

ECE – 304

JDP

Spring 2022

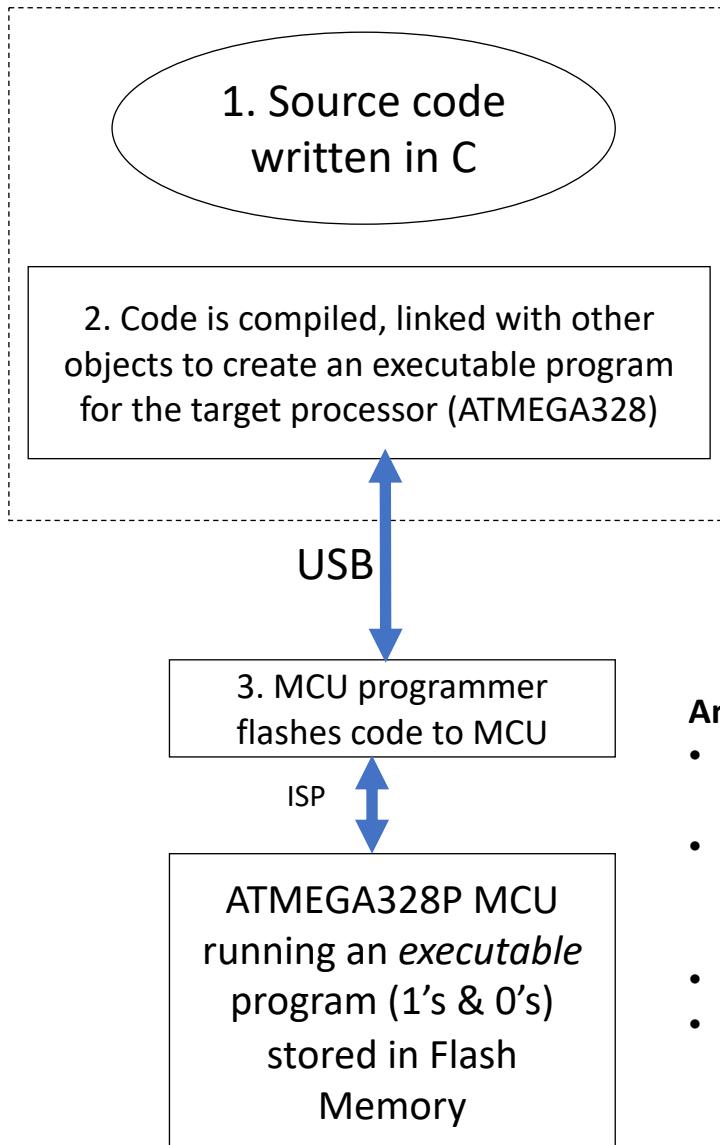
David McLaughlin, Professor
he/him
ECE Dept
University of Massachusetts at Amherst

JDP Schedule Spring 2022			
	Date	Class Meeting Topic	Due COB
Week 1	1/28/22	Lecture 1 - Course Intro	
Week 2	2/4/22	Lecture 2 - Project Specifications	collect kit
Week 3	2/11/22	Q&A with the pretend customer	
Week 4	2/18/22	Lecture 3 - Arduino & Sonar Demo	Build1 PDR report due
Week 5	2/25/22	Lecture 4 - Test plans, issues	
Week 6	3/4/22	No Class	Built1 Test plan & EVM1
Week 7	3/11/22	Build1 Report Due, no Class Meeting	Build1 report
	3/18/22	Spring Break	
Week 8	3/25/22	Lecture5 - avr-gcc tools demo	
Week 9	4/1/22	Lecture 6 - Risk Mgt, Q&A with pretend customer	
Week 10	4/8/22	Build2 PDR Due, no Class Meeting	Build2 PDR report due
Week 11	4/15/22	Senior Design Project (SDP) Topics	EVM2
Week 12	4/22/22	Senior Design Project (SDP) Topics	
Week 13	4/29/22	Build2 Report Due, no class meeting	Build2 report

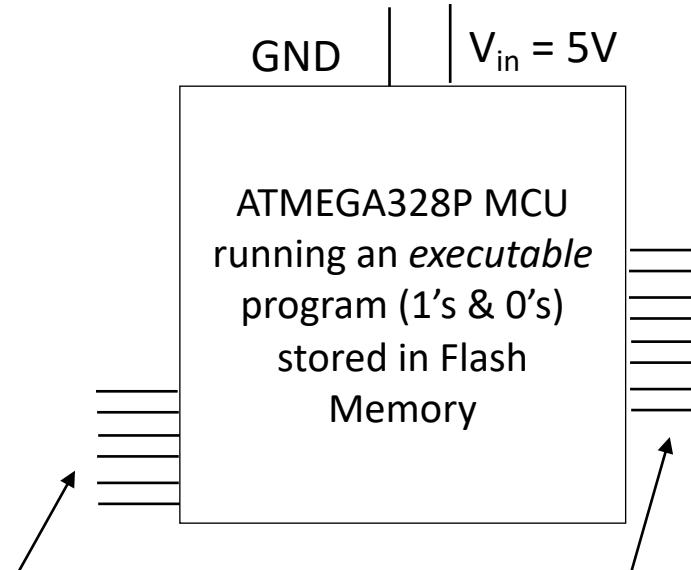
Lecture 3 - 2/25/21 –
Arduino MCU Dev board

Creating an Embedded System

Professional Integrated Development Environment (IDE) –
Microchip Studio 7 (Win 10) MPLAB X (Win, MacOS, Linux)
OR coding text editor (Visual Studio Code)



MCU Integrated Circuit



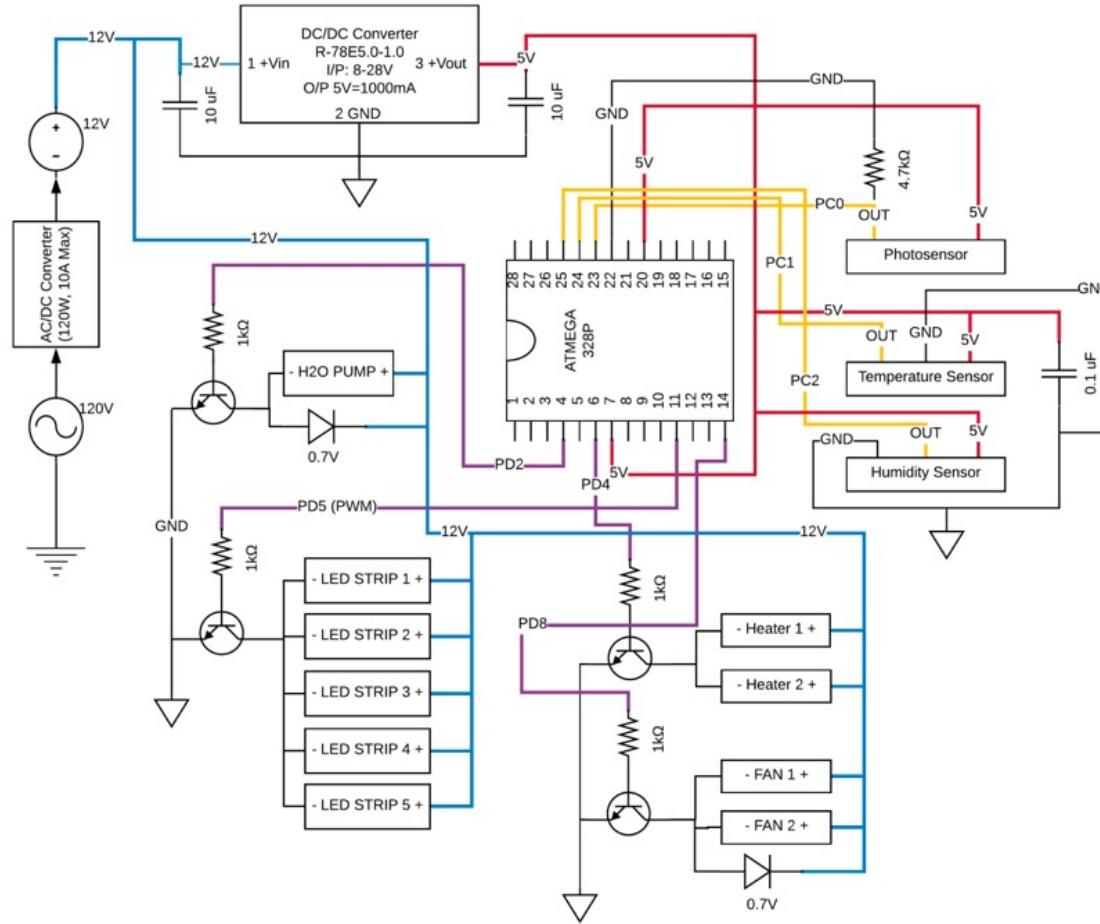
Analog Input (ADC) Pins

- Input voltage typically between 0 and 5V
- 10 bit representation of the input voltage ($2^{10} = 1024$)
- 0V interpreted as 0
- 5V interpreted as 1023

General Purpose Input & Output (GPIO) Pins

- logically 0 or 1
- electrically 0V or 5V
- Output pins: assert 0V (0) or 5V (1) on a pin, causing something external to happen
- Input pins: read 0V (0) or 5V (1) to interpret something external

Example



Creating an Embedded System in ECE 232 & ECE304 Spring 2022 (recommended)

Visual Studio Code (free editor)

1. Source code written in C

2. Command line: Code is compiled, linked with other objects to create an executable program for the target processor (ATMEGA328)

