Welcome to C programming for devices.

Arduino 1.

Topics for today:

* Types in Arduino C compiler.
* The Arduino Hardware.
* Analog to digital converter
* PWM (digital to analog)
* Serial.print
* Timer

Types in C:

* **boolean** (8 bit) - simple logical true/false
* **byte** (8 bit) - unsigned number from 0-255
* **char** (8 bit) - signed number from -128 to 127.
* **unsigned char** (8 bit) - same as 'byte'; Use 'byte' for reasons of clarity
* **word** (16 bit) - unsigned number from 0-65535
* **unsigned int** (16 bit)- the same as 'word'. Use 'word' for clarity and brevity
* **int** (16 bit) - signed number from -32768 to 32767.
* **unsigned long** (32 bit) - unsigned number from 0-4,294,967,295.
* **long** (32 bit) - signed number from -2,147,483,648 to 2,147,483,647
* **float** (32 bit) - signed number from -3.4028235E38 to 3.4028235E38.
  Floating point on the Arduino is not native;
  the compiler has to jump through hoops to make it work.
  If you can avoid it, you should.

Defining a type in C:

```
#define testtype byte

testtype x=1;
testtype y=2;
```

Variables in C:

Declare:

```
int x;
int y;
```

Assign:

```
x = 1;
y = 1;
```

Declare and assign:

```
int z = 3;
```

Variables in C:

* Global variables:
  * Declared out of any function
  * Working all over in the program
* Local variables:
  * Working in the function where it is declared

A standard C program

```
void main()
{
// The main function is always where the
program starts


}
```

An Arduino C program

```
void setup()
{
    // this function runs first and only ones
}

void loop()
{
    // this is where your program is written
}
```

Control the program:

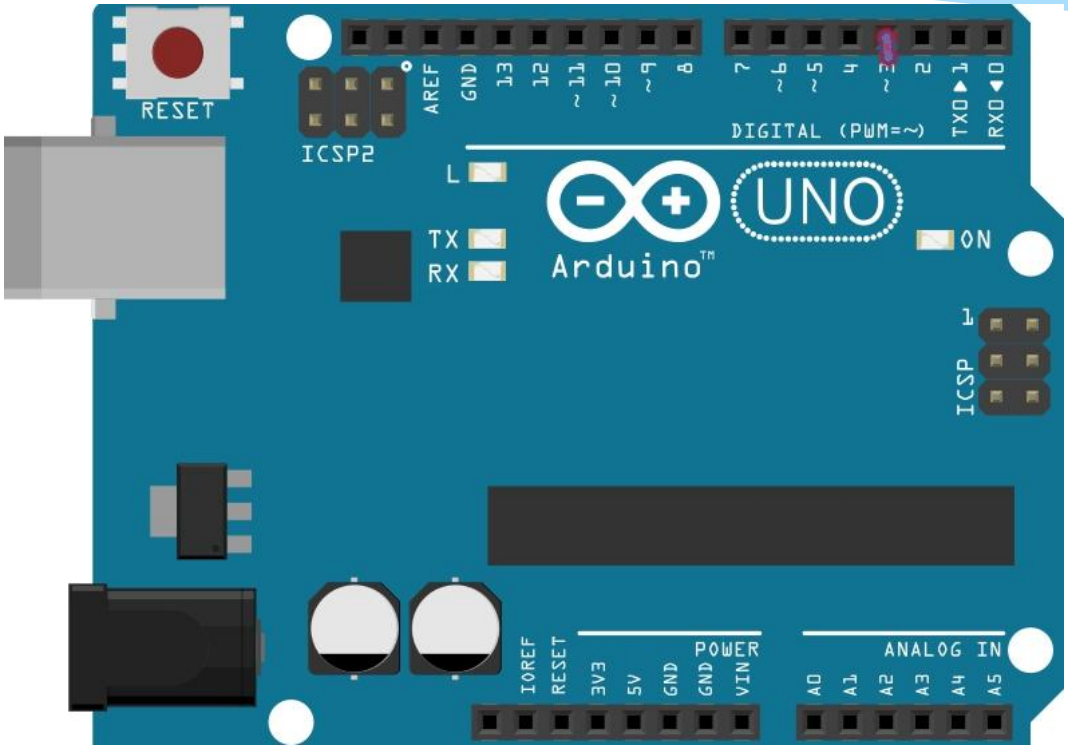
* The if statement:

```
If (statement is true)
    execute this line (otherwise don't)
```

