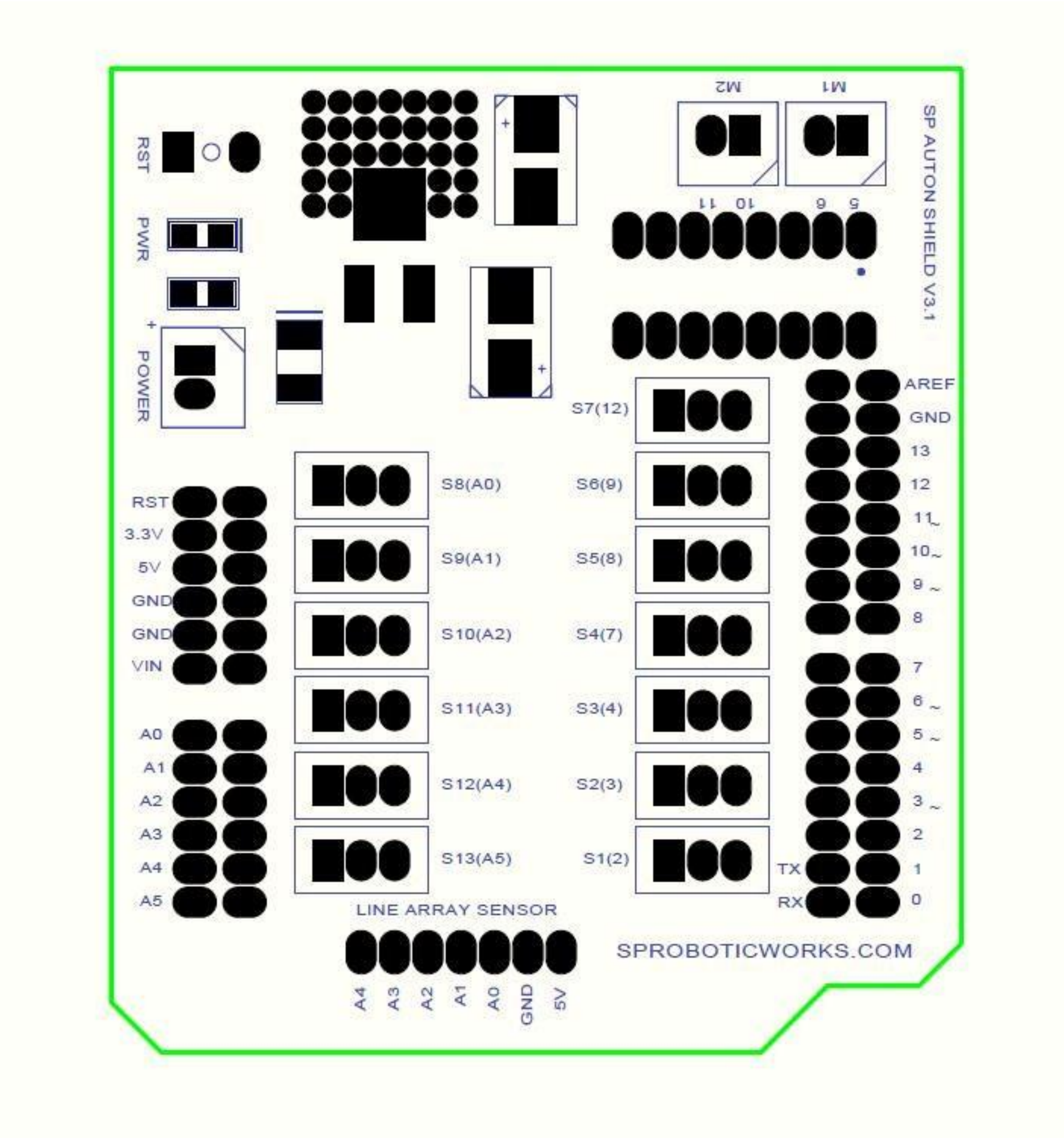
KiDOBOTiKZ

Engineering Real Engineers

ADVANCED LEVEL NOTES (LEVEL 7, 8 and 9)

