



//Robot Car Demo by Craig Turner, to program sequential movements. Date: 31/10/2012

```
/*  A simple(ish) sketch to help you play with your Arduino Motor Shield
    Maybe it could make your Arduino robot dance */
```

```
// Motor Shield is on a Robot car controlled by Arduino Uno. Note it is advisable to cool the chip
// using a heatsink (or fan or both) on the shield.
```

// Sketch is provided "As is" with no guarantees, or support from the Author.
// Help with Arduino and shields can be found by joining the forum on the Arduino website:
//<http://arduino.cc/en/>

```
#include <Stepper.h>

// Enable (PWM) outputs
#define ENA_PIN 3 // left
#define ENB_PIN 11 // right

// Direction outputs
#define DIRA_PIN 12 //left
#define DIRB_PIN 13//right

int mySpeed = 0;
```

```

int maxSpeed = 250;

float timeDo = 0;// time to do operation or command

/*Here, you can write your instructions to the robot in the array program[].
Each single char command takes a data value eg {'s', 255} means: at 255 [speed can be a number 0 -> 255]
Other commands take a time setting. eg {'f', 1} means: go forward for 1 second. Floating point numbers are
allowed for timed activities

The command list (feel free to expand on these):
/*
    h Stop
    f Forward
    b Backwards
    c Clockwise rotate
    a Anticlockwise rotate
    p Procrastinate (short forward and backward moves)
    w Wiggle (a wiggle is a series of short a and c rotations)

*/

//The program - alter as required: See You tube video: "Gampa Geek's Robot Tries Some Moves"
//Note: you can wrap each program in a method and call them in void loop()
//      see void dance() at the end

float program[] = {'s', 240, //speed 240

    'h', 5, 'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2, //forwards / backwards & wiggle x4
    'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2,
    'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2,
    'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2,

    'a', 1, 's', 120, 'a', 1, 's', 50, 'a', 1, // spin anticlockwise and slow down

    's', 240, 'c', 1, // spin clockwise and speed up

    'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2, //forwards / backwards & wiggle x3
    'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2,
    'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2};
// program loops forever

byte programLen = 0; // set program length

```

```

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

    // Configure all motor outputs for now
    pinMode(ENA_PIN, OUTPUT);

    pinMode(ENB_PIN, OUTPUT);

    pinMode(DIRA_PIN, OUTPUT);

    pinMode(DIRB_PIN, OUTPUT);
    // Both motors off for now
    analogWrite(ENA_PIN, 0);
    analogWrite(ENB_PIN, 0);

    allStop(0,0); // all stop
}

void loop ()
{
    int programLen = sizeof(program) / sizeof(float); // find size of array called program

    for (int i=0; i < programLen; i++)
    {
        // start of switch-case using commands *****
        switch (int(program[i]))
        {
            case 's': // SPEED
                mySpeed = program[i+1];

                if (mySpeed > maxSpeed) //
                {
                    mySpeed = maxSpeed; //do not allow speed above maxSpeed
                }
                break;

            case 'f': //FORWARD FOR TIME
                allAhead(mySpeed, program[i+1] * 1000); // * 1000 for seconds
                break;

            case 'b': //BACKWARD FOR TIME

```

```

    allBackwards(mySpeed, program[i+1] * 1000);
    break;

case 'c': //Rotate clockwise FOR TIME
    clockwise(mySpeed, program[i+1] * 1000);
    break;

case 'a': //Rotate anticlockwise FOR TIME
    anticlockwise(mySpeed, program[i+1] * 1000); //program * 1000
    break;

case 'h': //Halt FOR TIME
    allStop(0, program[i+1] * 1000);
    break;

case 'p': //procrastinate
    procrast(mySpeed, program[i+1]);
    break;

case 'w': // wiggle = short rotations
    wiggle(mySpeed, program[i+1] ); // speed, number of wiggles)
    break;

} // end of program switch case*****

} // end of program for loop

} // end of loop

// movement methods=====

void allStop(int speedMove, int timeDo) // stop all motors
{
    // Both motors off
    analogWrite(ENA_PIN, 0);
    analogWrite(ENB_PIN, 0);
    delay(timeDo);
}

void allAhead(int speedMove, int timeDo) // go forwards
{

```

```

    digitalWrite(DIRA_PIN, HIGH); // HIGH = Forwards, LOW = backwards
    digitalWrite(DIRB_PIN, HIGH);
    analogWrite(ENA_PIN, speedMove);
    analogWrite(ENB_PIN, speedMove);
    delay(timeDo);
}

void allBackwards(int speedMove, int timeDo) // go backwards
{
    digitalWrite(DIRA_PIN, LOW); // HIGH = Forwards, LOW = backwards
    digitalWrite(DIRB_PIN, LOW);
    analogWrite(ENA_PIN, speedMove);
    analogWrite(ENB_PIN, speedMove);
    delay(timeDo);
}

void anticlockwise(int speedMove, int timeDo)
{
    digitalWrite(DIRA_PIN, LOW); // HIGH = Forwards, LOW = backwards
    digitalWrite(DIRB_PIN, HIGH);
    analogWrite(ENA_PIN, speedMove);
    analogWrite(ENB_PIN, speedMove);
    delay(timeDo);
}

void clockwise(int speedMove, int timeDo)
{
    digitalWrite(DIRA_PIN, HIGH); // HIGH = Forwards, LOW = backwards
    digitalWrite(DIRB_PIN, LOW);
    analogWrite(ENA_PIN, speedMove);
    analogWrite(ENB_PIN, speedMove);
    delay(timeDo);
}

void procrast(int speedMove, int procrastNum) // move back and forth for a bit ie procrastinate
{
    for (int i = 0; i < procrastNum; i++)
    {
        allBackwards( speedMove, 0.25 * 1000); //backwards
        allAhead(speedMove , 0.25 * 1000); // forward
        allBackwards( speedMove, 0.25 * 1000); //backwards
        allAhead(speedMove , 0.25 * 1000); // forward
    }
}

void wiggle(int speedMove, int wigNum) // wigNum "wiggles"

```

```

{
  for (int i = 0; i < wigNum; i++)
  {
    anticlockwise(speedMove, 0.2 * 1000); // a wiggle is a rotation of 0.2 sec
    clockwise(speedMove, 0.2 * 1000); // ie 0.2 * 1000ms
    clockwise(speedMove, 0.2 * 1000); // ie 0.2 * 1000ms
    anticlockwise(speedMove, 0.2 * 1000); // a wiggle is a rotation of 0.2 sec
  }
}

void yourMoves (/*parameters*/)
{
  //code here

}

void dance()
{
  float program[] = {'s', 240, //speed 240

    'h', 5, 'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2, //forwards / backwards & wiggle x4
    'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2,
    'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2,
    'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2,

    'a', 1, 's', 120, 'a', 1, 's', 50, 'a', 1, // spin anticlockwise and slow down

    's', 240, 'c', 1, // spin clockwise and speed up

    'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2, //forwards / backwards & wiggle x3
    'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2,
    'f', 0.5, 'b', 0.5, 'f', 0.5, 'b', 0.5, 'w', 2};
  // program loops forever

}

```