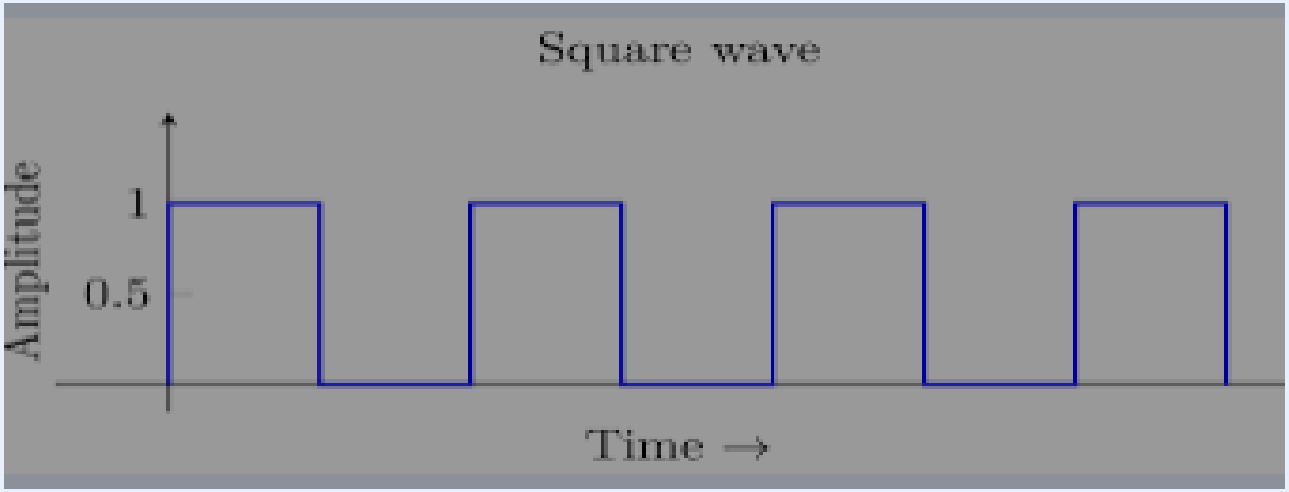
* Loops:

```
for – while – do while
```

Arduino UNO:

