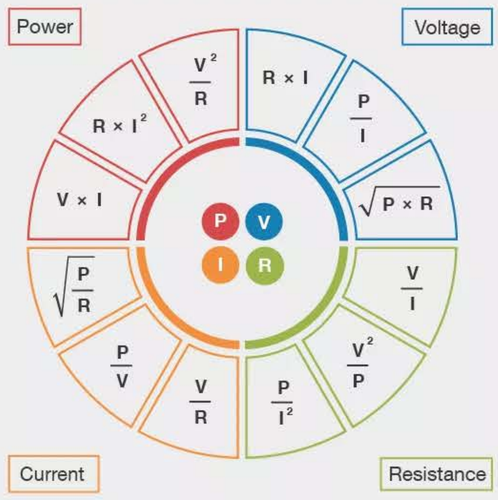
TinkrPostr

ELECTRONICS CHEAT SHEET POSTER

Your quick reference companion in learning, referencing and debugging your electronics projects

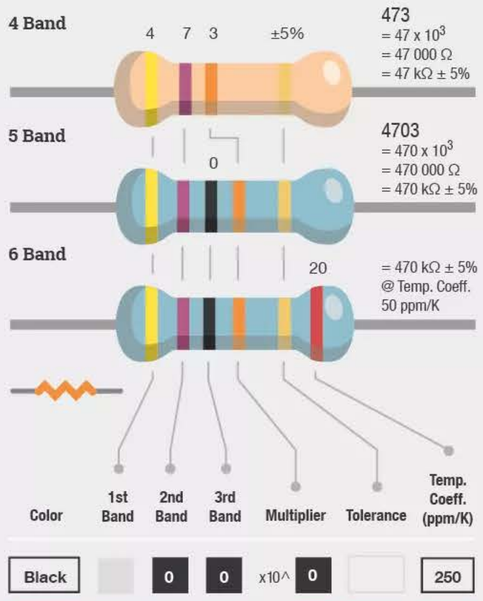
Ohm's Law



American Wire Gauge

AWG No.	Diameter (in)	Diameter (mm)	CS Area. (mm2)	Resistance (Ω/km)	Actual Cross Sect. Size
4/0	.4600	11.68	107.2	.1608	0000
3/0	.4096	10.40	85.03	.2028	000
2/0	.3648	9.266	67.43	.2557	00
1/0	.3249	8.252	53.48	.3224	0
1	.2893	7.348	42.41	.4066	1
2	.2576	6.544	33.63	.5127	2
3	.2294	5.827	26.67	.6465	3
4	.2043	5.189	21.15	.8152	4
5	.1819	4.621	16.77	1.028	5
6	.1620	4.115	13.30	1.296	6
7	.1443	3.665	10.55	1.634	
8	.1285	3.264	8.366	2.061	
9	.1144	2.906	6.634	2.599	
10	.1019	2.588	5.261	3.277	
11	.0907	2.305	4.172	4.132	
12	.0808	2.053	3.309	5.211	
13	.0720	1.828	2.624	6.571	
14	.0641	1.628	2.081	8.286	
15	.0571	1.450	1.650	10.45	
16	.0508	1.291	1.309	13.17	
18	.0403	1.024	.8231	20.95	
20	.0320	.8118	.5176	33.31	
22	.0253	.6438	.3255	52.96	
24	.0201	.5106	.2047	84.22	
26	.0159	.4049	.1288	133.9	
28	.0126	.3211	.08098	212.9	
30	.0100	.2546	.05093	338.6	
32	.00795	.2019	.03203	538.3	
34	.00630	.1601	.02014	856.0	
36	.00500	.1270	.01267	1361	
38	.00397	.1007	.00797	2164	
40	.00314	.0799	.00501	3441	

