#define COUNTER 30// ms

#define cycle 1/3

#define hull\_Pin1 2

#define hull\_Pin2 3

#define hull\_Pin3 4

#define A 1

#define B 2

#define C 3

#define A2 4

#define B2 5

#define C2 6

String x = "", x\_past = "";

int y = 0,y\_past = 0, newcommand = 0;

int count = -1,phase\_past = 0, initial\_phase=0,Pin1,Pin2,Pin3;

int pulse1, pulse2, pulse3,mark1 = 0,mark2 = 0,phase,index;

int pulse1count = 0, pulse2count = 0,prepulse1count;

int\* list = (int\*)calloc(120,sizeof(int));

int counter = COUNTER, counter1 = 0;

double c1 = COUNTER\*cycle, c2 = COUNTER\*cycle, c3 = COUNTER\*cycle;

int e1=0,e2=0,e3=0;

int k = 1, forward = 1;

void setup(){

//set pins as outputs

pinMode(9, OUTPUT);

pinMode(10, OUTPUT);

pinMode(11, OUTPUT);

pinMode(6, OUTPUT);

cli();

TCCR1A = 0;

TCCR1B = 0;

TCNT1 = 250;

OCR1A = 1000;

TCCR1B |= (1 << WGM12);

TCCR1B |= (1 << CS11) | (1 << CS10);

TIMSK1 |= (1 << OCIE1A);

Serial.begin(9600);

pinMode(hull\_Pin1, INPUT);

pinMode(hull\_Pin2, INPUT);

pinMode(hull\_Pin3, INPUT);

list[110] = B2;

list[10] = C2;

list[11] = C;

list[1] = A2;

list[101] = A;

list[100] = B;

pulse1 = digitalRead(hull\_Pin1);

pulse2 = digitalRead(hull\_Pin2);

pulse3 = digitalRead(hull\_Pin3);

index = 100\*pulse1+10\*pulse2+pulse3;

initial\_phase = list[index];

SetupPin();

sei();

TIMSK1 = 0;

//Pin1 = 12 - initial\_phase;

// Pin2 = 11 - (initial\_phase+1)%3;

// Pin3 = 11- (initial\_phase)%3;

}//end setup

ISR(TIMER1\_COMPA\_vect){//timer1 interrupt 1Hz toggles pin 13 (LED)

//generates pulse wave of frequency 1Hz/2 = 0.5kHz (takes two cycles for full wave- toggle high then toggle low)

counter--;

counter1++;

if(counter <= 0) counter = COUNTER;

if(e1 == 1) c1--;

if(e2 == 1) c2--;

if(e3 == 1) c3--;

if(c1 <= 0){

c1 = COUNTER\*cycle;

e1 = 0;

}

if(c2 <= 0){

c2 = COUNTER\*cycle;

e2 = 0;

}

if(c3 <= 0){

c3 = COUNTER\*cycle;

e3 = 0;

}

digitalWrite(6, HIGH);

if(counter == COUNTER){

digitalWrite(Pin1, HIGH);

e1 = 1;

}

if(counter == COUNTER\*2/3){

digitalWrite(Pin2, HIGH);

e2 = 1;

}

if(counter == COUNTER/3){

digitalWrite(Pin3, HIGH);

e3 = 1;

}

if(e1 == 0){

digitalWrite(Pin1, LOW);

}

if(e2 == 0){

digitalWrite(Pin2, LOW);

}

if(e3 == 0){

digitalWrite(Pin3, LOW);

}

}

void loop(){

/\* if(initial\_phase == 1){

Pin1 = 11;

Pin2 = 9;

Pin3 = 10;

}

if(initial\_phase == 2){

Pin1 = 9;

Pin2 = 11;

Pin3 = 10;

}

if(initial\_phase == 3){

Pin1 = 9;

Pin2 = 10;

Pin3 = 11;

}\*/

//Pin1 = 9;

//Pin2 = 11;

//Pin3 = 10;

readcommand();

//motor(9,10,11);

// digitalWrite(Pin3, HIGH);

// if(count>=6) cli();

if(newcommand&&count>=y) {

reset\_counter();

newcommand = 0;

initial\_phase = phase;

y\_past = 0;

Serial.print("stop");

}

reading\_hull();

// Serial.print(initial\_phase);

//Serial.print("\n");

/\* Serial.print(pulse1);

Serial.print(pulse2);

Serial.print(pulse3);

Serial.print("phase is");

Serial.print(phase);\*/

}

void SetupPin(){

if(initial\_phase%3 == A){

Pin1 = 9;

Pin2 = 10+forward;

Pin3 = 11-forward;

}

if(initial\_phase%3 == B){

Pin1 = 10;

Pin2 = 11-2\*forward;

Pin3 = 9+2\*forward;

}

if(initial\_phase%3 == 0){

Pin1 = 11;

Pin2 = 9+forward;

Pin3 = 10-forward;

}

}

void readcommand(){

if(Serial.available()) x = Serial.readString();

if(x!="" && x!= x\_past){

if(x == "F" || x =="f") {

forward = 1;

x\_past = x;

}

else if(x == "B" || x == "b") {

forward = 0;

x\_past = x;

}

else{

if(x.length()>1)

y = (int)x[1]+(int)x[0]\*10-528;

else y = (int)x[0] - 48;

if(y != y\_past){

x\_past = x;

TIMSK1 = 0;

SetupPin();

Serial.print(y);

count = 0;

newcommand = 1;

y\_past = y;

TIMSK1 = bit (OCIE1A);

}

}

}

}

void reading\_hull(){

int difference = 0;

pulse1 = digitalRead(hull\_Pin1);

pulse2 = digitalRead(hull\_Pin2);

pulse3 = digitalRead(hull\_Pin3);

index = 100\*pulse1+10\*pulse2+pulse3;

phase = list[index];

if(phase == 0) difference = 0;//Serial.print("error ")

else if(phase!=phase\_past&&phase < A2){

count++;

// Serial.print(count);

difference = phase - phase\_past;

// if(difference == 1 || difference == -2)

// if(forward)Serial.print("direction error ");

// else if(!forward)Serial.print("direction error ");

phase\_past=phase;

}

}

void reset\_counter(){

TIMSK1 = 0;

counter = COUNTER;

c1 = COUNTER\*cycle;

c2 = COUNTER\*cycle;

c3 = COUNTER\*cycle;

e1 = 0;

e2 = 0;

e3 = 0;

}

/\*void setup\_table(){

int i;

for( i = 0; i < max\_key; i++){

list[i] = ini\_x + i\*delta\_x;

}

}\*/