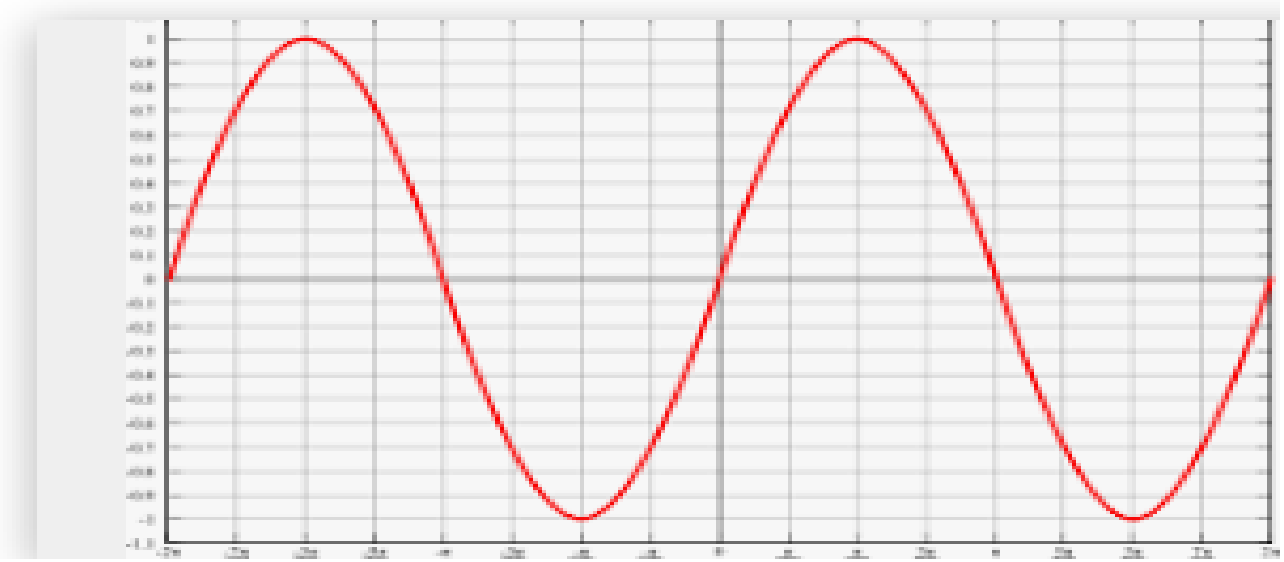

RESET
ICSP2
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM=~)
L
TX
RX
Arduino™
UNO
ON
ICSP
POWER
ANALOG IN
IOREF
RESET
3V3
5V
GND
GND
VIN
A0
A1
A2
A3
A4
A5

Digital square wave

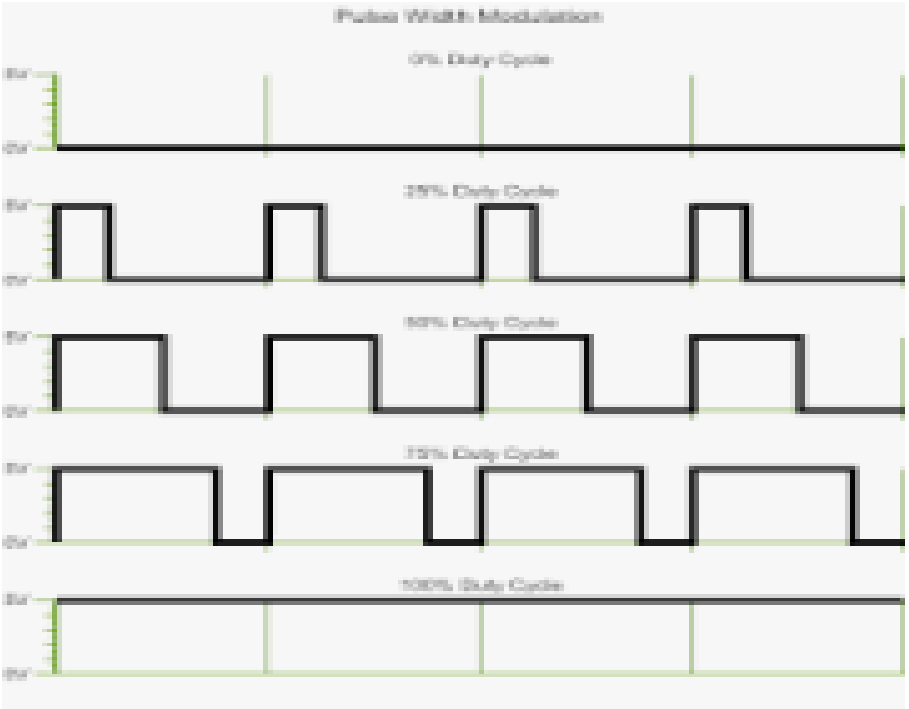


How can I visualize a frequency as square wave? - Mathe...
mathematica.stackexchange.com

Analog signal



Pulse Width Modulation



Pins and the use of them.

