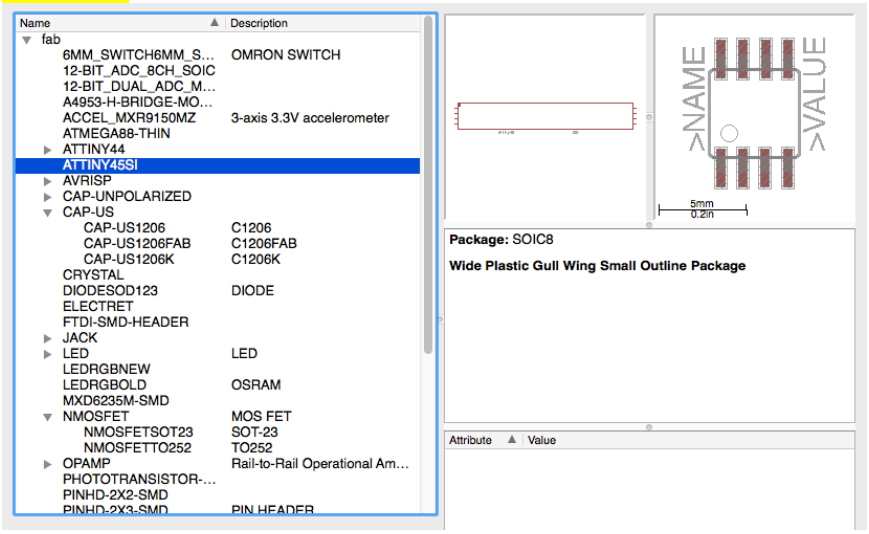
Stores energy, power reserve, but can release energy very quickly.
C1 – prevents or helps mediate the problems of brown-out.
Brown-out, power failure of sort, or a little dip in the current.

RES-US = R1 10k (RES-US1206FAB – R1206FAB)



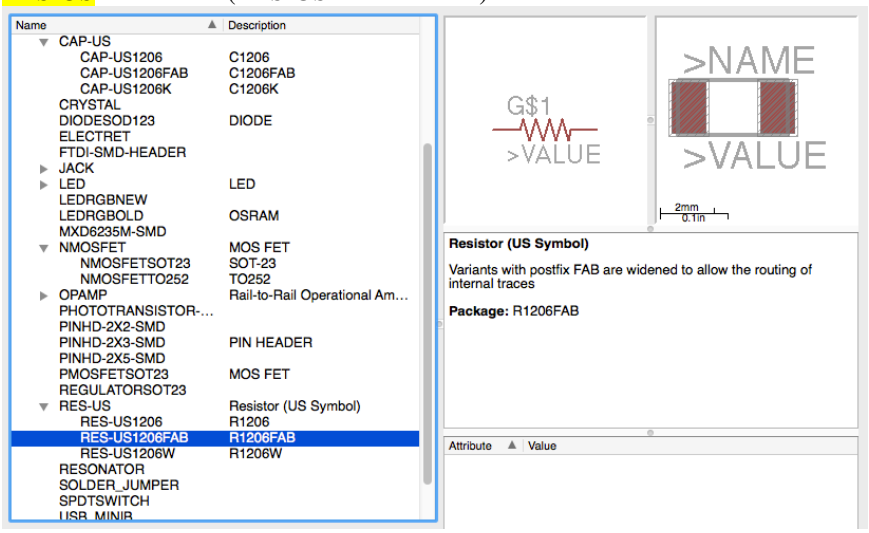
R1 – a pull-up resistor, pulls it up to to 5v – when connected the voltage on the line is pulled up to 5v. Borderline 1.2 v, the microcontroler switches off. IS needed on the reset line to make sure the microcontroller does not reset.

ATTINY45 = IC1 t45



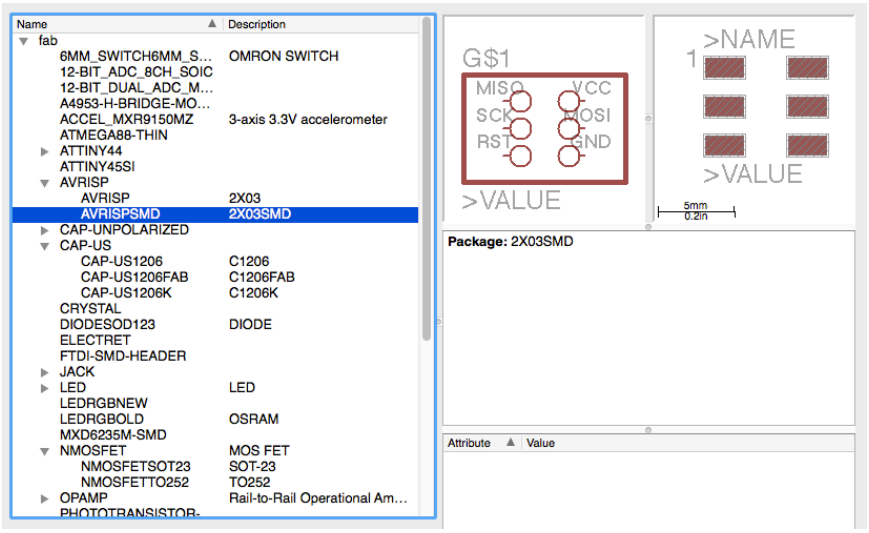
Microcontroller - Receives the voltage, it is the only thing it can read.
Adding, subtracting, shoving infmormation around, changing course of action. Sets of rules. It acts upon messages until told otherwise.