USB

Arduino Uno

Serial to USB converter

serial

ATMEGA328P MCU running an *executable* program (1's & 0's) stored in Flash Memory

Arduino Uno (for first build)

Serial to USB

Voltage regulator

Onboard LED

device peripheral

device peripheral

ATMEGA328P MCU running an *executable* program (1's & 0's) stored in Flash Memory

Analog Input (ADC) Pins

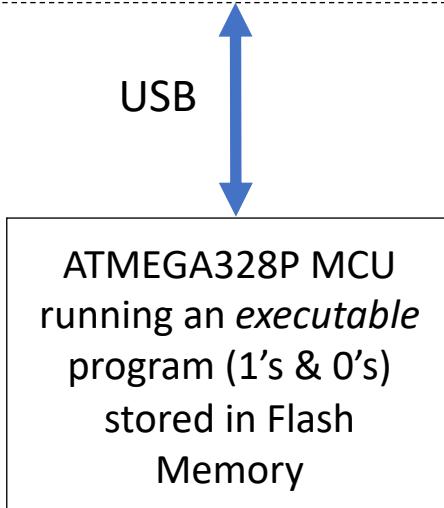
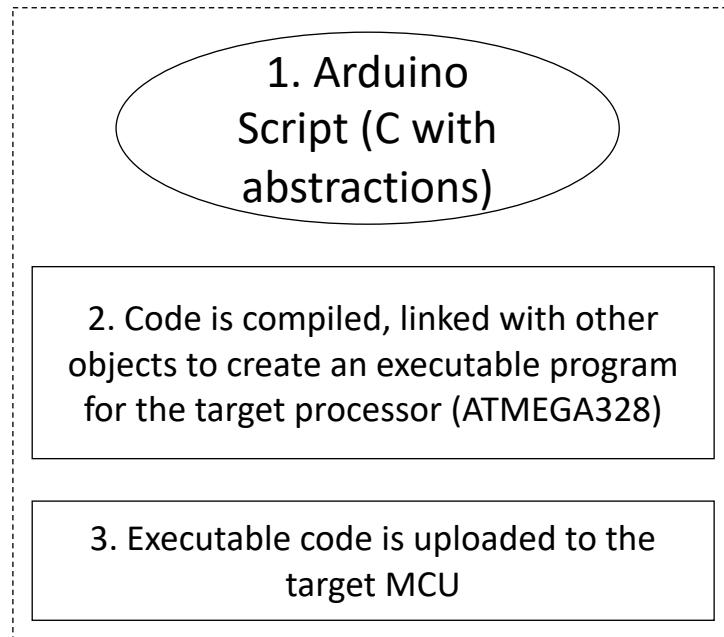
- Input voltage typically between 0 and 5V
- 10 bit representation of the input voltage ($2^{10} = 1024$)
- 0V interpreted as 0
- 5V interpreted as 1023

General Purpose Input & Output (GPIO) Pins

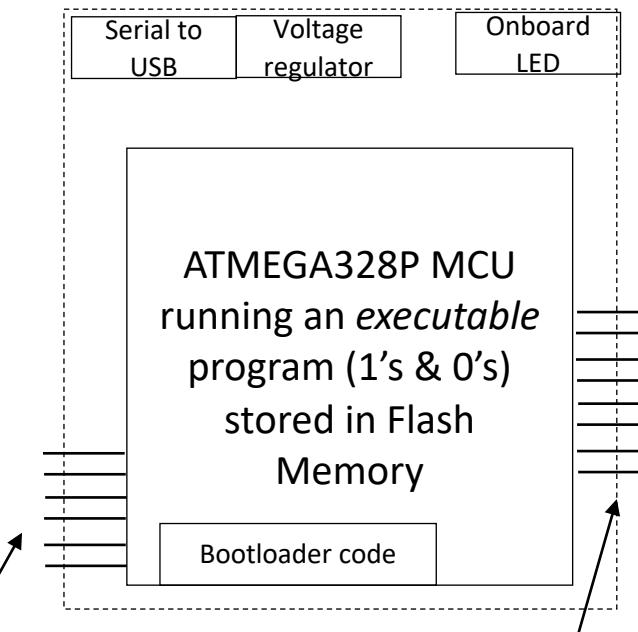
- logically 0 or 1
- electrically 0V or 5V
- Output pins: assert 0V (0) or 5V (1) on a pin, causing something external to happen
- Input pins: read 0V (0) or 5V (1) to interpret something external

Arduino Uno Implementation

Arduino (IDE)



Arduino Uno Development Board



Analog Input (ADC) Pins

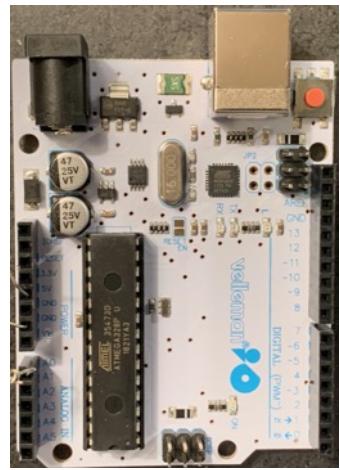
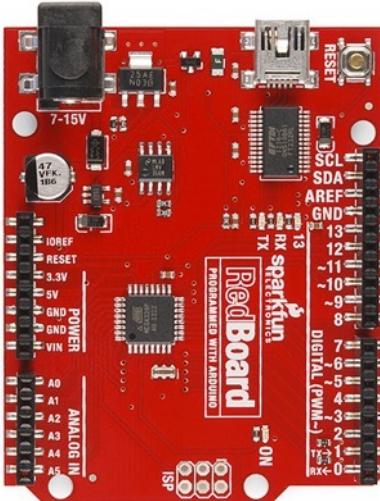
- Input voltage typically between 0 and 5V
- 10 bit representation of the input voltage ($2^{10} = 1024$)
- 0V interpreted as 0
- 5V interpreted as 1023

General Purpose Input & Output (GPIO) Pins

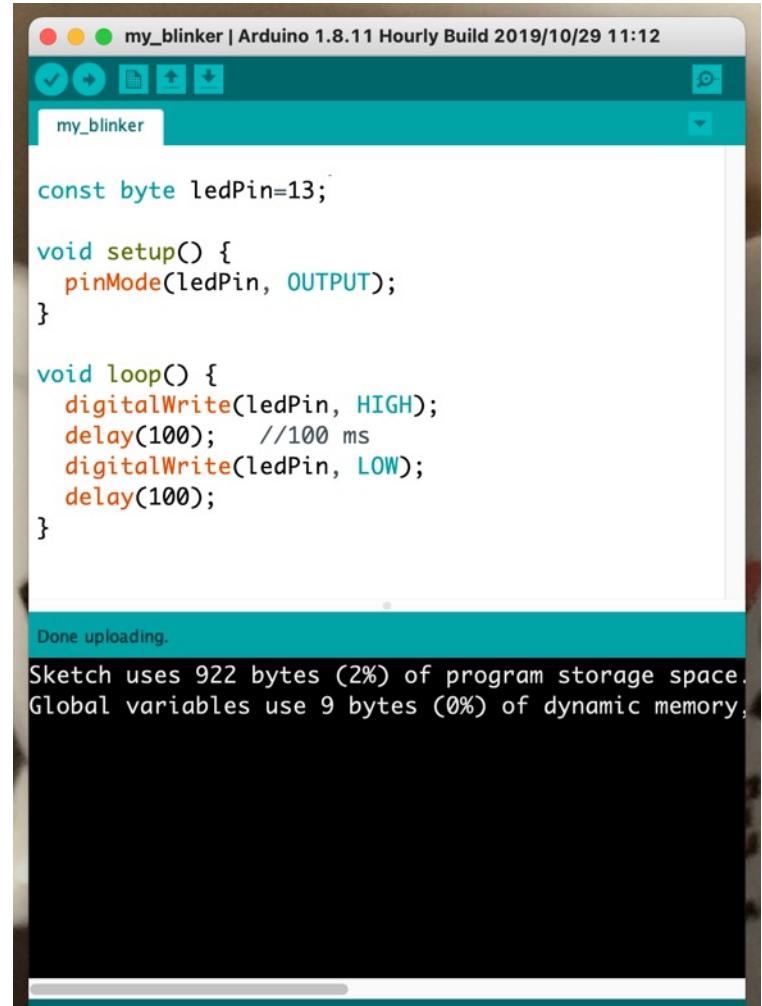
- logically 0 or 1
- electrically 0V or 5V
- Output pins: assert 0V (0) or 5V (1) on a pin, causing something external to happen
- Input pins: read 0V (0) or 5V (1) to interpret something external

Arduino Uno Dev Environment

Arduino Uno dev board
& compatible boards



Arduino IDE and many open source scripts



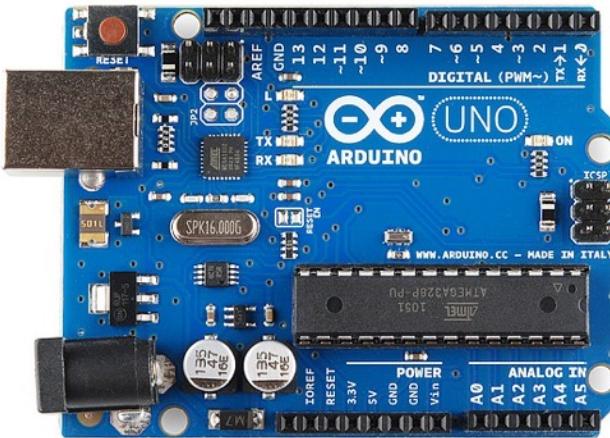
A screenshot of the Arduino IDE interface. The title bar says "my_blinker | Arduino 1.8.11 Hourly Build 2019/10/29 11:12". The code editor contains the following sketch:

```
const byte ledPin=13;

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

void loop() {
  digitalWrite(ledPin, HIGH);
  delay(100); //100 ms
  digitalWrite(ledPin, LOW);
  delay(100);
}
```

The status bar at the bottom shows: "Done uploading." and "Sketch uses 922 bytes (2%) of program storage space. Global variables use 9 bytes (0%) of dynamic memory," followed by "Arduino/Genuino Uno on /dev/cu.usbserial-1410".