0, 1	Serial communication.
2,3,4,5,6,7,8,9,10,11,12,13	Digital in/out.
3,5,6,9,10,11	PWM out. (Analog out)
A0,A1,A2,A3,A4,A5	Analog input

Arduino:

Peripheral Features

– Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode

– One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode

– Real Time Counter with Separate Oscillator

– Six PWM Channels

– 6-channel 10-bit ADC in PDIP Package

– One Programmable Serial USART

– One Byte-oriented 2-wire Serial Interface (Philips I2C compatible)

Arduino:

– One On-chip Analog Comparator
– Interrupt and Wake-up on Pin Change
• Special Microcontroller Features
– Power-on Reset and Programmable Brown-out Detection
– Internal Calibrated Oscillator
– External and Internal Interrupt Sources
– Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and

Extended Standby
• I/O and Packages
– 23 Programmable I/O Lines

Digital output pin:

Digital output:

```
// Pin which controls the onboard led.
int LED = 13;

// Set the pinmode as output.
pinMode(LED, OUTPUT);
digitalWrite(LED,HIGH);
```

Port vs Pin programming

```
digitalWrite(9,HIGH);
digitalWrite(9,LOW);          // 57 clock cycles
bitSet(PORTB,1);
bitClear(PORTB,1);
bitWrite(PORTB,0,value);      //  2 clock cycles
```

Port programming

Port B har pin **B0 to B5 8,9,10,11,12,13**

Port C har pin **C0 to C5 A0,A1,A2,A3,A4,A5**

Port D har Pin **D0 to D7 0,1,2,3,4,5,6,7**

```
bitSet(   PORTB, 2) :
bitClear( PORTB, 2) ;
PORTB       = B111111;
int    tal =   PINB ;
```

Digital input without pullup:

```
// Pin which controls INPUT
int B = 4;
 pinMode( B, INPUT );
 if ( digitalRead(B)==HIGH ){
 … …
 }
```

Digital input with pullup:

```
    int  B = 4;

// Set the pinmode as input.
    pinMode( 4, INPUT);

//activate built-in pull-up resistor.
    digitalWrite(B, HIGH);

//                          or simply
// pinmode, and build-in pull-up.
    pinMode( 4, INPUT_PULLUP);
```

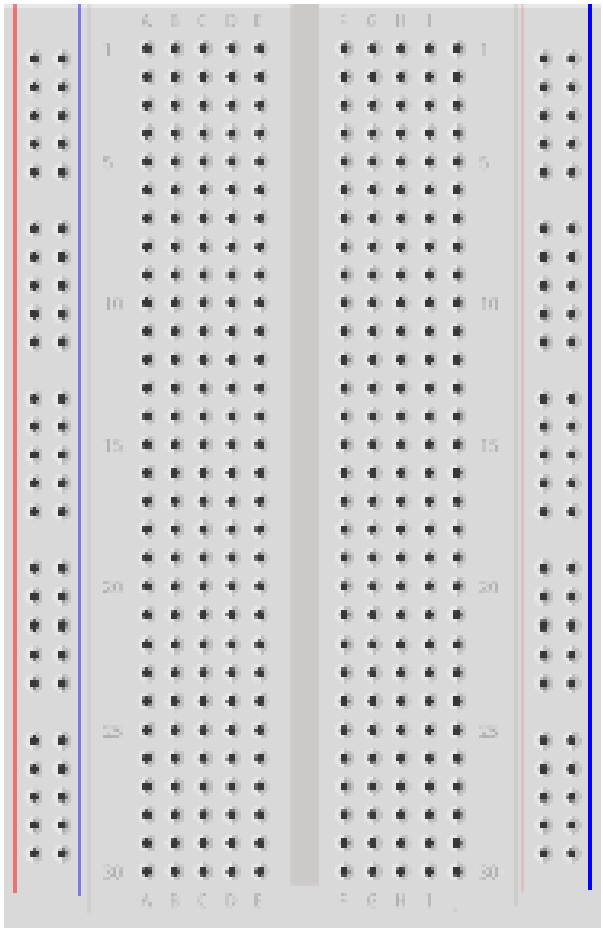
Analog input:

