Structure

·void setup() Initialize variables, pin mode, call library functions, etc.

·void loop() Consecutively execute statements within this functions

Constant

·HIGH | LOW Indicates the level of the digital IO port, HIGH indicates a high level (1), and LOW indicates a low level (0).

·INPUT | OUTPUT Indicates the direction of the digital IO port, INPUT indicates the input (high impedance state), and OUTPUT indicates the output (AVR can provide 5V voltage and 40mA current).

·true | false true Indicates true (1), false indicates false (0).

Digital I/O

·pinMode(pin, mode) Digital IO port input and output mode definition function, pin is 0 ~ 13, mode is INPUT or OUTPUT.

·digitalWrite(pin, value) Digital IO port output level definition function, pin is 0 ~ 13, value is HIGH or LOW. For example, define HIGH to drive LEDs.

·int digitalRead(pin) Digital IO port read input level function, pin is 0 ~ 13, value is HIGH or LOW. For example, you can read digital sensors.

Analog I/O

·int analogRead(pin) Analog IO port read function, pin is 0 ~ 5 (Arduino Diecimila is 0 ~ 5, Arduino nano is 0 ~ 7). For example, you can read an analog sensor (10-bit AD, 0 ~ 5V means 0 ~ 1023).

·analogWrite(pin, value) PWM digital IO port PWM output function. The Arduino digital IO port indicates that the PWM IO port can use this function. Pin is 3, 5, 6, 9, 10, 11, and value is 0 to 255. For example, it can be used for motor PWM speed regulation or play music .

Extension I/O

·shiftOut(dataPin, clockPin, bitOrder, value) SPI external IO extension function, we usually use 74HC595 with SPI interface for 8 IO extensions, dataPin is the data port, clockPin is the clock port, bitOrder is the data transmission direction (MSBFIRST high-order first, LSBFIRST low-order first), value indicates the transmission Data (0 ~ 255). In addition, an IO port is required to enable 74HC595 control.

·unsigned long pulseIn(pin, value) This function is used to record the pulse length, return time parameter (us), pin is 0 ～ 13, value is HIGH or LOW. For example, if the value is HIGH, when the pin input is high, start calculating time; when the pin input is low, and stop calculating time and then return to time.

Time function

·unsigned long millis()   Return time function (unit: ms). This function means that when the program runs, it starts timing and returns the recorded parameters. It takes about 50 days for the parameters to overflow.

·delay(ms)   Delay function (unit: ms).

·delayMicroseconds(us)   Delay function (unit: us).

Math function

·min(x, y)   Get the minimum

·max(x, y)   Get the maximum

·abs(x)   Get the absolute value

·constrain(x, a, b)   Constraint function, lower limit a, upper limit b, x must be between a and b.

·map(value, fromLow, fromHigh, toLow, toHigh)   Constraint function, value must be between fromLow and toLow and fromHigh and toHigh.

·pow(base, exponent)   Square function, exponent to power of base.

·sq(x)   Calculate the square

·sqrt(x)   Square root

Trigonometric function

·sin(rad)

·cos(rad)

·tan(rad)

Random number function

·randomSeed(seed) Random number port definition function, seed indicate read analog port analogRead (pin) function.

·long random(max)   Random number function that returns data greater than or equal to 0, and less than max.

·long random(min, max)   Random number function that returns data greater than or equal to min, and less than max.

External interrupt function

·attachInterrupt(interrupt, , mode)   External interrupts can only use digital IO ports 2 and 3; interrupt indicates the initial 0 or 1 of the interrupt port, which indicates a function function; mode: LOW low-level interrupt, CHANGE interrupts when there is a change, RISING rising edge interrupt, FALLING falling edge interrupt .

·detachInterrupt(interrupt)   Interrupt switch, interrupt = 1 indicates on, interrupt = 0 indicates off.

Interrupt enable function

·interrupts()   Enable interrupt

·noInterrupts()   Disable interrupt

Serial Receive-Send Function

·Serial.begin(speed)   The serial port defines the baud rate function, and speed indicates the baud rate, such as 9600, 19200, and so on.

·int Serial.available()   Determine the buffer status.

·int Serial.read()   Read the serial port and return the received parameters.

·Serial.flush()   Clear the buffer.

·Serial.print(data)   Serial output data.

·Serial.println(data)   The serial port outputs data with a carriage return character.

/************Arduino programming language library file*************/

Official library file

·EEPROM - EEPROM Read and write libraries

·Ethernet - Ethernet controller library

·LiquidCrystal - LCD control library

·Servo - Servo control library

·SoftwareSerial - Any digital IO port analog serial library

·Stepper - Stepper motor library

·Wire - TWI/I2C Bus library

·Matrix – LED matrix control library

·Sprite – LED Matrix image processing control library

Unofficial library file

·DateTime - a library for keeping track of the current date and time in software.

·Debounce - for reading noisy digital inputs (e.g. from buttons)

·Firmata - for communicating with applications on the computer using a standard serial protocol.

·GLCD - graphics routines for LCD based on the KS0108 or equivalent chipset.

·LCD - control LCDs (using 8 data lines)

·LCD 4 Bit - control LCDs (using 4 data lines)

·LedControl - for controlling LED matrices or seven-segment displays with a MAX7221 or MAX7219.

·LedControl - an alternative to the Matrix library for driving multiple LEDs with Maxim chips.

·Messenger - for processing text-based messages from the computer

·Metro - help you time actions at regular intervals

·MsTimer2 - uses the timer 2 interrupt to trigger an action every N milliseconds.

·OneWire - control devices (from Dallas Semiconductor) that use the One Wire protocol.

·PS2Keyboard - read characters from a PS2 keyboard.

·Servo - provides software support for Servo motors on any pins.

·Servotimer1 - provides hardware support for Servo motors on pins 9 and 10

·Simple Message System - send messages between Arduino and the computer

·SSerial2Mobile - send text messages or emails using a cell phone (via AT commands over software serial)

·TextString - handle strings

·TLC5940 - 16 channel 12 bit PWM controller.

·X10 - Sending X10 signals over AC power lines