Arduino Uno
plastic dual in-line package (PDIP) MCU
surface mount MCU (SMT)

USB Type A –Type B cable

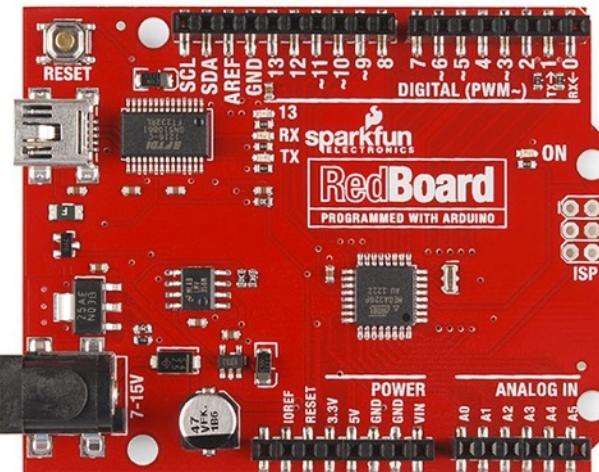
USB – Serial Chip: ATmega16U2 w/ custom firmware
Arduino USB driver

SparkFun Redboard
surface mount MCU (SMT)

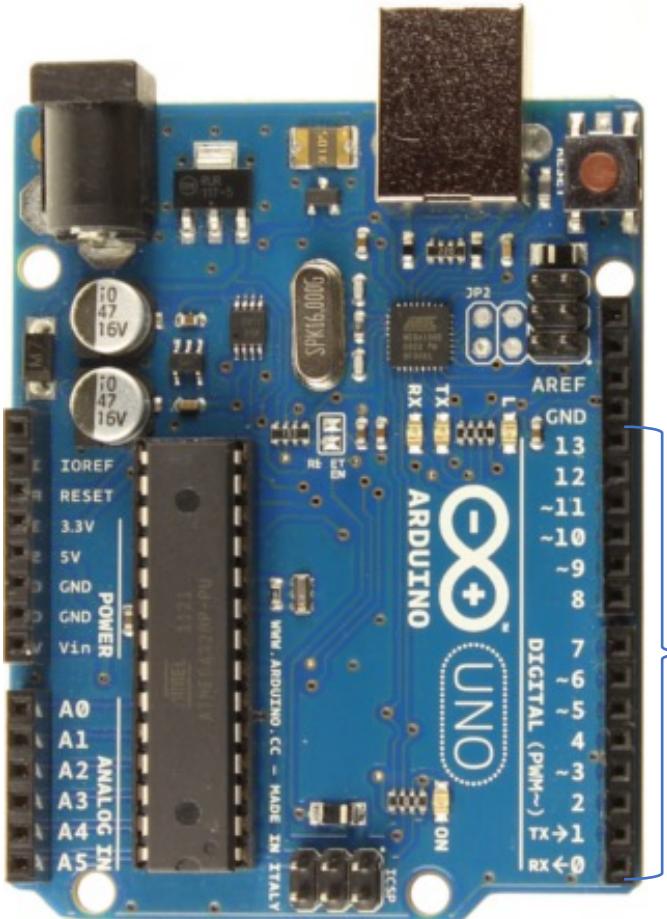
USB Type A – micro B cable

USB – Serial Chip: FTDI FT232
FTDI VCP Driver

<https://ftdichip.com/drivers/vcp-drivers/>



14 Digital Input & Output Pins (HIGH = 5V; LOW = 0V)



- Individually set as either INPUT or OUTPUT
- Pins can source or sink 40 mA max.
- Input: Internal pull-up resistors 20-50kOhm
- Output: pins marked ~ can generate pseudo-analog outputs via pulse-width modulation (PWM)
- Use pins 0 or 1 with care (or don't use)

Code Instructions

```
pinMode(); //set as INPUT or OUTPUT  
digitalWrite(); //set HIGH or LOW  
digitalRead(); //read an input. HIGH or LOW  
analogWrite(); //set output 0-255 PWM
```

Analog Input Pins

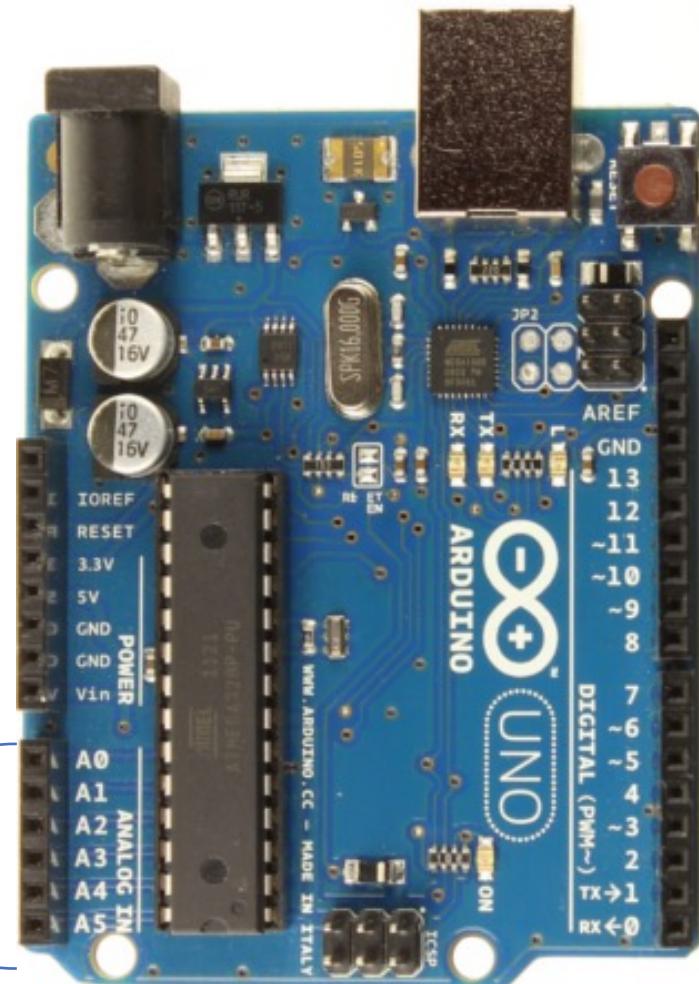
6 analog inputs.

10 bits resolution.

Can set reference voltage to 1.1 or 5v

0000000000= 0 =0V

1111111111=1024=5V (or 1.1V)



<http://arduino.cc/en/Main/arduinoBoardUno>

Programming the Arduino Uno

Use the Integrated Development Environment (IDE) – freeware. Obtain from <https://www.arduino.cc/en/Main/Software>

Programs aka “Scripts”

```
/* comments */  
  
#include <libraries>  
  
function A()  
{  
    instruction;  
    instruction;  
}  
  
function B()  
{  
    instruction;  
    instruction;  
}
```

All **Arduino scripts** must have these 2 functions:

```
void setup()  
{  
    /* this is run once after reset */  
}  
  
void loop()  
{  
    /* this one loops continuously */  
}
```

Contrast All **ANSI C** programs must have this 1 function:

```
void main()  
{  
    /* this is run once after reset */  
}
```

Anatomy of an Arduino Script (A generic picture)

Preprocessor
directives

```
#define XXX YYY          //preprocessor directive if needed  
#include ABC.h          //another preprocessor directive if needed
```

Must have
this function.
It runs once,
after power-
up or reset

```
void setup(){  
    statement;           //comment  
    statement;           //comment  
    ...  
    statement;  
}
```

```
void loop(){  
    statement;           //comment  
    statement;           //comment  
    ---  
    statement;           //comment  
}
```

```
int other_function(void){    //addition functions, if needed  
    statement;  
    statement;  
}
```

5 types of statements in C:
declaration ex: int x,y;
assignment ex: x=36;
function ex: printf("hello");
control structure ex: if (x>36) {... }
null ex: ;

All C statements end in ;

Software & Hardware to blink an LED on & off

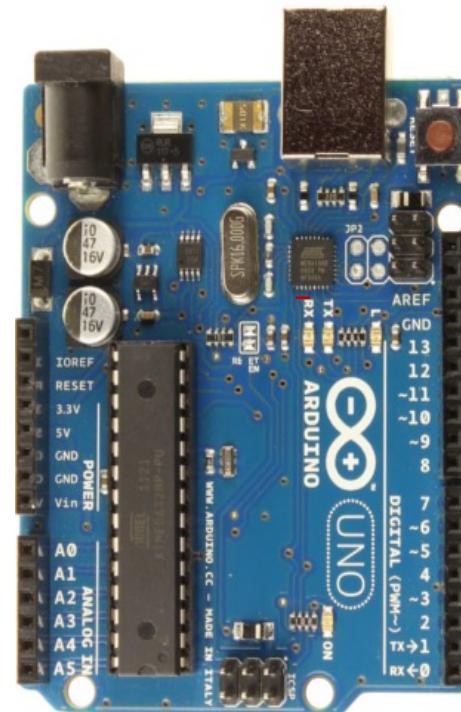
Blink | Arduino 1.0

```
/*
Blink
Turns on an LED on for one second, then off for one second, repeatedly.

This example code is in the public domain.
*/
void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH);    // set the LED on
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);     // set the LED off
  delay(1000);              // wait for a second
}
```

1 Arduino Uno on /dev/tty.usbmodemfd131



High (5V), Low (0V)

FUNCTIONS

For controlling the Arduino board and performing computations.