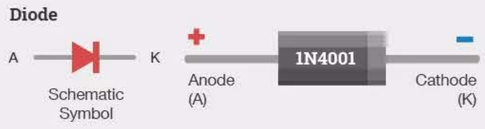
Resistor Color Coding



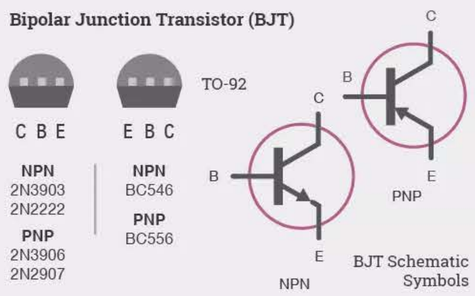
Color	1st Band	2nd Band	3rd Band	Multiplier	Tolerance	Temp. Coeff. (ppm/K)
Black		0	0	x10^0		250
Brown	1	1	1	x10^1	±1%	100
Red	2	2	2	x10^2	±2%	50
Orange	3	3	3	x10^3		15
Yellow	4	4	4	x10^4		25
Green	5	5	5	x10^5	±.50%	20
Blue	6	6	6	x10^6	±.25%	10
Violet	7	7	7	x10^7	±.10%	5
Grey	8	8	8		±.05%	1
White	9	9	9			
Gold				x10^-1	±5%	
Silver				x10^-2	±10%	

Diodes and Transistors

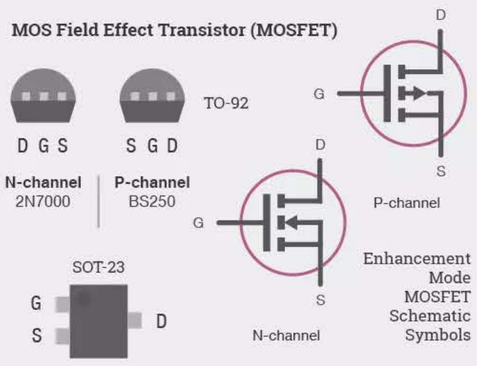
Diode



Bipolar Junction Transistor (BJT)



MOS Field Effect Transistor (MOSFET)



Component pinouts shown are the most common and may still vary per device manufacturer. Refer to the specific device datasheet to be sure.

Capacitor Coding

Common Capacitors

Capacitance Conversion Table

Microfarads (uF)		Nanofarads (nF)		Picofarads (pF)
0.000001 uF	=	0.001 nF	=	1 pF
0.00001 uF	=	0.01 nF	=	10 pF
0.0001 uF	=	0.1 nF	=	100 pF
0.001 uF	=	1 nF	=	1 000 pF
0.01 uF	=	10 nF	=	10 000 pF
0.1 uF	=	100 nF	=	100 000 pF
1 uF	=	1 000 nF	=	1 000 000 pF

Max. Operating Voltage

1H	50 V	2E	250 V
2A	100 V	2G	400 V
2T	150 V	2J	630 V
2D	200 V		

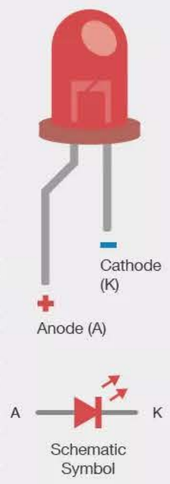
Tolerance

B	±.01 pF	H	±3%
C	±.25 pF	J	±5%
D	±.50 pF	K	±10%
F	±1%	M	±20%
G	±2%	Z	+80% -20%

Light Emitting Diode (LED)

Typical LED Characteristics

Color	Wavelength (nm)	Typical Forward Voltage (V) @ 20 mA
Red	630 - 660	1.8
Orange	605 - 620	2.0
Yellow	585 - 595	2.2
Green	550 - 570	3.5
Blue	430 - 505	3.6
White	450	4.0
Ultraviolet	850 - 940	1.2



Surface Mount Devices (SMDs)

