11

Glossary

ARDUINO CHEAT SHEET

Content for this Cheat Sheet provided by Gavin from Robots and Dinosaurs.
For more information visit: http://arduino.cc/en/Reference/Extended

Structure
void **setup**() void **loop**()

Control Structures
if (x<5){ } **else** { }
switch (myvar) {
  case 1:
  break;
  case 2:
  break;
  default:
}
for (int i=0; i <= 255; i++){ }
while (x<5){ }
do { } **while** (x<5);
continue; //Go to next in do/for/while loop
return x; // Or 'return;' for voids.
goto // considered harmful :-)

Further Syntax
// (single line comment)
/* (multi-line comment) */
#**define** DOZEN 12 //Not baker's!
#**include** <avr/pgmspace.h>

General Operators
= (assignment operator)
+ (addition) - (subtraction)
* (multiplication) / (division)
% (modulo)
== (equal to) != (not equal to)
< (less than) > (greater than)
<= (less than or equal to)
>= (greater than or equal to)
&& (and) || (or) ! (not)

Pointer Access
& reference operator
* dereference operator

Bitwise Operators
& (bitwise and) | (bitwise or)
^ (bitwise xor) ~ (bitwise not)
<< (bitshift left) >> (bitshift right)

Compound Operators
++ (increment) -- (decrement)
+= (compound addition)
-= (compound subtraction)
*= (compound multiplication)
/= (compound division)
&= (compound bitwise and)
|= (compound bitwise or)

Constants
HIGH | LOW
INPUT | OUTPUT
true | false
143 // **Decimal** number
0173 // **Octal** number
0b11011111 //**Binary**
0**x**7B // **Hex** number
7**U** // Force unsigned
10**L** // Force long
15**UL** // Force long unsigned
10.0 // Forces floating point
2.4**e**5 // 240000

Data Types
void
boolean (0, 1, false, true)
char (e.g. 'a' -128 to 127)
unsigned char (0 to 255)
byte (0 to 255)
int (-32,768 to 32,767)
unsigned int (0 to 65535)
word (0 to 655word (0 to 65535)
long (-2,147,483,648 to 2,147,483,647)
unsigned long (0 to 4,294,967,295)
float (-3.4028235E+38 to 3.4028235E+38)
double (currently same as float)
sizeof(myint) // returns 2 bytes

Strings
char S1[15];
char S2[8]={'a','r','d','u','i','n','o'};
char S3[8]={'a','r','d','u','i','n','o','\0'};
//Included \0 null termination
char S4[ ] = "arduino";
char S5[8] = "arduino";
char S6[15] = "arduino";

Arrays
int myInts[6];
int myPins[] = {2, 4, 8, 3, 6};
int mySensVals[6] = {2, 4, -8, 3, 2};

Conversion
char() **byte**()
int() **word**()
long() **float**()

Qualifiers
static // persists between calls
volatile // use RAM (nice for ISR)
const // make read-only
PROGMEM // use flash

Digital I/O
pinMode(pin, [INPUT,OUTPUT])
digitalWrite(pin, value)
int **digitalRead**(pin)
//Write High to inputs to use pull-up res

Analog I/O
analogReference([DEFAULT, INTERNAL,EXTERNAL])
int **analogRead**(pin) //Call twice if switching pins from high Z source.
analogWrite(pin, value) // PWM

Advanced I/O
tone(pin, freqhz)
tone(pin, freqhz ,duration_ms)
noTone(pin)
shiftOut(dataPin, clockPin, [MSBFIRST,LSBFIRST], value)
unsigned long **pulseIn**(pin,[HIGH,LOW])

Time
unsigned long **millis**() // 50 days overflow.
unsigned long **micros**() // 70 min overflow
delay(ms)
delayMicroseconds(us)

Math
min(x, y) **max**(x, y) **abs**(x)
constrain(x, minval, maxval)
map(val, fromL, fromH, toL, toH)
pow(base, exponent) **sqrt**(x)
sin(rad) **cos**(rad) **tan**(rad)

Random Numbers
randomSeed(seed) // Long or int
long **random**(max)
long **random**(min, max)

Bits and Bytes
lowByte()
highByte()
bitRead(x,bitn)
bitWrite(x,bitn,bit)
bitSet(x,bitn)
bitClear(x,bitn)
bit(bitn) //bitn: 0-LSB 7-MSB

External Interrupts
attachInterrupt(interrupt, function, [LOW,CHANGE,RISING,FALLING])
detachInterrupt(interrupt)
interrupts()
noInterrupts()

Libraries:

Serial.
  begin([300, 1200, 2400, 4800, 9600,14400, 19200, 28800, 38400, 57600,115200])
  end()
  int **available**()
  int **read**()
  flush()
  print()
  println()
  write()

EEPROM (#include <EEPROM.h>)
byte **read**(intAddr)
  write(intAddr,myByte)