<https://www.arduino.cc/reference/en/>

Digital I/O

`digitalRead()`
`digitalWrite()`
`pinMode()`

Analog I/O

`analogRead()`
`analogReference()`
`analogWrite()`

Zero, Due & MKR Family

`analogReadResolution()`
`analogWriteResolution()`

Advanced I/O

`noTone()`
`pulseIn()`
`pulseInLong()`
`shiftIn()`
`shiftOut()`
`tone()`

Time

`delay()`
`delayMicroseconds()`
`micros()`
`millis()`

Math

`abs()`
`constrain()`
`map()`
`max()`
`min()`
`pow()`
`sq()`
`sqr()`

Trigonometry

`cos()`
`sin()`
`tan()`

Characters

`isAlpha()`
`isAlphaNumeric()`
`isAscii()`
`isControl()`
`isDigit()`
`isGraph()`
`isHexadecimalDigit()`
`isLowerCase()`
`isPrintable()`
`isPunct()`
`isSpace()`
`isUpperCase()`
`isWhitespace()`

Random Numbers

`random()`
`randomSeed()`

Bits and Bytes

`bit()`
`bitClear()`
`bitRead()`
`bitSet()`
`bitWrite()`
`highByte()`
`lowByte()`

External Interrupts

`attachInterrupt()`
`detachInterrupt()`

Interrupts

`interrupts()`
`noInterrupts()`

Communication

`Serial`
`Stream`

USB

`Keyboard`
`Mouse`

VARIABLES

<https://www.arduino.cc/reference/en/>

Arduino data types and constants.

Constants

HIGH | LOW

INPUT | OUTPUT | INPUT_PULLUP

LED_BUILTIN

true | false

Floating Point Constants

Integer Constants

Conversion

(unsigned int)

(unsigned long)

byte()

char()

float()

int()

long()

word()

Data Types

array

bool

boolean

byte

char

double

float

int

long

short

size_t

string

String()

unsigned char

unsigned int

unsigned long

void

word

Variable Scope & Qualifiers

const

scope

static

volatile

Utilities

PROGMEM

sizeof()

STRUCTURE

<https://www.arduino.cc/reference/en/>

The elements of Arduino (C++) code.

Sketch	Arithmetic Operators	Pointer Access Operators
loop()	% (remainder)	& (reference operator)
setup()	* (multiplication)	* (dereference operator)
	+ (addition)	
	- (subtraction)	
	/ (division)	
	= (assignment operator)	
Control Structure	Comparison Operators	Bitwise Operators
break	!= (not equal to)	& (bitwise and)
continue	< (less than)	<< (bitshift left)
do...while	<= (less than or equal to)	>> (bitshift right)
else	== (equal to)	^ (bitwise xor)
for	> (greater than)	(bitwise or)
goto	>= (greater than or equal to)	~ (bitwise not)
if		
return		
switch...case		
while		
	Boolean Operators	Compound Operators
Further Syntax	! (logical not)	%= (compound remainder)
#define (define)	&& (logical and)	&= (compound bitwise and)
#include (include)	(logical or)	*= (compound multiplication)
/* */ (block comment)		++ (increment)
// (single line comment)		+= (compound addition)
;		-- (decrement)
{ (curly braces)		-= (compound subtraction)
		/= (compound division)
		^= (compound bitwise xor)
		= (compound bitwise or)

Software & Hardware to turn on an LED when a push-button switch is pressed

LED_pushbutton_test_1 | Arduino 0021

File Edit Sketch Tools Help

LED_pushbutton_test_1

```
#define LED 13 //the pin for LED
#define BUTTON 7 //the input pin for button
int val = 0; //val will be used to store the state of the input
//pin

void setup() {
  pinMode(LED, OUTPUT); //tell Arduino LED is an output
  pinMode(BUTTON, INPUT); //and BUTTON is an input
}

void loop() {
  val = digitalRead(BUTTON); //read input value and store it

  if (val == HIGH) {
    digitalWrite(LED, HIGH); // turn LED on
  } else {
    digitalWrite(LED, LOW);
  }
}
```

Done Saving.

The sketch name had to be modified. Sketch names can only consist of ASCII characters and numbers (but cannot start with a number). They should also be less than 64 characters long.

Important: Internal pull-up resistor for INPUT mode.



showTime | Arduino 1.8.11 Hourly Build 2019/10/29 11:12

```
/* showTime
This script will print the time elapsed, in ms, since the
last reset of the Arduino. The output will be sent to the
serial monitor of the Arduino IDE via the Serial.print
command. Note the the Serial.begin(9600) command is used
within the setup() function to initialize the serial port
and set its baud rate to 9600 bits per second. Be sure to
set the baud rate of the serial monitor to 9600 bps also.
*/
unsigned long time;

void setup(){
  Serial.begin(9600);
}

void loop(){
  Serial.print("Time: ");
  time = millis();
  Serial.print(time);
  Serial.println(" ms");
  delay(1000);
}
```

Done uploading.

Sketch uses 1828 bytes (5%) of program storage space. Maximum is 32256 bytes.
Global variables use 198 bytes (9%) of dynamic memory, leaving 1850 bytes for lo

/dev

```
Time: 0 ms
Time: 999 ms
Time: 1999 ms
Time: 3000 ms
Time: 4000 ms
Time: 5001 ms
Time: 6000 ms
Time: 7001 ms
Time: 8001 ms
Time: 9001 ms
Time: 10002 ms
Time: 11002 ms
Time: 12002 ms
Time: 13002 ms
Time: 14003 ms
```