SMD Resistor Markings

3 Digit

473 = 47 x 10³ = 47 000 Ω = 47 kΩ

4 Digit

4702 = 470 x 10² Ω = 47 000 Ω = 47 kΩ

with Radix Point

4R7 = 4.7 Ω

with Radix Point

0R47 = 0.47 Ω

SMD Capacitor Markings

Tantalum

473 = 47 x 10³ pF = 47 nF @ 16V

Electrolytic Capacitor

473 = 47 x 10³ pF = 47 nF @ 16V

Electrical Units

Basic Electrical Units

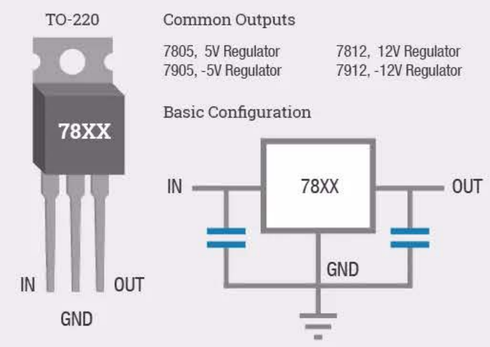
Quantity	Abbrev. / Unit		Quantity	Abbrev. / Unit	
Capacitance	F	Farad	Inductance	H	Henry
Charge	C	Coulomb	Magnetic Flux	Wb	Weber
Current	A	Ampere	Potential	V	Volt
Energy	J	Joule	Power	W	Watt
Force	N	Newton	Resistance	Ω	Ohm
Frquency	Hz	Hertz			

Metric Prefixes

Tera-	T	×10 ¹²	1 000 000 000 000
Giga-	G	×10 ⁹	1 000 000 000
Mega-	M	×10 ⁶	1 000 000
Kilo-	K	×10 ³	1 000
Hecto-	H	×10 ²	100
Deka-	Da	×10 ¹	10
(base)	-	×10 ⁰	1
Deci-	d	×10 ⁻¹	0.1
Centi-	c	×10 ⁻²	0.01
Milli-	m	×10 ⁻³	0.00 1
Micro-	µ	×10 ⁻⁶	0.00 000 1
Nano-	n	×10 ⁻⁹	0.00 000 000 1
Pico-	p	×10 ⁻¹²	0.00 000 000 000 1

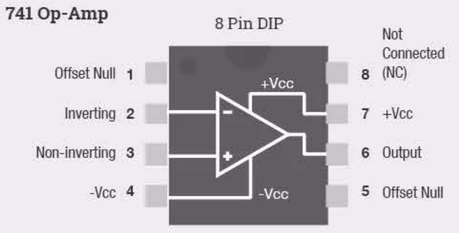
Regulator

LM78XX Regulator

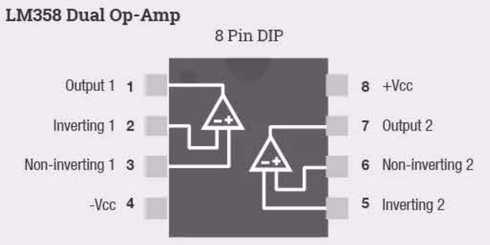


Op-Amp

741 Op-Amp



LM358 Dual Op-Amp

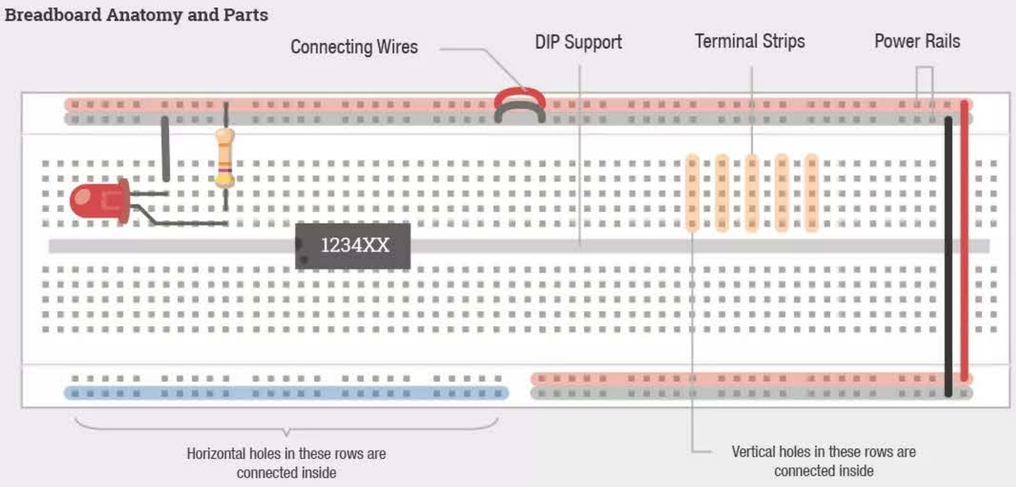


555 IC

555 IC Pinout

The Breadboard

Breadboard Anatomy and Parts



Other Stuffs

Thanks to the Contributors

Richard Crowley
Michael Lebon
Max McMullen
Cate Cannon
John Quinn
Jean François Dupuis
Joseph Johnson
Andrew Goulah

References

Books:
Engineer's Mini-Notebook by Forrest Mims III, Encyclopedia of Electric Components

Web Sources:
dangerousprototypes.com
blog.ricardoarturocabral.com
sizes.com
en.wikipedia.org
venkel.com

Compiled and Illustrated by: Joseph Ricafort

Let's Keep in Touch!