AUTON SHIELD



Date:

Obstacle Avoider (with 3 sensors):

LS	CS	RS	LM		RM	
			T1	T2	T1	T2
0	0	0	0	1	0	1
0	0	1	1	0	0	1
1	0	0	0	1	1	0
0	1	0	1	0	1	0
			0	1	1	0
1	1	0	1	0	1	0
			0	1	1	0
1	0	1	1	0	1	0
			0	1	1	0
0	1	1	1	0	1	0
			1	0	0	1
1	1	1	1	0	1	0
			0	1	1	0

Program:

```
int m1t1=5,m1t2=6,m2t1=10,m2t2=11,sen1=2,sen2=3,sen3=4;
int s1,s2,s3;
void setup( )
{
        pinMode(m1t1,OUTPUT);
        pinMode(m1t2,OUTPUT);
        pinMode(m2t1,OUTPUT);
        pinMode(m2t2,OUTPUT);
        pinMode(sen1,INPUT);
        pinMode(sen2,INPUT);
        pinMode(sen3,INPUT);
}
```

```
void loop( )
{
        s1=digitalRead(sen1);
        s2=digitalRead(sen2);
        s3=digitalRead(sen3);
        if(s1==LOW && s2 ==LOW && s3==LOW)
        {
        analogWrite(m1t1,0);
        analogWrite(m1t2,255);
        analogWrite(m2t1,0);
        analogWrite(m2t2,255);
        }
        else if(s1==LOW && s2 ==LOW && s3==HIGH)
        {
        analogWrite(m1t1,255);
        analogWrite(m1t2,0);
        analogWrite(m2t1,0);
        analogWrite(m2t2,255);
        }
        else if(s1==HIGH && s2 ==LOW && s3==LOW)
        {
        analogWrite(m1t1,0);
        analogWrite(m1t2,255);
        analogWrite(m2t1,255);
        analogWrite(m2t2,0);
        }
        else if(s1==LOW && s2 ==HIGH && s3==HIGH)
        {
        analogWrite(m1t1,255);
        analogWrite(m1t2,0);
        analogWrite(m2t1,255);
        analogWrite(m2t2,0);
        delay(200);
        analogWrite(m1t1,255);
        analogWrite(m1t2,0);
        analogWrite(m2t1,0);
```

```
    analogWrite(m2t2,255);
    delay(300);
    }
    else
    {
    analogWrite(m1t1,255);
    analogWrite(m1t2,0);
    analogWrite(m2t1,255);
    analogWrite(m2t2,0);
    delay(200);
    analogWrite(m1t1,255);
    analogWrite(m1t2,0);
    analogWrite(m2t1,0);
    analogWrite(m2t2,255);
    delay(300);
    }
}
```

Date:

Light Follower (with 3 sensors):

LS	CS	RS	LM		RM	
			T1	T2	T1	T2
0	0	0	0	1	1	0
1	0	0	1	0	0	1
0	0	1	0	1	1	0
0	1	0	0	1	0	1
1	1	0	0	1	0	1
0	1	1	0	1	0	1
1	0	1	0	1	0	1
1	1	1	0	1	0	1

Program:

Not for students: Similar to the obstacle avoider type….(delays are not there)

Date:

Pit Avoider (with 3 sensors):

LS	CS	RS	LM		RM	
			T1	T2	T1	T2
1	1	1	0	1	0	1
1	1	0	1	0	0	1
0	1	1	0	1	1	0
1	0	1	1	0	1	0
			1	0	0	1
0	1	0	1	0	1	0
			1	0	0	1
1	0	0	1	0	1	0
			1	0	0	1
0	0	1	1	0	1	0
			0	1	1	0
0	0	0	1	0	1	0
			1	0	0	1

Program:

Not for students: Similar to the obstacle avoider type....(delays should be inserted when two operations come into action ie moving back as well as turning right – except first three cases)

Date:

Line Tracer (with 3 line-array sensors):