Autoscroll Show timestamp

OLED_simple | Arduino 1.8.19

OLED_simple

```
*****
* OLED_simple.ino This code demonstrates the use of a small
* 128 row x 64 column organic LED (OLED) display using a 4
* wire (VCC=5V, GND, SDA, SCL) I2C Interface.
* First, install the Adafruit SSD1306 library using the
* library manager in your Arduino IDE. (You might be promoted to
* also install Adafruit GFX Library and the Adafruit BusIO library.
* Your Arduino IDE might be smart enough to tell you to install these.
* If not, and if this code doesn't compile, then install them
* via the library manager.
* Date      Author      Revision
* 10/31/21   D. McLaughlin    initial code write for Engin 100/Digital Ready
* 2/15/22   D. McLaughlin    revised comments, prepared for ECE-304 Spring 2022
* *****

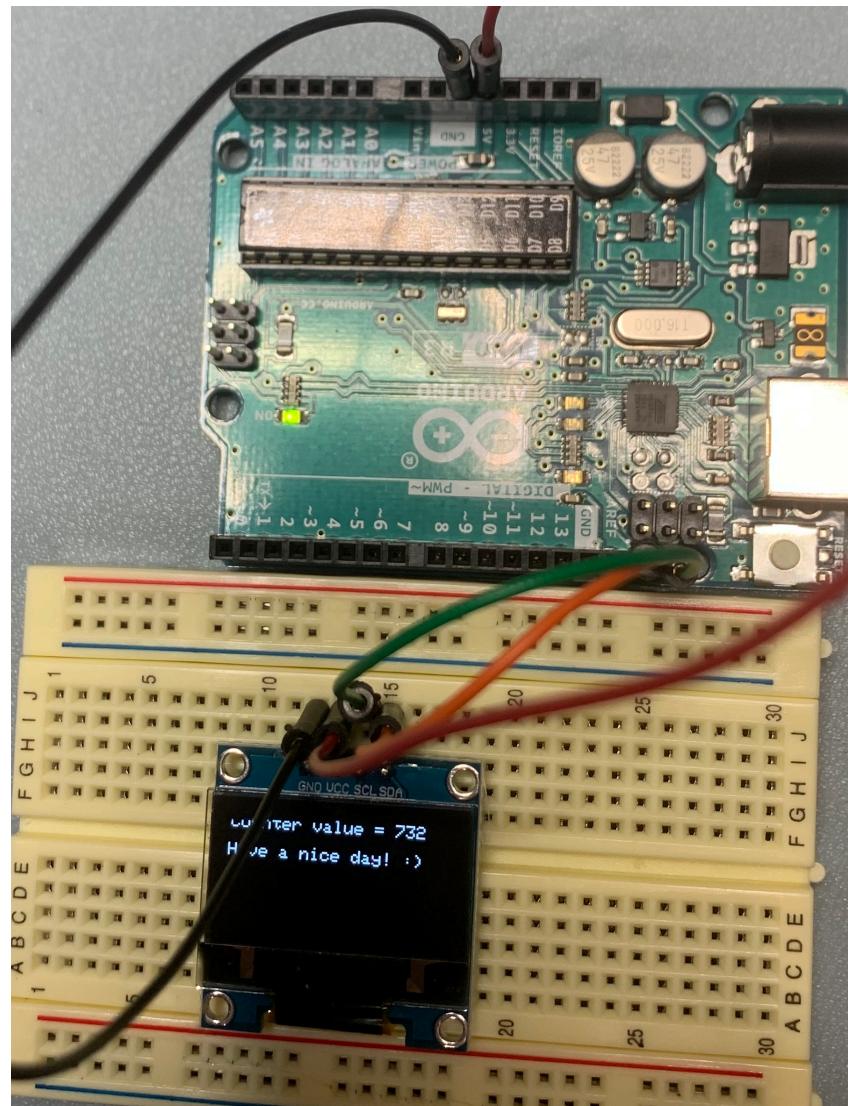
#include <SPI.h>
#include <Adafruit_SSD1306.h>
Adafruit_SSD1306 display(128,64, &Wire, 4);
int myCounter; //this is the variable that will be displayed

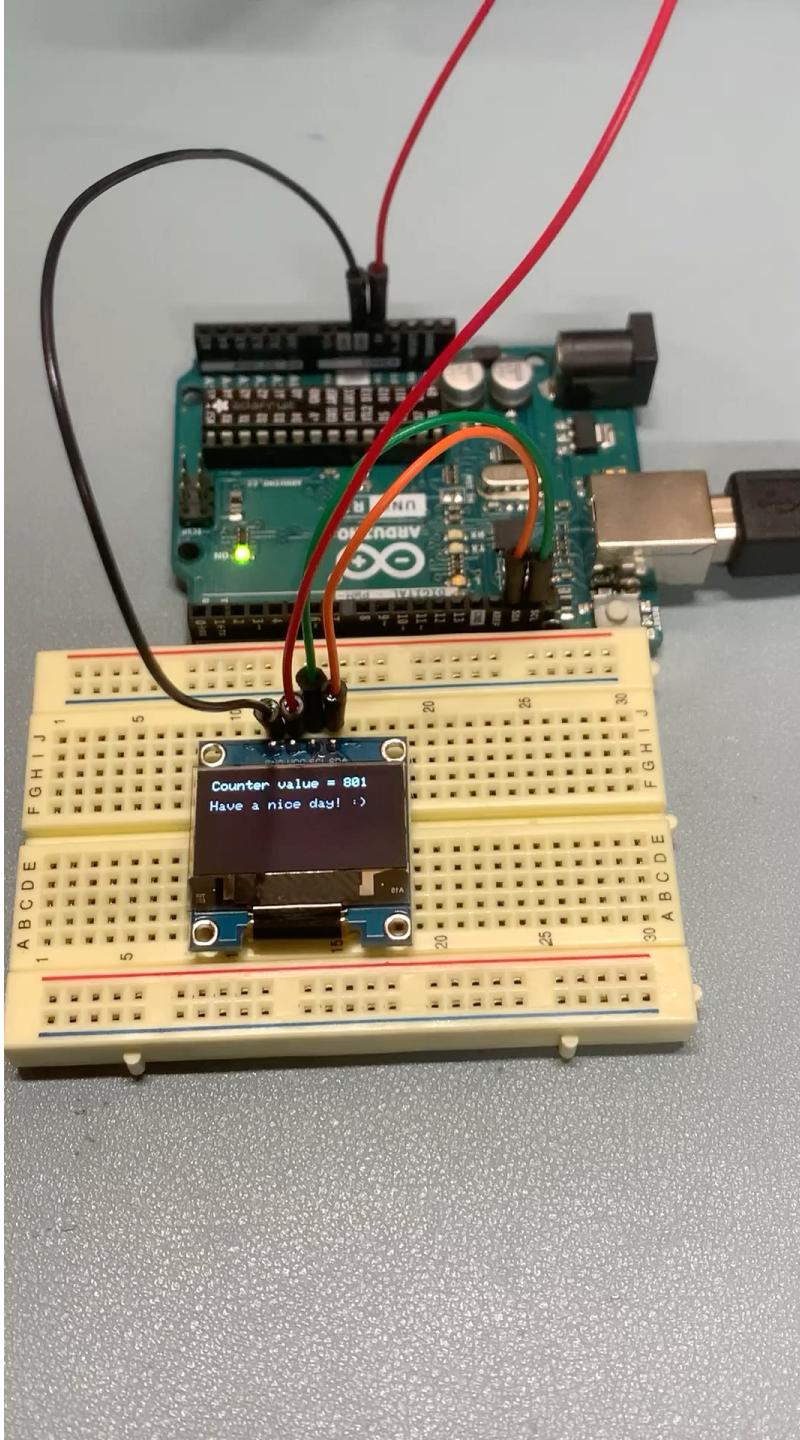
void setup() {
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(SSD1306_WHITE);
  myCounter = 0;
}

void loop() {
  display.clearDisplay();
  display.setCursor(1,1);
  myCounter = myCounter +1;
  display.print("Counter value = ");
  display.println(myCounter);
  display.println("");
  display.println("Hey there.\nHave a nice day! :)");
  display.display();
  if (myCounter > 1000){
    myCounter = 0;
  }
  delay(100);
}

***** End of File *****/