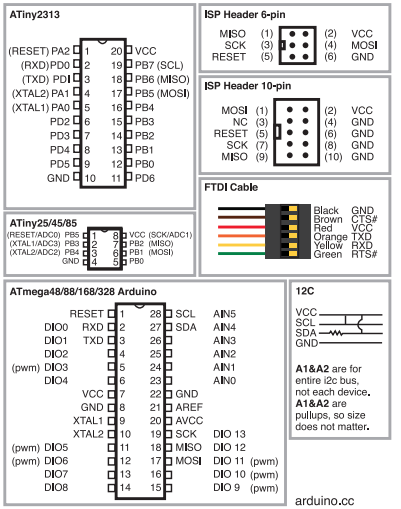
Servo (#include <Servo.h>)
  attach(pin , [min_uS, max_uS])
  write(angle) // 0-180
  writeMicroseconds(uS) //1000-2000,1500 is midpoint
  read() // 0-180
  attached() //Returns boolean
  detach()

SoftwareSerial(RxPin,TxPin)
  // #include<**SoftwareSerial**.h>
  begin(longSpeed) // up to 9600
  char **read**() // blocks till data
  print(myData) or **println**(myData)

Wire (#include <Wire.h>) // For **I2C**
  begin() // Join as master
  begin(addr) // Join as slave @ addr
  requestFrom(address, count)
  beginTransmission(addr) // Step 1
  send(mybyte) // Step 2
  send(char * mystring)
  send(byte * data, size)
  endTransmission() // Step 3
  byte **available**() // Num of bytes
  byte **receive**() //Return next byte
  onReceive(handler)
  onRequest(handler)

	ATMega168	ATMega328	ATMega1280
Flash (2k for boobtloader)	16kB	32kB	128kB
SRAM	1kB	2kB	8kB
EEPROM	512B	1kB	4kB

	Duemilanove/ Nano/ Pro/ ProMini	Mega
# of IO	14 + 6 analog (Nano has 14 + 8)	54 + 16 analog
Serial Pins	0 - RX 1 - TX	0 - RX1 1 - TX1 19 - RX2 18 - TX2 17 - RX3 16 - TX3 15 - RX4 14 - TX4
Ext Interrupts	2 - (Int 0) 1 - (Int 1)	2,3,21,20,19,18 (IRQ0 - IFQ5)
PWM Pins	5,6 - Timer 0 9,10 - Timer 1 3,11 - Timer 2	0 - 13
SPI	10 - SS 11 - MOSI 12 - MISO 13 - SCK	53 - SS 51 - MOSI 50 - MISO 52 - SCK
I2C	Analog4 - SDA Analog5 - SCK	20 - SDA 21 - SCL



arduino.cc

// Glossary

ADC: Analog to Digital Converter. Any method of converting an analog signal (a voltage) to a digital signal (a number). Here is the equation necessary to do this:

$$ADC\ Value = \frac{(Voltage\ on\ Pin(mV))*(Max.\ ADC\ Value)}{(System\ Voltage(mV))}$$

Analog: A measurement or signal that has values between On and Off. Examples include voltage, pressure, any type of wave, and volume. One useful metaphor for teaching is the zipper (if you have a zipper on your jacket or hoodie). In comparison to the button (Digital) the zipper has many states between completely open and closed. Analog's counter part is Digital.

array: Arrays are a way to store several variables in a list, or array. In Arduino arrays are declared in a couple different ways. Example: int arrayName [6]; this will create an array called arrayName with six spaces for variables inside it. A value of 100 would be assigned to the first space in arrayName like this: arrayName [0] = 100; (The first value is assigned to the 0 slot.) Here is an example of declaring and assigning values in an array at the same time: arrayName [6] = {100, 150, 300, 50, 100, 120}; Here is an example of referencing the sixth value in the array arrayName: arrayName [5].

Bias: A state describing voltage and current in a component.

Boolean: A variable type or form of logic based on the assumption that any given variable or state can only have one of two values; true or false, HIGH or LOW, 1 or 0.

bounce: Bounce occurs when a switch (or other type of input) attempts to change it's position to open or closed but does not stay in that position. Due to this you will see the electrical signal rising and falling when it should be at a constant value. If you're experiencing bounce issues, try using the delay( ) function.

button: A digital input with only two states; pressed or not pressed. Depending on the layout of the circuit these two states can correspond to either HIGH or LOW.

char: Variable type character, 8-bit size, any single symbol. Examples: A, a, 1, or !

comments: Used to write notes in code that are not part of the execution. For a single line use //, for a block of lines start with /* and end with */.

constrain: A function used to constrain a value between a given range. Example: sensVal = constrain (sensVal, 10, 150); This constrains the sensVal value between 10 and 150. If it is under 10 sensVal is assigned 10, if it is over 150 sensVal is assigned 150.

current: This can refer to either the the flow of electrical charge or the rate of flow of electrical charge, measured in mA.

Digital: A measurement or signal that has only two values, On and Off. On and Off can also be expressed as HIGH and LOW, as well as 1 and 0. Examples include Boolean logic, open or closed and button state. One useful metaphor for teaching is the button (if you have a button on your jacket or hoodie). In comparison to the zipper (Analog) the button has only two states; connected and unconnected. Digital's counter part is Analog.

diode: A two terminal electrical component that only conducts in one direction.