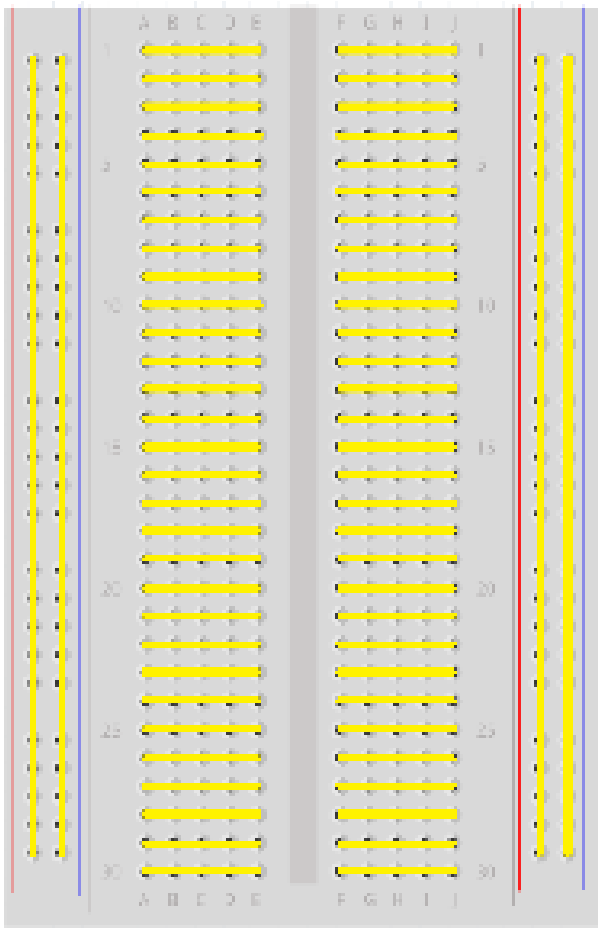
```
// Analog input pin from the potentiometer.
    int analogInPin = A0;
    sensorValue     = analogRead( analogInPin );
// wait 2 milliseconds before the next loop
// for the analog-to-digital converter to settle
// after the last reading:
```

Analog output (PWM):

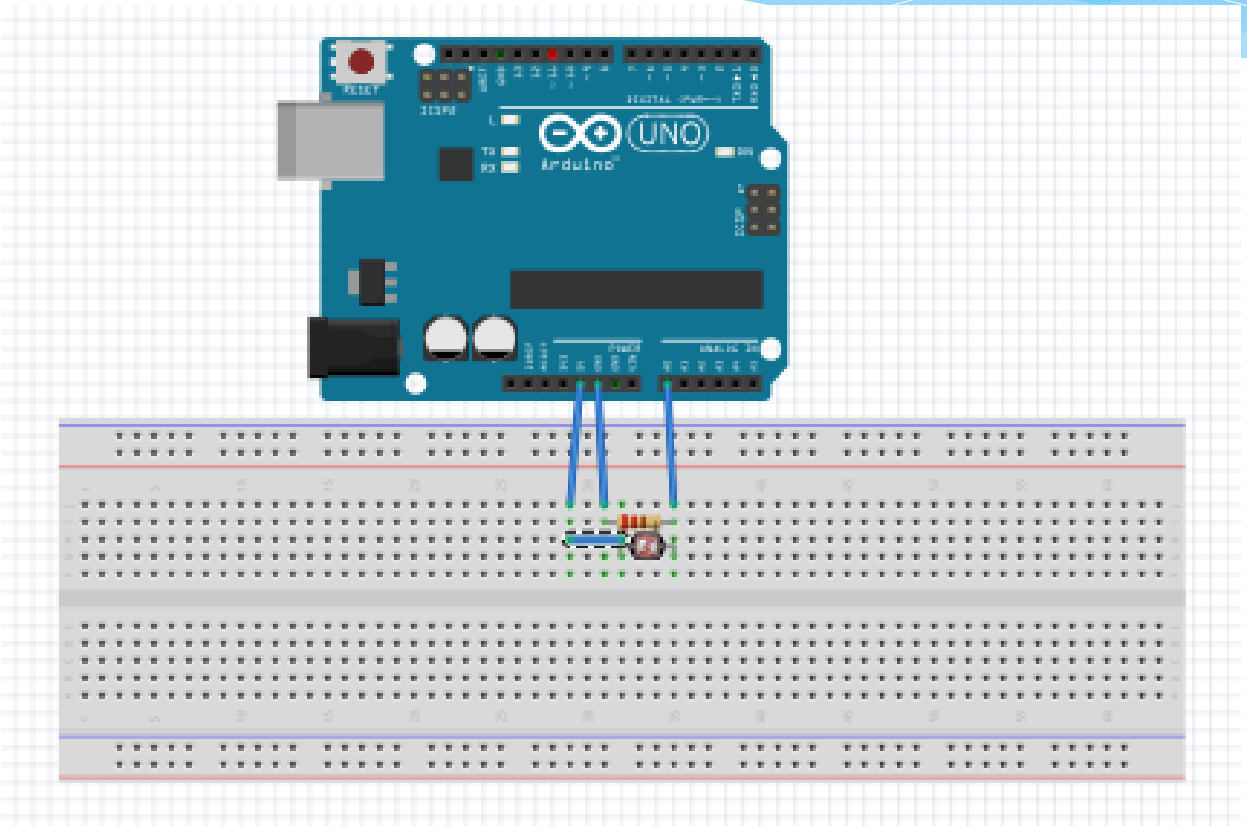
```
// Analog output pin where LED is attached to :
    int analogOutPin = 9;
    anValue  = analogRead( analogInPin );
    outValue = map( anValue, 0, 1023, 0, 255 );
    analogWrite( analogOutPin, outValue );
```