LS	CS	RS	LM		RM	
			T1	T2	T1	T2
0	1	0	0	1	0	1
1	0	0	0	0	0	1
0	0	1	0	1	0	0
1	1	0	0	150	0	1
0	1	1	0	1	0	150
1	1	1	0	1	0	0
0	0	0	0	1	0	0
1	0	1	0	1	0	0

Program:

Not for students: Similar to the obstacle avoider type….(delays are not there in a line tracer)

Left-Wall Follower (with 2 sensors):

LS	CS	LM		RM	
		T1	T2	T1	T2
1	0	0	1	0	0
0	0	0	0	0	1
0	1	0	1	1	0
		Delay			
1	1	0	1	1	0
		Delay			

Program:

Not for students: Similar to the obstacle avoider type but there are only two sensors here….(delays are there even though two operations are not there because the robot has to take a perfect 90 degrees when a wall comes in front, the delay should be included)

Date:

Autonomous Sumo Robot:

Concept:

5 sensors will be used (3 for sensing if there is an obstacle, 2 for sensing the line)

Line sensors – one in the front and the other at the back

The three obstacle sensors have to sense the obstacle and make the robot go towards it (Light follower – with a small change, instead of making the robot turn to search for the obstacle, the robot should move forward searching for obstacle)

If the front line detector detects line, the robot should move back and turn

If the back line detector detects line, the robot should move forward

NOT FOR STUDENTS:

If possible, this can also be done – when the front line detector detects line, the robot should move back, then check if there is no line and then only turn

```
If(front_line == HIGH)
{
//go back with a delay
If (front_line == LOW)
{
//turn any direction with a delay
}
```

Date:

Computer Controlled Robot:

Concept:

When

A is sent, move left

D → move right

W → move forward

S → stop

X → move back

M → increase speed (in steps of 20)

N → decrease speed (in steps of 20)

Program:

```
int m1t1=5,m1t2=6,m2t1=10,m2t2=11;
int value;
int speed=255;
void setup( )
{
pinMode(m1t1,OUTPUT);
pinMode(m1t2,OUTPUT);
pinMode(m2t1,OUTPUT);
pinMode(m2t2,OUTPUT);

Serial.begin(9600);
}
```

```
void loop( )
{
        value = Serial.read();
        switch(value)
        {
        case 'w':
                analogWrite(m1t1,0);
                analogWrite(m1t2,speed);
                analogWrite(m2t1,0);
                analogWrite(m2t2,speed);
                break;
        case 'a':
                analogWrite(m1t1,speed);
                analogWrite(m1t2,0);
                analogWrite(m2t1,0);
                analogWrite(m2t2,speed);
                break;

        ……………………………similarly for d,s,x (no delays anywhere)


        case 'n':
                speed=speed-20;
                break;
        case 'm':
                speed=speed+20;
                break;


        }
}
```

Date:

Auto-calibration Robot:

Concept:

Start of the program: the robot should calibrate itself for the surrounding lighting conditions:

1. Start Turning in any one direction
2. Read the value and store in a variable (light)
3. Add the value to a variable named sum (sum=sum+light)
4. Give a delay of 100
5. Repeat the steps 2, 3 and 4 a hundred times
6. Stop the robot
7. Calculate the average (sum/100);
8. This average is the surrounding lighting condition, now set the threshold (ie fire value to about 150points above this average)
9. Now the robot has to spray water only if it is close to the fire, ie the light-sensing value will be close to 1023.
   So set the value to 900

Loop Program: The robot keeps rotating –searching for fire. When fire value is detected in any sensor, the robot turns towards it (just like light follower). When in any of the sensor the water value is reached, the robot stops and the water starts spraying).