Ground: In electrical engineering, ground or earth may be the reference point in an electrical circuit from which other voltages are measured, or a common return path for electric current, or a direct physical connection to the Earth.

float (signal): A signal due to a pin that is not attached to anything. A floating pin can read anything between HIGH and LOW.

float (data type): Variable type float, used for floating point operations (i.e. numbers with decimal points).

for: for(variable declaration and assignment; condition; variable increment){ }
A form of iteration, code inside the curly brackets will repeat until the condition is false. Example of a for loop that will loop four times:
for (i = 0, i < 4, i++) {//do this code each time};

forward bias: The state of a component describing voltage saturation necessary to activate a component.

footprint: A footprint is the pattern on a circuit board to which your parts are attached. This includes the electrical connections and silkscreen.

flyback diode: Used to reduce voltage spikes seen across inductive loads due to a sudden loss in voltage from the power source.

if statements:
if(condition){ }
else if (condition) { }
else { }
This will execute the code inside the curly brackets if the condition is true, and if the condition is false it will test the "else if" condition. If the "else if" condition is true it will execute the code in the second set of curly brackets. Otherwise it will execute the code inside the third set of curly brackets.

Input: A signal or data received by a processor.

int: Variable type integer, 16-bit size, any number between -32,768 and 32,767.
iteration: When something happens over and over and over,

but changes a little each time.

lead: Electrical contacts for parts, these usually look like wires or pins extending off the part.

LED: A light emitting diode.

Library: A collection of code that has been packaged so it can be included and then used in code. Servo example: #include <Servo.h> (This includes the library so it can be used) Servo myservo; (This creates a servo object so the functions inside the library can be used) myservo.write(90); (This uses the write function in the servo library, setting the servo's angle to 90 degrees.
loop ( ): Looks like- void loop ( ){ } The main loop in an Arduino sketch, this where the action happens. The loop ( ) function is present in every single Arduino sketch. The Arduino will execute the code inside the curly brackets, once it has finished this code it will start over from the beginning of the loop function.

map: A function used to re-map a value between a range to a value between another given range. Example: sensVal = map (sensVal, 10, 150, 100, 1500); This maps the sensVal value from somewhere between 10 and 150 to somewhere between 100 and 1500 proportionally.

microcontroller: A tiny computer on a single integrated circuit with a core processor, memory and programmable input/output.

motor: An electrical component that converts electrical energy to mechanical energy.

Ohm's Law: V = I * R where V is Voltage, I is Current and R is Resistance. A handy way to figure out any one of these values given the other two. The complicated version: Ohm's law states that the current through a conductor between two points is directly proportional to the potential difference or voltage across the two points, and inversely proportional to the resistance between them.

operators: Similar to mathematical symbols, pay special attention to the difference between = and ==.

output: A signal or data transmitted by a processor.

piezo element: A digital component which moves a disc to one of two possible positions to create an analog output via vibration, often used to create annoying melodies with little aesthetic value.

Pin: A place to connect electrical circuits to one another.

pinMode: Arduino command used to set pins to either INPUT or OUTPUT.
Looks like- pinMode (pinNumber, value); where pinNumber is the pin to be set and value is either INPUT or OUTPUT. pinMode can also be used to turn Analog In pins into Digital pins.

potentiometer: A voltage divider with a dial to change the values of the two resistors inside.

pseudo-code: A human-readable way of describing a computer program so that it is easier to understand. See the description of 'if statements' in the glossary for an example.

Pulse Width Modulation: A commonly used technique for controlling digital systems to create a simulated analog output. Often abbreviated as PWM.9.

relay: An electrically operated switch.

Resistance: A measure of the opposition to electrical current. Measured in Ω (ohms).

resistor: Component used to restrict the amount of current that can flow through a circuit. Rated in Ω (ohms).

Serial: universal asynchronous receiver/transmitter.

Serial Monitor: Window in Arduino programming environment that allows the user to monitor serial communication.

setup (): Looks like- void setup ( ){ } All the code between the curly brackets will run once when the Arduino program first starts. (As well as each time it is restarted.)

shift register: In this case a chip which allows you to change the value of it's output pins, starting with either the first or last pin, by "shifting" a new value in and "shifting" all the old values towards the opposite end of the chip. The shift register uses three pins (latch, clock, and data) to control eight output pins.

sketch: The code created in your Arduino environment which is then saved onto your Arduino board to make something happen.

sketchbook: Folder where your sketches are stored.

temperature sensor: A sensor used to convert temperature to an analog reading, in this case a voltage, which can be read by your Arduino.

trace: A copper path on a PCB necessary for electrical conductivity between parts.

transistor: A semiconductor device used to amplify or control an electronic signal.

value: Worth or symbol stored in a variable.

variable: A symbolic name associated with a value and whose associated value may be changed.

voltage: The difference in electrical potential between any two given points of a circuit.

voltage saturation: When a component has the necessary voltage present to allow it to operate.