Breadboard





Measure the light:



Setup function

```
void setup() // analog reading
{
    // initialize serial com at 9600 bps:
    Serial.begin(9600);
    // Analog input pin that reads the light
    int lightPin = A0;
}
```

Loop function

```
void loop()
{
    int lightvalue;
    int outputValue;
    lightvalue = analogRead(lightPin);
    delay(2);
    outputValue = map(lightValue, 0, 1023, 0,255);
    Serial.println(outputValue);
}
```

Timer setup function

```
int second=0, minut=0;          // Global variables

void setup()                    // timertest
{
    Serial.begin(9600);         // initialize serial
                                // communication
    noInterrupts();             // disable all interrupts
    TCCR1A = 0;                 //
    TCCR1B = 0;                 //
    timer1_startvalue = 3036; // preload timer 65536-
                                // 16MHz/256/1Hz
    TCNT1 = timer1_startvalue;// preload timer
    bitSet(TCCR1B, CS12);       // 256 prescaler
    bitSet(TIMSK1, TOIE1);      // enable timer1 overflow
                                // interrupt
    interrupts();               // enable all interrupts
}
```

Loop function:

```
void loop()
{
    Serial.print(minut);
    Serial.print(':');
    if(sekund<10) Serial.print('0');
    Serial.println(sekund);
}
```

Interrupt routine:

```
ISR(TIMER1_OVF_vect)                    // interrupt service
                                        // routine
{
    TCNT1 = timer1_startvalue; // reload timer1
    sekund++;                           //
    if (sekund > 59)                    //
    {
      sekund = 0;
      minut++;
    }
}
```

Assignment A1

1. Make a program with following function.
2. By use of the delay function flash the LED connected to pin 13.

Assignment A2

1. Make a program with the following function
2. Connect a wire to an input port.
3. Turn the LED on.
4. by connecting the wire to ground.
5. Turn off by disconnection the wire from ground

Assignment A3

1. Connect the keypad 3x4 to input and output pins.
2. Make a program that decodes the keypad.
3. Let the serialprint show the pressed number.

Assignment A3A

1. Connect the keypad 3x4 to input and output pins.
2. Make a program that decodes the keypad.
3. Make a secret code of 4 digits, when the code is entered it turns the LED (pin 13) on or off

Assignment A4

1. Connect a LDR and a 1K resister in serial, connect the LED to 5 volt, and resistor til ground, the middel-point between resistor and LED connects to an analog input pin.
2. With Serial.print check the values of the light in shadow and in light.
3. Turn on the LED pin 13 on when the LDR is in the shadow.

Assignment A5

* Try with the resistor and the LDR to check the speed of the train, you have to turn on/off a timer, and then read the value with the Serial.print().