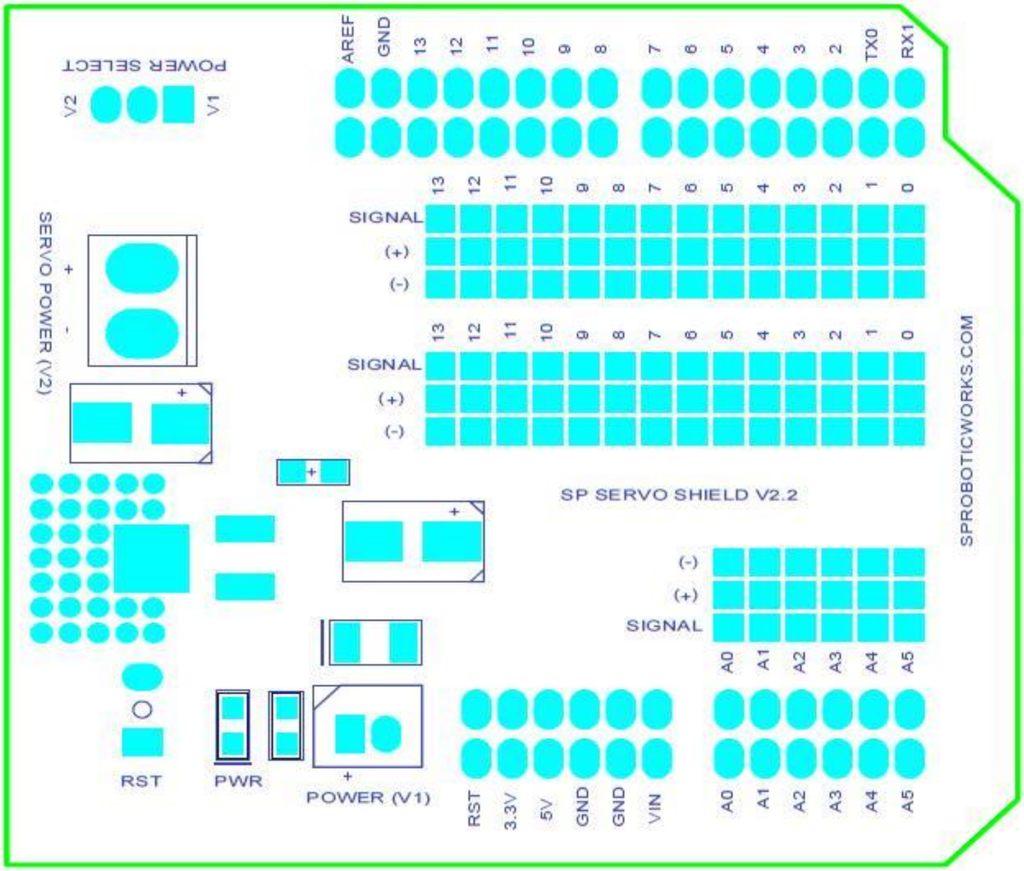
Program:

```
int m1t1=5,m1t2=6,m2t1=10,m2t2=11,sen=A0;
int light, fire, water, i, sum=0, average=0;
void setup( )
{
        pinMode(m1t1,OUTPUT);
        pinMode(m1t2,OUTPUT);
        pinMode(m2t1,OUTPUT);
        pinMode(m2t2,OUTPUT);
        pinMode(sen,INPUT);

        analogWrite(m1t1,255);
        analogWrite(m1t2,0);
        analogWrite(m2t1,0);
        analogWrite(m2t2,255);

        for(i=0; i<100; i=i+1)
        {
        light=analogRead(sen);
        sum = sum+light;
        }
        average=sum/100;
        fire=average+150;
        water=900;
}
void loop( )
{
        light = analogRead(sen);
        if(light>fire && light<water)
        {
                analogWrite(m1t1,0);
                analogWrite(m1t2,200);
                analogWrite(m2t1,0);
                analogWrite(m2t2,200);
        }
```

```
    if(light<fire)
    {
        analogWrite(m1t1,200);
        analogWrite(m1t2,0);
        analogWrite(m2t1,0);
        analogWrite(m2t2,200);
    }
    if(light<water)
    {
        analogWrite(m1t1,0);
        analogWrite(m1t2,0);
        analogWrite(m2t1,0);
        analogWrite(m2t2,0);
    }

}
```

Date:

LEVEL 8

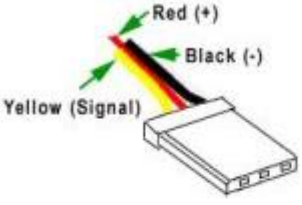

POWER SELECT
V2
V1
AREF
GND
13
12
11
10
9
8
7
6
5
4
3
2
TX0
RX1
1
0
SIGNAL
(+)
(-)
SERVO POWER (V2)
+
-
SP SERVO SHIELD V2.2
SPROBOTICWORKS.COM
A0
A1
A2
A3
A4
A5
RST
PWR
POWER (V1)
3.3V
5V
VIN

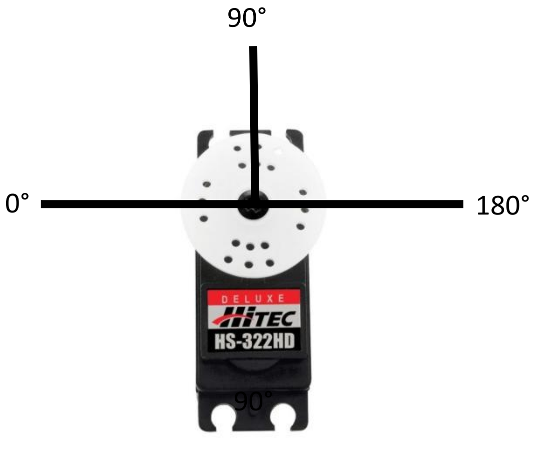
SERVO MOTOR

A servo motor is made up of a normal DC motor, a gear set, a position-sensing device (usually a potentiometer that is, a volume control knob), and a control circuit. Unlike a normal DC Motor, the servo motor holds its shaft position at a particular degree and hence is useful for pick and place applications and other walking or arm robots

When the servo motor receives a control signal that represents the *desired output position*, it applies power to its DC motor until its shaft turns to that position. It uses the potentiometer to determine the rotational position of the shaft.

Servo motor has 3 wires: supply, ground, and control. The power source must be constantly given and signal can be given whenever required.





EXP1_SERVO_WRITE

```
#include<Servo.h>
Servo s1;
void setup()
{
        s1.attach(2);
}
void loop()
{
        s1.write(90);
}
```

EXP2_SERVO_SWEEP

```
#include<Servo.h>
Servo s1;
int i;
void setup()
{
        s1.attach(2);
}
void loop()
{
        for(i=0; i<180; i=i+1)
        {
                s1.write(i);
                delay(5);
        }
        for(i=180; i>0; i=i-1)
        {
                s1.write(i);
                delay(5);
        }
}
```

Date:

EXP3_ARM_INTIALIZE

```
#include<Servo.h>
Servo s1;
Servo s2;
Servo s3;
Servo s4;
Servo s5;

void setup()
{
        s1.attach(2);
        s2.attach(3);
        s3.attach(4);
        s4.attach(5);
        s5.attach(6);
}
void loop()
{
        s1.write(30);
        s2.write(120);
        s3.write(120);
        s4.write(90);
        s5.write(0);
}
```

EXP4_PICK_AND_PLACE

```
#include<Servo.h>
Servo s1;
Servo s2;
Servo s3;
Servo s4;
Servo s5;
void setup()
{
        s1.attach(2);
        s2.attach(3);
        s3.attach(4);
        s4.attach(5);
        s5.attach(6);
}
void loop()
{
        s1.write(120);
        s2.write(90);
        s3.write(120);
        s4.write(90);
        s5.write(0);
        delay(1000);

        s5.write(180);
        delay(1000);

        s1.write(60);
```

```
    delay(1000);

    s2.write(180);
    s3.write(90);
    s4.write(0);
    delay(1000);

    s5.write(0);
    delay(1000);

    s2.write(90);
    delay(1000);
}
```