```





Four_seven_LED_test | Arduino 1.8.19

```

Four_seven_LED_test

/*
 * four_seven_LED_test. This code demonstrates a 4 digit 7 segment LED display
 * by running through the numbers 000 - 999 and repeating. Thus code was
 * written in C then modified for the setup() and loop() functions of the
 * Arduino IDE. Using Sparkfun COM-11409 common cathode 4 digit 7 segment
 * display with data sheet here: https://www.sparkfun.com/products/11409
 * Revision History
 * Date Author Revision
 * 2/22/21 D.McLaughlin initial C code "adc3DigitDisplay.c" for ECE-231 spring 2021
 * 2/15/22 D.McLaughlin rewritten as Arduino script for ECE-304 demo spring 2022.
 * Tested using Arduino 1.8.19 on MacBook Pro i9, macOS Big Sur 11.6.3
 */

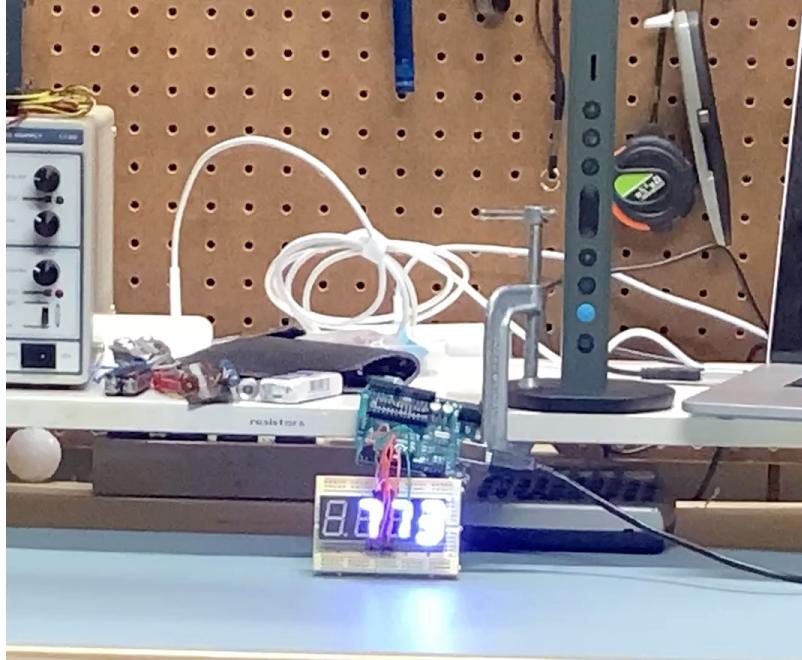
#include "avr/io.h"
#include "util/delay.h"
#define DIG2 PB2 //enables DIG2. ATmega328P pin PB2/Arduino Uno pin 10
#define DIG3 PB1 //enables DIG3. ATmega328P pin PB1/Arduino Uno pin 9
#define DIG4 PB0 //enables DIG4. ATmega328P pin PB0/Arduino Uno pin 8

void disp3Digits(unsigned char , unsigned char , unsigned char);

void setup() {
    DDRD = 0xFF; //7segment pins
    DDRB = (1 << 2) | (1 << 1) | (1 << 0); //digit enables
}

void loop() {
    unsigned char digit1, digit2, digit3;
    for (int value = 0; value<999; value++) {
        digit1 = value % 10;
        digit2 = (value / 10) % 10;
        digit3 = (value / 100);
        disp3Digits(digit1, digit2, digit3);
    }
}

/* Illuminate 3 digits of the 4 digit 7 segment display
 * The 7 segments (a, b, c, d, e, f, g) correspond to ATmega328P pins D0-D7,
 * Arduino Uno pins 0-6. Enable pins DIG2, DIG3, DIG4 correspond to PB0, PB1, PB0,
 * which corresponds to Arduino Uno pins 8, 9, 10 */
void disp3Digits(unsigned char d1,
    unsigned char d2,
    unsigned char d3)
{
    /* Set up the 4 digit 7 segment display
     * D0-D7 = A-G
     * D8-D11 = H-J
     * D12-D15 = K-L
     * D16-D19 = M-O
     * D20-D23 = P-S
     * D24-D27 = T-V
     * D28-D31 = W-Z
     * D32-D35 = AA-CC
     * D36-D39 = DD-FF
     * D40-D43 = GG-FF
     * D44-D47 = HH-FF
     * D48-D51 = KK-FF
     * D52-D55 = LL-FF
     * D56-D59 = MM-FF
     * D60-D63 = NN-FF
     * D64-D67 = OO-FF
     * D68-D71 = PP-FF
     * D72-D75 = RR-FF
     * D76-D79 = SS-FF
     * D80-D83 = TT-FF
     * D84-D87 = VV-FF
     * D88-D91 = ZZ-FF
     * D92-D95 = AA-FF
     * D96-D99 = CC-FF
     * D100-D103 = DD-FF
     * D104-D107 = FF-FF
     * D108-D111 = GG-FF
     * D112-D115 = HH-FF
     * D116-D119 = KK-FF
     * D120-D123 = LL-FF
     * D124-D127 = MM-FF
     * D128-D131 = NN-FF
     * D132-D135 = OO-FF
     * D136-D139 = PP-FF
     * D140-D143 = RR-FF
     * D144-D147 = SS-FF
     * D148-D151 = TT-FF
     * D152-D155 = VV-FF
     * D156-D159 = ZZ-FF
     * D160-D163 = AA-FF
     * D164-D167 = CC-FF
     * D168-D171 = DD-FF
     * D172-D175 = FF-FF
     * D176-D179 = GG-FF
     * D180-D183 = HH-FF
     * D184-D187 = KK-FF
     * D188-D191 = LL-FF
     * D192-D195 = MM-FF
     * D196-D199 = NN-FF
     * D200-D203 = OO-FF
     * D204-D207 = PP-FF
     * D208-D211 = RR-FF
     * D212-D215 = SS-FF
     * D216-D219 = TT-FF
     * D220-D223 = VV-FF
     * D224-D227 = ZZ-FF
     * D228-D231 = AA-FF
     * D232-D235 = CC-FF
     * D236-D239 = DD-FF
     * D240-D243 = FF-FF
     * D244-D247 = GG-FF
     * D248-D251 = HH-FF
     * D252-D255 = KK-FF
     * D256-D259 = LL-FF
     * D260-D263 = MM-FF
     * D264-D267 = NN-FF
     * D268-D271 = OO-FF
     * D272-D275 = PP-FF
     * D276-D279 = RR-FF
     * D280-D283 = SS-FF
     * D284-D287 = TT-FF
     * D288-D291 = VV-FF
     * D292-D295 = ZZ-FF
     * D296-D299 = AA-FF
     * D300-D303 = CC-FF
     * D304-D307 = DD-FF
     * D308-D311 = FF-FF
     * D312-D315 = GG-FF
     * D316-D319 = HH-FF
     * D320-D323 = KK-FF
     * D324-D327 = LL-FF
     * D328-D331 = MM-FF
     * D332-D335 = NN-FF
     * D336-D339 = OO-FF
     * D340-D343 = PP-FF
     * D344-D347 = RR-FF
     * D348-D351 = SS-FF
     * D352-D355 = TT-FF
     * D356-D359 = VV-FF
     * D360-D363 = ZZ-FF
     * D364-D367 = AA-FF
     * D368-D371 = CC-FF
     * D372-D375 = DD-FF
     * D376-D379 = FF-FF
     * D380-D383 = GG-FF
     * D384-D387 = HH-FF
     * D388-D391 = KK-FF
     * D392-D395 = LL-FF
     * D396-D399 = MM-FF
     * D400-D403 = NN-FF
     * D404-D407 = OO-FF
     * D408-D411 = PP-FF
     * D412-D415 = RR-FF
     * D416-D419 = SS-FF
     * D420-D423 = TT-FF
     * D424-D427 = VV-FF
     * D428-D431 = ZZ-FF
     * D432-D435 = AA-FF
     * D436-D439 = CC-FF
     * D440-D443 = DD-FF
     * D444-D447 = FF-FF
     * D448-D451 = GG-FF
     * D452-D455 = HH-FF
     * D456-D459 = KK-FF
     * D460-D463 = LL-FF
     * D464-D467 = MM-FF
     * D468-D471 = NN-FF
     * D472-D475 = OO-FF
     * D476-D479 = PP-FF
     * D480-D483 = RR-FF
     * D484-D487 = SS-FF
     * D488-D491 = TT-FF
     * D492-D495 = VV-FF
     * D496-D499 = ZZ-FF
     * D500-D503 = AA-FF
     * D504-D507 = CC-FF
     * D508-D511 = DD-FF
     * D512-D515 = FF-FF
     * D516-D519 = GG-FF
     * D520-D523 = HH-FF
     * D524-D527 = KK-FF
     * D528-D531 = LL-FF
     * D532-D535 = MM-FF
     * D536-D539 = NN-FF
     * D540-D543 = OO-FF
     * D544-D547 = PP-FF
     * D548-D551 = RR-FF
     * D552-D555 = SS-FF
     * D556-D559 = TT-FF
     * D560-D563 = VV-FF
     * D564-D567 = ZZ-FF
     * D568-D571 = AA-FF
     * D572-D575 = CC-FF
     * D576-D579 = DD-FF
     * D580-D583 = FF-FF
     * D584-D587 = GG-FF
     * D588-D591 = HH-FF
     * D592-D595 = KK-FF
     * D596-D599 = LL-FF
     * D600-D603 = MM-FF
     * D604-D607 = NN-FF
     * D608-D611 = OO-FF
     * D612-D615 = PP-FF
     * D616-D619 = RR-FF
     * D620-D623 = SS-FF
     * D624-D627 = TT-FF
     * D628-D631 = VV-FF
     * D632-D635 = ZZ-FF
     * D636-D639 = AA-FF
     * D640-D643 = CC-FF
     * D644-D647 = DD-FF
     * D648-D651 = FF-FF
     * D652-D655 = GG-FF
     * D656-D659 = HH-FF
     * D660-D663 = KK-FF
     * D664-D667 = LL-FF
     * D668-D671 = MM-FF
     * D672-D675 = NN-FF
     * D676-D679 = OO-FF
     * D680-D683 = PP-FF
     * D684-D687 = RR-FF
     * D688-D691 = SS-FF
     * D692-D695 = TT-FF
     * D696-D699 = VV-FF
     * D700-D703 = ZZ-FF
     * D704-D707 = AA-FF
     * D708-D711 = CC-FF
     * D712-D715 = DD-FF
     * D716-D719 = FF-FF
     * D720-D723 = GG-FF
     * D724-D727 = HH-FF
     * D728-D731 = KK-FF
     * D732-D735 = LL-FF
     * D736-D739 = MM-FF
     * D740-D743 = NN-FF
     * D744-D747 = OO-FF
     * D748-D751 = PP-FF
     * D752-D755 = RR-FF
     * D756-D759 = SS-FF
     * D760-D763 = TT-FF
     * D764-D767 = VV-FF
     * D768-D771 = ZZ-FF
     * D772-D775 = AA-FF
     * D776-D779 = CC-FF
     * D780-D783 = DD-FF
     * D784-D787 = FF-FF
     * D788-D791 = GG-FF
     * D792-D795 = HH-FF
     * D796-D799 = KK-FF
     * D800-D803 = LL-FF
     * D804-D807 = MM-FF
     * D808-D811 = NN-FF
     * D812-D815 = OO-FF
     * D816-D819 = PP-FF
     * D820-D823 = RR-FF
     * D824-D827 = SS-FF
     * D828-D831 = TT-FF
     * D832-D835 = VV-FF
     * D836-D839 = ZZ-FF
     * D840-D843 = AA-FF
     * D844-D847 = CC-FF
     * D848-D851 = DD-FF
     * D852-D855 = FF-FF
     * D856-D859 = GG-FF
     * D860-D863 = HH-FF
     * D864-D867 = KK-FF
     * D868-D871 = LL-FF
     * D872-D875 = MM-FF
     * D876-D879 = NN-FF
     * D880-D883 = OO-FF
     * D884-D887 = PP-FF
     * D888-D891 = RR-FF
     * D892-D895 = SS-FF
     * D896-D899 = TT-FF
     * D900-D903 = VV-FF
     * D904-D907 = ZZ-FF
     * D908-D911 = AA-FF
     * D912-D915 = CC-FF
     * D916-D919 = DD-FF
     * D920-D923 = FF-FF
     * D924-D927 = GG-FF
     * D928-D931 = HH-FF
     * D932-D935 = KK-FF
     * D936-D939 = LL-FF
     * D940-D943 = MM-FF
     * D944-D947 = NN-FF
     * D948-D951 = OO-FF
     * D952-D955 = PP-FF
     * D956-D959 = RR-FF
     * D960-D963 = SS-FF
     * D964-D967 = TT-FF
     * D968-D971 = VV-FF
     * D972-D975 = ZZ-FF
     * D976-D979 = AA-FF
     * D980-D983 = CC-FF
     * D984-D987 = DD-FF
     * D988-D991 = FF-FF
     * D992-D995 = GG-FF
     * D996-D999 = HH-FF
     * D1000-D1003 = KK-FF
     * D1004-D1007 = LL-FF
     * D1008-D1011 = MM-FF
     * D1012-D1015 = NN-FF
     * D1016-D1019 = OO-FF
     * D1020-D1023 = PP-FF
     * D1024-D1027 = RR-FF
     * D1028-D1031 = SS-FF
     * D1032-D1035 = TT-FF
     * D1036-D1039 = VV-FF
     * D1040-D1043 = ZZ-FF
     * D1044-D1047 = AA-FF
     * D1048-D1051 = CC-FF
     * D1052-D1055 = DD-FF
     * D1056-D1059 = FF-FF
     * D1060-D1063 = GG-FF
     * D1064-D1067 = HH-FF
     * D1068-D1071 = KK-FF
     * D1072-D1075 = LL-FF
     * D1076-D1079 = MM-FF
     * D1080-D1083 = NN-FF
     * D1084-D1087 = OO-FF
     * D1088-D1091 = PP-FF
     * D1092-D1095 = RR-FF
     * D1096-D1099 = SS-FF
     * D1100-D1103 = TT-FF
     * D1104-D1107 = VV-FF
     * D1108-D1111 = ZZ-FF
     * D1112-D1115 = AA-FF
     * D1116-D1119 = CC-FF
     * D1120-D1123 = DD-FF
     * D1124-D1127 = FF-FF
     * D1128-D1131 = GG-FF
     * D1132-D1135 = HH-FF
     * D1136-D1139 = KK-FF
     * D1140-D1143 = LL-FF
     * D1144-D1147 = MM-FF
     * D1148-D1151 = NN-FF
     * D1152-D1155 = OO-FF
     * D1156-D1159 = PP-FF
     * D1160-D1163 = RR-FF
     * D1164-D1167 = SS-FF
     * D1168-D1171 = TT-FF
     * D1172-D1175 = VV-FF
     * D1176-D1179 = ZZ-FF
     * D1180-D1183 = AA-FF
     * D1184-D1187 = CC-FF