RES-US = R2 49.9k (RES-US1206 – R1206) = R2 49.9k



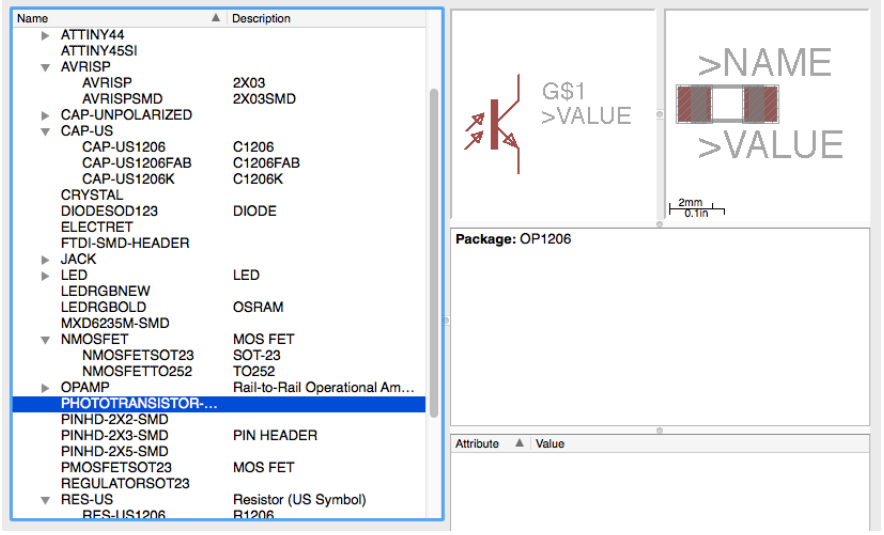
Limits the amount of current – with division of voltage

AVRISPSMD = J1 ISP (in system programming)



Connector (allowing the board to be programmed, through a cable)

PHOTOTRANSISTOR = T1



Acts like a gate

Reset line – is needed to put the programmes on the microcontroller, puts the microcontroller in programming mode.

Made notes with Bas on the design.

Saving the images:

In layers – Select all – OK

In layers – Select bottom layer.

Chose  - Rect, and draw a square around the board circuits – approx 1 grid outside the square around it.
Possible to click info, double click the image and change the layer.

In layers – select 20 Dimensions – and export image as monochrome (resolution: 1200; image size: 2204 x 810): Name of square.

When the board is finally defined the traces are exported. Before exporting turn off all layers except the top one. Export as an image - monochrome (resolution: 1200; image size: 2204 x 810): Name of board.

The images:

Light-Speaker-Square and Light-Speakers-Traces were opened in Photoshop.

The Light-Speaker-Square was filled with (the bucket) with white color.

Assembling of the components:

Milling of the board:

Tape board to table

Turn on machine – push view to move chuck to left end of working area

In fabmodules open image –

Output format - Roland mill .rml
Process - PCB trace (1/64)

Turn off view

Calculate
Move ti xmin ymin
Adjust smin ymin settings – 2-2

If right – losen mill end and lower to surface of board
Check loaction of tool
Send if ok

The bug in Fabmodules

I had to adjust the cutting depth – from 1.0 to 1.5

To cut the outline, first press view and change the tool to 1/32. Then load the image – outline and set to Roland Mill (.rml) and ouline - calculate.

Press xmin ymin to bring the tool to the right starting position, then lower the tool to the surface of the board and fix. Calculate again and press send.

Soldering

Some adventure with soldering – write about mistakes, connecting between circuits with wire and glueing.

A wire connects to a connector and a speaker.

Preparation for programming

The board was connected to the programmer board and the speaker and battery was also connected to the board, with a connector to their relevant pins. The wires need to be connected to ground and vcc.

Arduino

In Arduino I started a new sketch and saved it as sketch_output_board

3 components to set in the Arduino – the board, processor, select the right clock source (external-internal).

Board – Attiny
Processor – 45
Clock – internal 8Mhz (datasheet, Neil's lecture) 1 Mhz use less energy than 8 Mhz

The output will be digitally communicated from the microcontroller

The speaker is connected to the battery and the mosfet (acts as a switch). Microcontroller can open or close it. The other side of the mosfet is connected to ground. When the microcontroller closes the switch then the speaker (magnet and around it is a coil, with a gap in between. This wire is connected to the coin. The magnets contracts and there is a pull of the surface of the speaker, moving air. This creates a tone).

Herz – number of vibrations pr second

1000 x second

Code now:

```
int analogPin = 3;     // potentiometer wiper (middle terminal)
connected to analog pin 3

                       // outside leads to ground and +5V

int val = 0;           // variable to store the value read

int digitalPin = 1;     // potentiometer wiper (middle terminal)
connected to analog pin 3


void setup() {

pinMode(1, OUTPUT);

}

void loop() // run over and over
{

 //  val = analogRead(analogPin);    // read the input pin

  digitalWrite(1, HIGH);
  delayMicroseconds(100); // Approximately 10% duty cycle @ 1KHz
  digitalWrite(1, LOW);
  delayMicroseconds(1000 - 100)
```

}

Compile and upload

Connection problems with output board

Measure voltage of battery

Fix soldering

Tools – burn bootloader

Confirmed output!

Melody test

Syntax – tells what you how to use the function (conditioned)

Parameters – tell what the different syntax items mean

Returns – function can return information

The melody can be seen as a bucket with multiple slots
The name of the variable is melody
Square signifies (array) – not a normal container, but contains multiple containers.