EXP5_SERVO_SLOWMOVE

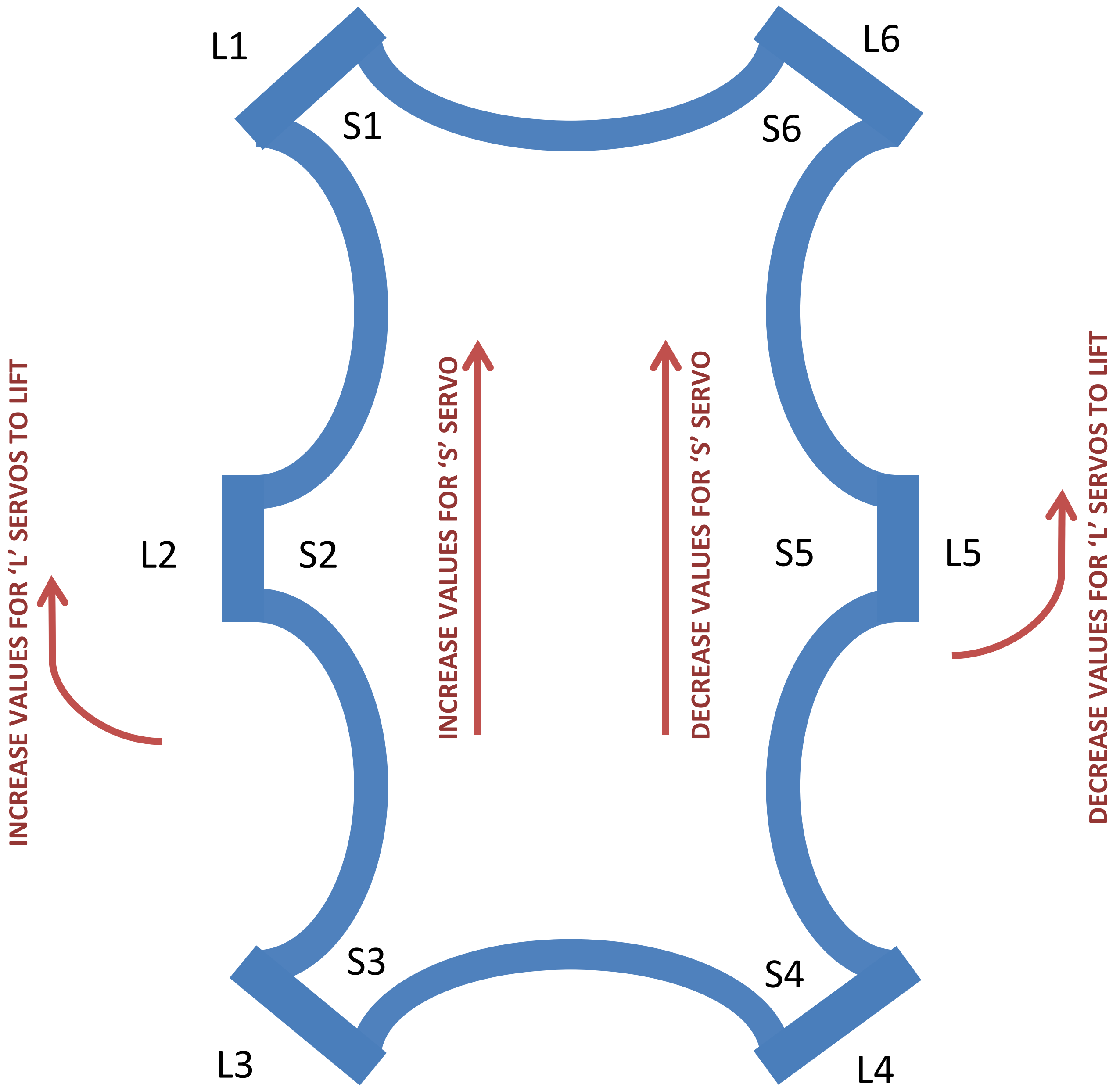
```
#include<VarSpeedServo.h>
VarSpeedServo s1;
VarSpeedServo s2;
VarSpeedServo s3;
VarSpeedServo s4;
VarSpeedServo s5;
void setup()
{
        s1.attach(2);
        s2.attach(3);
        s3.attach(4);
        s4.attach(5);
        s5.attach(6);
}
void loop()
{
        s1.slowmove(120,200);
        s2.slowmove(90,200);
        s3.slowmove(120,200);
        s4.slowmove(90,200);
        s5.slowmove(0,200);
        delay(1000);

        s5.slowmove(180,200);
        delay(1000);

        s1.slowmove(60,200);
        delay(1000);


        s2.slowmove(180,200);
        s3.slowmove(90,200);
        s4.slowmove(0,200);
        delay(1000);


        s5.slowmove(0,200);
        delay(1000);


        s2.slowmove(90,200);
        delay(1000);
}
```