     * D1188-D1191 = DD-FF
     * D1192-D1195 = FF-FF
     * D1196-D1199 = GG-FF
     * D1200-D1203 = HH-FF
     * D1204-D1207 = KK-FF
     * D1208-D1211 = LL-FF
     * D1212-D1215 = MM-FF
     * D1216-D1219 = NN-FF
     * D1220-D1223 = OO-FF
     * D1224-D1227 = PP-FF
     * D1228-D1231 = RR-FF
     * D1232-D1235 = SS-FF
     * D1236-D1239 = TT-FF
     * D1240-D1243 = VV-FF
     * D1244-D1247 = ZZ-FF
     * D1248-D1251 = AA-FF
     * D1252-D1255 = CC-FF
     * D1256-D1259 = DD-FF
     * D1260-D1263 = FF-FF
     * D1264-D1267 = GG-FF
     * D1268-D1271 = HH-FF
     * D1272-D1275 = KK-FF
     * D1276-D1279 = LL-FF
     * D1280-D1283 = MM-FF
     * D1284-D1287 = NN-FF
     * D1288-D1291 = OO-FF
     * D1292-D1295 = PP-FF
     * D1296-D1299 = RR-FF
     * D1300-D1303 = SS-FF
     * D1304-D1307 = TT-FF
     * D1308-D1311 = VV-FF
     * D1312-D1315 = ZZ-FF
     * D1316-D1319 = AA-FF
     * D1320-D1323 = CC-FF
     * D1324-D1327 = DD-FF
     * D1328-D1331 = FF-FF
     * D1332-D1335 = GG-FF
     * D1336-D1339 = HH-FF
     * D1340-D1343 = KK-FF
     * D1344-D1347 = LL-FF
     * D1348-D1351 = MM-FF
     * D1352-D1355 = NN-FF
     * D1356-D1359 = OO-FF
     * D1360-D1363 = PP-FF
     * D1364-D1367 = RR-FF
     * D1368-D1371 = SS-FF
     * D1372-D1375 = TT-FF
     * D1376-D1379 = VV-FF
     * D1380-D1383 = ZZ-FF
     * D1384-D1387 = AA-FF
     * D1388-D1391 = CC-FF
     * D1392-D1395 = DD-FF
     * D1396-D1399 = FF-FF
     * D1400-D1403 = GG-FF
     * D1404-D1407 = HH-FF
     * D1408-D1411 = KK-FF
     * D1412-D1415 = LL-FF
     * D1416-D1419 = MM-FF
     * D1420-D1423 = NN-FF
     * D1424-D1427 = OO-FF
     * D1428-D1431 = PP-FF
     * D1432-D1435 = RR-FF
     * D1436-D1439 = SS-FF
     * D1440-D1443 = TT-FF
     * D1444-D1447 = VV-FF
     * D1448-D1451 = ZZ-FF
     * D1452-D1455 = AA-FF
     * D1456-D1459 = CC-FF
     * D1460-D1463 = DD-FF
     * D1464-D1467 = FF-FF
     * D1468-D1471 = GG-FF
     * D1472-D1475 = HH-FF
     * D1476-D1479 = KK-FF
     * D1480-D1483 = LL-FF
     * D1484-D1487 = MM-FF
     * D1488-D1491 = NN-FF
     * D1492-D1495 = OO-FF
     * D1496-D1499 = PP-FF
     * D1500-D1503 = RR-FF
     * D1504-D1507 = SS-FF
     * D1508-D1511 = TT-FF
     * D1512-D1515 = VV-FF
     * D1516-D1519 = ZZ-FF
     * D1520-D1523 = AA-FF
     * D1524-D1527 = CC-FF
     * D1528-D1531 = DD-FF
     * D1532-D1535 = FF-FF
     * D1536-D1539 = GG-FF
     * D1540-D1543 = HH-FF
     * D1544-D1547 = KK-FF
     * D1548-D1551 = LL-FF
     * D1552-D1555 = MM-FF
     * D1556-D1559 = NN-FF
     * D1560-D1563 = OO-FF
     * D1564-D1567 = PP-FF
     * D1568-D1571 = RR-FF
     * D1572-D1575 = SS-FF
     * D1576-D1579 = TT-FF
     * D1580-D1583 = VV-FF
     * D1584-D1587 = ZZ-FF
     * D1588-D1591 = AA-FF
     * D1592-D1595 = CC-FF
     * D1596-D1599 = DD-FF
     * D1600-D1603 = FF-FF
     * D1604-D1607 = GG-FF
     * D1608-D1611 = HH-FF
     * D1612-D1615 = KK-FF
     * D1616-D1619 = LL-FF
     * D1620-D1623 = MM-FF
     * D1624-D1627 = NN-FF
     * D1628-D1631 = OO-FF
     * D1632-D1635 = PP-FF
     * D1636-D1639 = RR-FF
     * D1640-D1643 = SS-FF
     * D1644-D1647 = TT-FF
     * D1648-D1651 = VV-FF
     * D1652-D1655 = ZZ-FF
     * D1656-D1659 = AA-FF
     * D1660-D1663 = CC-FF
     * D1664-D1667 = DD-FF
     * D1668-D1671 = FF-FF
     * D1672-D1675 = GG-FF
     * D1676-D1679 = HH-FF
     * D1680-D1683 = KK-FF
     * D1684-D1687 = LL-FF
     * D1688-D1691 = MM-FF
     * D1692-D1695 = NN-FF
     * D1696-D1699 = OO-FF
     * D1700-D1703 = PP-FF
     * D1704-D1707 = RR-FF
     * D1708-D1711 = SS-FF
     * D1712-D1715 = TT-FF
     * D1716-D1719 = VV-FF
     * D1720-D1723 = ZZ-FF
     * D1724-D1727 = AA-FF
     * D1728-D1731 = CC-FF
     * D1732-D1735 = DD-FF
     * D1736-D1739 = FF-FF
     * D1740-D1743 = GG-FF
     * D1744-D1747 = HH-FF
     * D1748-D1751 = KK-FF
     * D1752-D1755 = LL-FF
     * D1756-D1759 = MM-FF
     * D1760-D1763 = NN-FF
     * D1764-D1767 = OO-FF
     * D1768-D1771 = PP-FF
     * D1772-D1775 = RR-FF
     * D1776-D1779 = SS-FF
     * D1780-D1783 = TT-FF
     * D1784-D1787 = VV-FF
     * D1788-D1791 = ZZ-FF
     * D1792-D1795 = AA-FF
     * D1796-D1799 = CC-FF
     * D1800-D1803 = DD-FF
     * D1804-D1807 = FF-FF
     * D1808-D1811 = GG-FF
     * D1812-D1815 = HH-FF
     * D1816-D1819 = KK-FF
     * D1820-D1823 = LL-FF
     * D1824-D1827 = MM-FF
     * D1828-D1831 = NN-FF
     * D1832-D1835 = OO-FF
     * D1836-D1839 = PP-FF
     * D1840-D1843 = RR-FF
     * D1844-D1847 = SS-FF
     * D1848-D1851 = TT-FF
     * D1852-D1855 = VV-FF
     * D1856-D1859 = ZZ-FF
     * D1860-D1863 = AA-FF
     * D1864-D1867 = CC-FF
     * D1868-D1871 = DD-FF
     * D1872-D1875 = FF-FF
     * D1876-D1879 = GG-FF
     * D1880-D1883 = HH-FF
     * D1884-D1887 = KK-FF
     * D1888-D1891 = LL-FF
     * D1892-D1895 = MM-FF
     * D1896-D1899 = NN-FF
     * D1900-D1903 = OO-FF
     * D1904-D1907 = PP-FF
     * D1908-D1911 = RR-FF
     * D1912-D1915 = SS-FF
     * D1916-D1919 = TT-FF
     * D1920-D1923 = VV-FF
     * D1924-D1927 = ZZ-FF
     * D1928-D1931 = AA-FF
     * D1932-D1935 = CC-FF
     * D1936-D1939 = DD-FF
     * D1940-D1943 = FF-FF
     * D1944-D1947 = GG-FF
     * D1948-D1951 = HH-FF
     * D1952-D1955 = KK-FF
     * D1956-D1959 = LL-FF
     * D1960-D1963 = MM-FF
     * D1964-D1967 = NN-FF
     * D1968-D1971 = OO-FF
     * D1972-D1975 = PP-FF
     * D1976-D1979 = RR-FF
     * D1980-D1983 = SS-FF
     * D1984-D1987 = TT-FF
     * D1988-D1991 = VV-FF
     * D1992-D1995 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D1999 = LL-FF
     * D1996-D1999 = MM-FF
     * D1996-D1999 = NN-FF
     * D1996-D1999 = OO-FF
     * D1996-D1999 = PP-FF
     * D1996-D1999 = RR-FF
     * D1996-D1999 = SS-FF
     * D1996-D1999 = TT-FF
     * D1996-D1999 = VV-FF
     * D1996-D1999 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D1999 = LL-FF
     * D1996-D1999 = MM-FF
     * D1996-D1999 = NN-FF
     * D1996-D1999 = OO-FF
     * D1996-D1999 = PP-FF
     * D1996-D1999 = RR-FF
     * D1996-D1999 = SS-FF
     * D1996-D1999 = TT-FF
     * D1996-D1999 = VV-FF
     * D1996-D1999 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D1999 = LL-FF
     * D1996-D1999 = MM-FF
     * D1996-D1999 = NN-FF
     * D1996-D1999 = OO-FF
     * D1996-D1999 = PP-FF
     * D1996-D1999 = RR-FF
     * D1996-D1999 = SS-FF
     * D1996-D1999 = TT-FF
     * D1996-D1999 = VV-FF
     * D1996-D1999 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D1999 = LL-FF
     * D1996-D1999 = MM-FF
     * D1996-D1999 = NN-FF
     * D1996-D1999 = OO-FF
     * D1996-D1999 = PP-FF
     * D1996-D1999 = RR-FF
     * D1996-D1999 = SS-FF
     * D1996-D1999 = TT-FF
     * D1996-D1999 = VV-FF
     * D1996-D1999 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D1999 = LL-FF
     * D1996-D1999 = MM-FF
     * D1996-D1999 = NN-FF
     * D1996-D1999 = OO-FF
     * D1996-D1999 = PP-FF
     * D1996-D1999 = RR-FF
     * D1996-D1999 = SS-FF
     * D1996-D1999 = TT-FF
     * D1996-D1999 = VV-FF
     * D1996-D1999 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D1999 = LL-FF
     * D1996-D1999 = MM-FF
     * D1996-D1999 = NN-FF
     * D1996-D1999 = OO-FF
     * D1996-D1999 = PP-FF
     * D1996-D1999 = RR-FF
     * D1996-D1999 = SS-FF
     * D1996-D1999 = TT-FF
     * D1996-D1999 = VV-FF
     * D1996-D1999 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D1999 = LL-FF
     * D1996-D1999 = MM-FF
     * D1996-D1999 = NN-FF
     * D1996-D1999 = OO-FF
     * D1996-D1999 = PP-FF
     * D1996-D1999 = RR-FF
     * D1996-D1999 = SS-FF
     * D1996-D1999 = TT-FF
     * D1996-D1999 = VV-FF
     * D1996-D1999 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D1999 = LL-FF
     * D1996-D1999 = MM-FF
     * D1996-D1999 = NN-FF
     * D1996-D1999 = OO-FF
     * D1996-D1999 = PP-FF
     * D1996-D1999 = RR-FF
     * D1996-D1999 = SS-FF
     * D1996-D1999 = TT-FF
     * D1996-D1999 = VV-FF
     * D1996-D1999 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D1999 = LL-FF
     * D1996-D1999 = MM-FF
     * D1996-D1999 = NN-FF
     * D1996-D1999 = OO-FF
     * D1996-D1999 = PP-FF
     * D1996-D1999 = RR-FF
     * D1996-D1999 = SS-FF
     * D1996-D1999 = TT-FF
     * D1996-D1999 = VV-FF
     * D1996-D1999 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D1999 = LL-FF
     * D1996-D1999 = MM-FF
     * D1996-D1999 = NN-FF
     * D1996-D1999 = OO-FF
     * D1996-D1999 = PP-FF
     * D1996-D1999 = RR-FF
     * D1996-D1999 = SS-FF
     * D1996-D1999 = TT-FF
     * D1996-D1999 = VV-FF
     * D1996-D1999 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D1999 = LL-FF
     * D1996-D1999 = MM-FF
     * D1996-D1999 = NN-FF
     * D1996-D1999 = OO-FF
     * D1996-D1999 = PP-FF
     * D1996-D1999 = RR-FF
     * D1996-D1999 = SS-FF
     * D1996-D1999 = TT-FF
     * D1996-D1999 = VV-FF
     * D1996-D1999 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D1999 = LL-FF
     * D1996-D1999 = MM-FF
     * D1996-D1999 = NN-FF
     * D1996-D1999 = OO-FF
     * D1996-D1999 = PP-FF
     * D1996-D1999 = RR-FF
     * D1996-D1999 = SS-FF
     * D1996-D1999 = TT-FF
     * D1996-D1999 = VV-FF
     * D1996-D1999 = ZZ-FF
     * D1996-D1999 = AA-FF
     * D1996-D1999 = CC-FF
     * D1996-D1999 = DD-FF
     * D1996-D1999 = FF-FF
     * D1996-D1999 = GG-FF
     * D1996-D1999 = HH-FF
     * D1996-D1999 = KK-FF
     * D1996-D199
```