You can drop me some message at any means you prefer:

Email: josephricafortjr@gmail.com
Facebook: facebook.com/joricafort
Follow me! @josephricafort

Made possible by:

FUNDED WITH INDIEGOGO

PRINTFUL

FUNDAMENTALS REFERENCE POSTER

Arduino Programming Cheat Sheet

Primary source: Arduino Language Reference
http://arduino.cc/en/Reference/

Structure & Flow

Basic Program Structure
```
void setup() {
  // Runs once when sketch starts
}
void loop() {
  // Runs repeatedly
}
```

Control Structures
```
if (x < 5) { ... } else { ... }
while (x < 5) { ... }
for (int i = 0; i < 10; i++) { ... }
break;     // Exit a loop immediately
continue;  // Go to next iteration
switch (var) {
  case 1:
    ...
    break;
  case 2:
    ...
    break;
  default:
    ...
}
return x;  // x must match return type
return;    // For void return type
```

Function Definitions
```
<ret. type> <name>(<params>) { ... }
e.g. int double(int x) {return x*2;}
```

Operators

General Operators

=	assignment		
+	add	-	subtract
*	multiply	/	divide
%	modulo		
==	equal to	!=	not equal to
<	less than	>	greater than
<=	less than or equal to		
>=	greater than or equal to		
&&	and	||	or
!	not		

Compound Operators

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

Bitwise Operators

&	bitwise and	|	bitwise or
^	bitwise xor	~	bitwise not
<<	shift left	>>	shift right

Pointer Access

&	reference: get a pointer
*	dereference: follow a pointer

Variables, Arrays, and Data

Data Types

boolean	true | false
char	-128 - 127, 'a' '$' etc.
unsigned char	0 - 255
byte	0 - 255
int	-32768 - 32767
unsigned int	0 - 65535
word	0 - 65535
long	-2147483648 - 2147483647
unsigned long	0 - 4294967295
float	-3.4028e+38 - 3.4028e+38
double	currently same as float
void	i.e., no return value

Strings
```
char str1[8] =
  {'A','r','d','u','i','n','o','\0'};
  // Includes \0 null termination
char str2[8] =
  {'A','r','d','u','i','n','o'};
  // Compiler adds null termination
char str3[] = "Arduino";
char str4[8] = "Arduino";
```

Numeric Constants

123	decimal
0b01111011	binary
0173	octal - base 8
0x7B	hexadecimal - base 16
123U	force unsigned
123L	force long
123UL	force unsigned long
123.0	force floating point
1.23e6	1.23*10^6 = 1230000

Qualifiers

static	persists between calls
volatile	in RAM (nice for ISR)
const	read-only
PROGMEM	in flash

Arrays
```
int myPins[] = {2, 4, 8, 3, 6};
int myInts[6];   // Array of 6 ints
myInts[0] = 42;  // Assigning first
                 // index of myInts
myInts[6] = 12;  // ERROR! Indexes
                 // are 0 though 5
```

Built-in Functions

Pin Input/Output

Digital I/O - pins 0-13 A0-A5
```
pinMode(pin,
  [INPUT, OUTPUT, INPUT_PULLUP])
int digitalread(pin)
digitalWrite(pin, [HIGH, LOW])
```

Analog In - pins A0-A5
```
int analogRead(pin)
analogReference(
  [DEFAULT, INTERNAL, EXTERNAL])
```

PWM Out - pins 3 5 6 9 10 11
```
analogWrite(pin, value)
```

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

Time
```
unsigned long millis()
  // Overflows at 50 days
unsigned long micros()
  // Overflows at 70 minutes
delay(msec)
delayMicroseconds(usec)
```