EXP6_SENSE_AND_PICK

```
#include<VarSpeedServo.h>
VarSpeedServo s1;
VarSpeedServo s2;
VarSpeedServo s3;
VarSpeedServo s4;
VarSpeedServo s5;
int sensor=12, value;
void setup()
{
        s1.attach(2);
        s2.attach(3);
        s3.attach(4);
        s4.attach(5);
        s5.attach(6);
        pinMode(sensor,INPUT);
}
void loop()
{
        value=digitalRead(sensor);
        s1.slowmove(120,200);
        s2.slowmove(90,200);
        s3.slowmove(120,200);
        s4.slowmove(90,200);
        s5.slowmove(0,200);
        delay(1000);
```

```
    if(value==HIGH)
    {
    s5.slowmove(180,200);
    delay(1000);

    s1.slowmove(60,200);
    delay(1000);

    s2.slowmove(180,200);
    s3.slowmove(90,200);
    s4.slowmove(0,200);
    delay(1000);

    s5.slowmove(0,200);
    delay(1000);

    s2.slowmove(90,200);
    delay(1000);
    }
}
```

LEVEL 9


L1
L6
S1
S6
INCREASE VALUES FOR 'L' SERVOS TO LIFT
INCREASE VALUES FOR 'S' SERVO
DECREASE VALUES FOR 'S' SERVO
DECREASE VALUES FOR 'L' SERVOS TO LIFT
L2
S2
S5
L5
S3
S4
L3
L4

Date:

EXP7_HEXAPOD_STANDING

```
#include<VarSpeedServo.h>
VarSpeedServo s1;
VarSpeedServo s2;
VarSpeedServo s3;
VarSpeedServo s4;
VarSpeedServo s5;
VarSpeedServo s6;
VarSpeedServo L1;
VarSpeedServo L2;
VarSpeedServo L3;
VarSpeedServo L4;
VarSpeedServo L5;
VarSpeedServo L6;

void setup()
{
        s1.attach(1);
        s2.attach(2);
        s3.attach(3);
        s4.attach(4);
        s5.attach(5);
        s6.attach(6);
        L1.attach(7);
        L2.attach(8);
        L3.attach(9);
        L4.attach(10);
        L5.attach(11);
        L6.attach(12);
}
void loop()
{
        s1.slowmove(120,200);
        s2.slowmove(90,200);
        s3.slowmove(60,200);
        s4.slowmove(120,200);
        s5.slowmove(90,200);
        s6.slowmove(60,200);

        L1.slowmove(90,200);
        L2.slowmove(90,200);
        L3.slowmove(90,200);
        L4.slowmove(90,200);
        L5.slowmove(90,200);
        L6.slowmove(90,200);
}
```

EXP8_HEXAPOD_WALKING

Before void loop, everything is same as standing

```
void loop()
{
        //code for standing
        //then lift leg L2 (10 deg), move the S2 forward (30deg) and place L2 back on the ground
        L2.slowmove(100,200);
        delay(200);

        s2.slowmove(120,200);
        delay(200);

        L2.slowmove(90,200);
        delay(200);

//then lift leg L4 (10 deg), move the S4 forward (30deg) and place L4 back on the ground
        L4.slowmove(80,200);
        delay(200);

        s2.slowmove(60,200);
        delay(200);

        L2.slowmove(90,200);
        delay(200);
//then lift leg L1 (10 deg), move the S1 forward (20deg) and place L1 back on the ground

//then lift leg L6 (10 deg), move the S6 forward (20deg) and place L1 back on the ground

//move the S3 forward (30deg) and move the S4 forward (30deg)

}
```