Confused? Some helpful references...

Arduino.cc tutorials...

<https://www.arduino.cc/en/Guide/ArduinoUno>

McLaughlin's e-book written for non-ECE majors has example code:

<http://openbooks.library.umass.edu/funee/part/9-0/>

Adafruit is emerging as a goto site for making, rapid prototyping, learning, purchasing...

<https://www.adafruit.com>

Step-by-Step Tutorials for Downloading the Arduino IDE,
Connecting your Arduino Uno board, writing first code:

Option 1

https://create.arduino.cc/projecthub/Arduino_Genuino/getting-started-with-the-arduino-desktop-ide-623be4

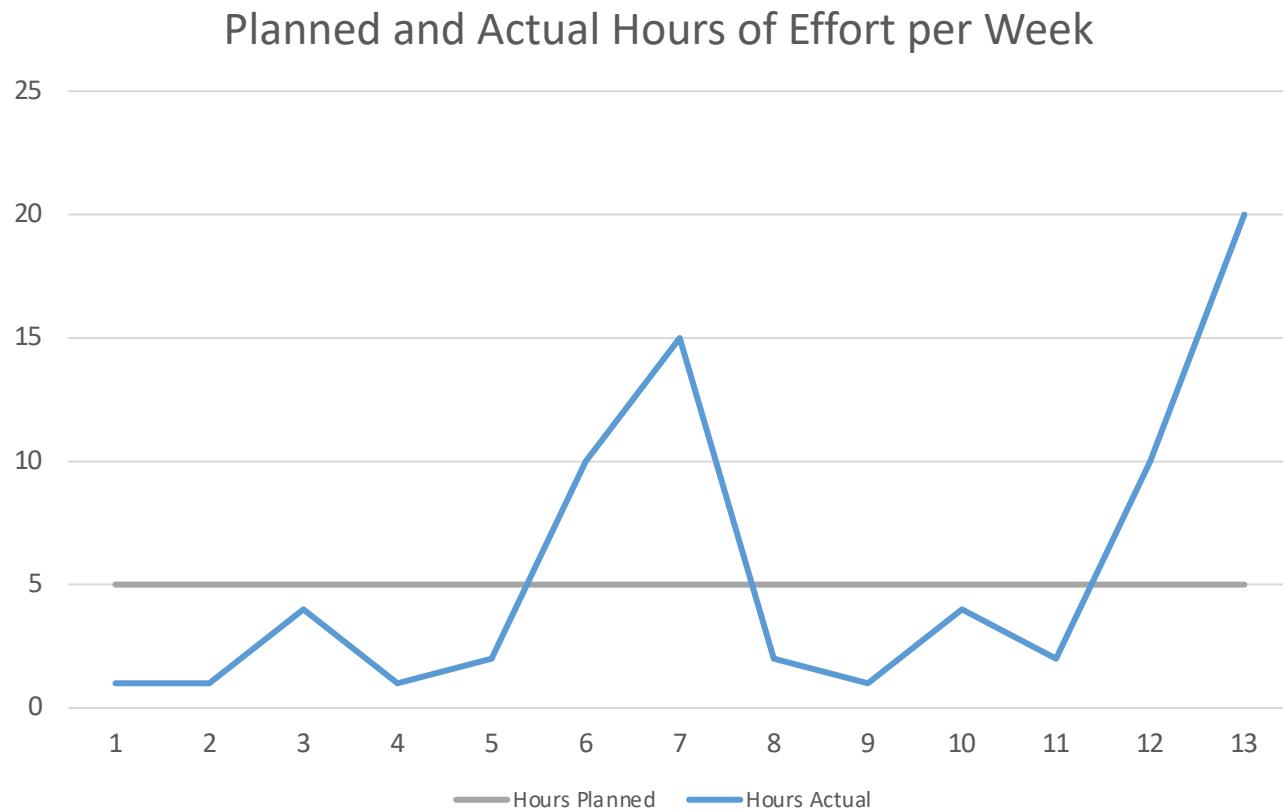
Option 2 Learn Arduino by Simon Monk (lessons 0-3)

<https://learn.adafruit.com/series/learn-arduino>

Option 3

<http://arduinotogo.com/2016/09/09/chapter-3-programming-the-arduino/>

For now, just keep track of your # hours spent in ECE-304.
Don't inflate them. This data is for you!



Assignment

- Commence detailed design & construction of subsystems
- Don't try to sit down and build the entire system at once. Build & test subsystems using your block diagram.
- Begin thinking about how you will
 - integrate the subsystems together
 - test the whole system
 - verify that the system meets your requirements
- Next week: System test & requirement verification