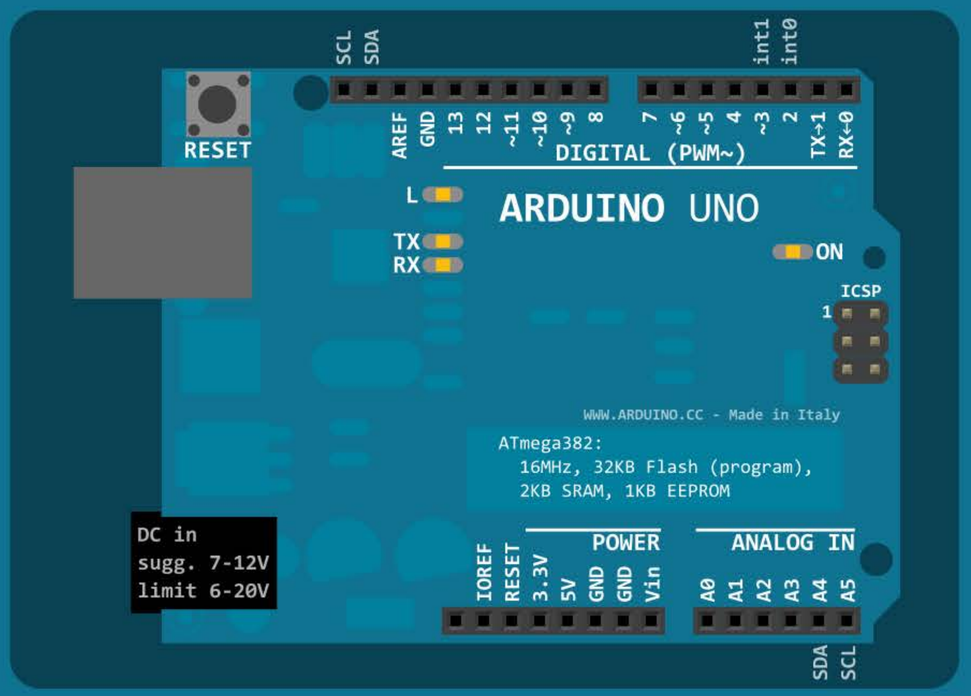
Math
```
min(x, y)   max(x, y)   abs(x)
sin(rad)    cos(rad)    tan(rad)
sqrt(x)     pow(base, exponent)
constrain(x, minval, maxval)
map(val, fromL, fromH, toL, toH)
```

Random Numbers
```
randomSeed(seed) // long or int
long random(max) // 0 to max-1
long random(min, max)
```

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

Type Conversions
```
char(val)       byte(val)
int(val)        word(val)
long(val)       float(val)
```

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



Libraries

Serial - comm. with PC or via RX/TX
```
begin(long speed)  // Up to 115200
end()
int available()  // #bytes available
int read()   // -1 if none available
int peek()   // Read w/o removing
flush()
print(data)    println(data)
write(byte)    write(char * string)
write(byte * data, size)
SerialEvent()  // Called if data rdy
```

SoftwareSerial.h - comm. on any pin
```
SoftwareSerial(rxPin, txPin)
begin(long speed)  // Up to 115200
listen()       // Only 1 can listen
isListening()  // at a time.
read, peek, print, println, write
  // Equivalent to Serial library
```

EEPROM.h - access non-volatile memory
```
byte read(addr)
write(addr, byte)
EEPROM[index]  // Access as array
```

Servo.h - control servo motors
```
attach(pin, [min_uS, max_uS])
write(angle)  // 0 to 180
writeMicroseconds(uS)
   // 1000-2000; 1500 is midpoint
int read()    // 0 to 180
bool attached()
detach()
```

Wire.h - I²C communication
```
begin()     // Join a master
begin(addr) // Join a slave @ addr
requestFrom(address, count)
beginTransmission(addr) // Step 1
send(byte)              // Step 2
send(char * string)
send(byte * data, size)
endTransmission()       // Step 3
int available() // #bytes available
byte receive()  // Get next byte
onReceive(handler)
onRequest(handler)
```


by Mark Liffiton

Adapted from:
- Original: Gavin Smith
- SVG version: Frederic Dufourg
- Arduino board drawing: Fritzing.org