EXP9_HEXAPOD_LEFT_TURN

Before void loop, everything is same as standing

```
void loop()
{
        //code for standing
        //then lift leg L2 (10 deg), move the S2 backward (50deg) and place L2
back on the ground
        L2.slowmove(100,200);
        delay(200);

        s2.slowmove(40,200);
        delay(200);

        L2.slowmove(90,200);
        delay(200);

//then lift leg L4 (10 deg), move the S4 forward (50deg) and place L4 back on
the ground
        L4.slowmove(80,200);
        delay(200);

        s2.slowmove(40,200);
        delay(200);

        L2.slowmove(90,200);
        delay(200);
//then lift leg L1 (10 deg), move the S1 backward (50deg) and place L1 back on
the ground
//then lift leg L6 (10 deg), move the S6 forward (50deg) and place L1 back on the ground

//move the S3 backward (50deg) and move the S4 forward (50deg)

}
```

EXP10_HEXAPOD_RIGHT_TURN

Before void loop, everything is same as standing

```
void loop()
{
        //code for standing
        //then lift leg L2 (10 deg), move the S2 forward (50deg) and place L2
back on the ground
        L2.slowmove(100,200);
        delay(200);

        s2.slowmove(140,200);
        delay(200);

        L2.slowmove(90,200);
        delay(200);

//then lift leg L4 (10 deg), move the S4 backward (50deg) and place L4 back on
the ground
        L4.slowmove(80,200);
        delay(200);

        s2.slowmove(140,200);
        delay(200);

        L2.slowmove(90,200);
        delay(200);
//then lift leg L1 (10 deg), move the S1 forward (50deg) and place L1 back on
the ground
//then lift leg L6 (10 deg), move the S6 backward (50deg) and place L1 back on
the ground

//move the S3 forward (50deg) and move the S4 backward (50deg)

}
```

Date:

EXP11_HEXAPOD_MOVING_AROUND

Before void loop, everything is same as standing

```
int i;
void loop()
{
        for(i=0; i<5; i=i+1)
        {
                stand();
                delay(200);
                straight();
        }
        for(i=0; i<5; i=i+1)
        {
                stand();
                delay(200);
                left();
        }
        for(i=0; i<5; i=i+1)
        {
                stand();
                delay(200);
                straight();
        }
        for(i=0; i<5; i=i+1)
        {
                stand();
                delay(200);
                right();
        }
}
void stand()
{
        //code for standing
}
void straight()
{
        //code for straight
}
void left()
{
        //code for left
}
void right()
{
        //code for right
}
```

EXP12_HEXAPOD_WALKING_2

Before void loop, everything is same as standing

```
void setup()
{
After attach code for servo…
//code for standing (with a delay of 500)
}
void loop()
{
        //then lift leg L2, L4 and L6 (10 deg), move the S2, S4 and S6 forward (30deg) and place L2, L4 and L6 back on the ground
        L2.slowmove(100,200);
        L4.slowmove(80,200);
        L6.slowmove(80,200);
        delay(200);

        s2.slowmove(120,200);
        s4.slowmove(90,200);
        s6.slowmove(30,200);
        delay(200);

        L2.slowmove(90,200);
        L4.slowmove(90,200);
        L6.slowmove(90,200);
        delay(200);
//then while lifting leg L1, L3 and L5 (10 deg), move the S2, S4 and S6 back to
original positions.  Then move S1, S3 and S5 forward and place L1, L3 and L5
back on the ground

        L1.slowmove(100,200);
        L3.slowmove(100,200);
        L5.slowmove(80,200);
        s2.slowmove(90,200);
        s4.slowmove(120,200);
        s6.slowmove(60,200);
        delay(200);

        s1.slowmove(150,200);
        s3.slowmove(90,200);
        s5.slowmove(60,200);
        delay(200);

        L2.slowmove(90,200);
        L4.slowmove(90,200);
        L6.slowmove(90,200);
        delay(200);

//then bring move the S1, S3 and S5 back to original positions without delay so
that the action coincides with the next three legs lifting while in void loop()

        s1.slowmove(120,200);
        s3.slowmove(60,200);
        s5.slowmove(90,200);
}
```