

List of Functions

This list contains all of the functions in m3pi.h. Other fundamental programming structures, such as if statements and while statements, will be explained as they are being used. Refer to the specific worksheet for those. This sheet only contains the useful functions.

Functions

#include mbed.h

#include m3pi.h

#include m3pimaze.h

These three are not functions as such, but **libraries**. They contain lists of functions which can be used in your program. These are essential to any program you use. (you will only need one of m3pi.h or m3pimaze.h, only the maze programs use m3pi.h).

m3pi m3pi; or **m3pi m3pi(p23, p9, p10);** This is needed in all programs, just remember to include it if you are starting from scratch.

m3pi.reset(); Forces a hardware reset on the m3pi robot.

m3pi.left_motor(speed); This allows you to set the speed of the left motor only (between -1 and 1)

m3pi.right_motor(speed); Same as above but for right motor.

m3pi.forward(speed); drives both motors forward at the same speed.

m3pi.backward(speed); drives both motors backward at the same speed. Does not need negative numbers.

m3pi.left(speed); turns the robot left on the spot.

m3pi.right(speed); turns the robot right on the spot.

m3pi.stop(); stops the robot on the spot.

m3pi.pot_voltage(); reads the voltage on the potentiometer.

m3pi.battery(); reads the voltage on the battery.

m3pi.line_position(); reads the position of the detected line.

m3pi.sensor_auto_calibrate(); automatically calibrates the optical sensors.

m3pi.calibrate(); sets calibration manually to the current settings.

m3pi.reset_calibration (); clears the current calibration settings.

m3pi.leds(integer); Writes to the 8 LEDs (must be referenced as BusOut leds(LED1 etc...)), also note, writes in binary, so a value of 3 would turn on the first 2.

m3pi.locate(int x, int y); moves the cursor on the 8x2 LCD display, remember the numbers start at 0.

m3pi.cls(); clears the LCD.

m3pi.putc(int c); sends a character directly to the m3pi serial interface.

m3pi.getc(); receives a character directly from the m3pi serial interface.

m3pi.printf("example"); sends the word example to the LCD screen at the current cursor position.

m3pi.readsensor(sensor); reads the individual sensor values and puts the values from 0 to 1000 into an array called sensor. 0 is completely light and 1000 is completely dark. This function only works with the m3pimaze.h library.

The functions in mbed.h are more fundamental, and won't be as useful for this project. If you need to know these however, there is lots of information at <http://mbed.org/handbook/Homepage>.

Useful knowledge

Speed of rotation: 720 degrees per second at half speed.

Forward/Backward: 470mm per second at half speed.

While (1) {chunk} will make the chunk of programming repeat forever.

If(condition) {chunk} will make the chunk run if the condition is met.

Else if (condition) {chunk} will make the chunk run, but only if the first if condition is not met.

You can use more than one condition. For example::

if(condition1 && condition2) {chunk} will run if only both conditions are met.

Else {chunk} will make that chunk run if the above if condition(s) are not met, so in all other cases.