```
/*
  TheDirectCode.ino

  Linear actuator with joystick control

  The Circuit:
  - Arduino nano or uno
  - Actuonix actuator
  - 3 axis Joystick with button

 */

#include <Servo.h>
Servo myServo;
#define PIN_SERVO (5)
int d = 5;
int delayMS = 200;
int i = 50;
int VRy = 0;
int lastState = 1;

//servo setup and function that controlls the servo in microseconds instead of degrees to achive higher accuracy
void SetStrokePerc(float strokePercentage)
{
  if ( strokePercentage >= 1.0 && strokePercentage <= 99.0 ) // clamps stroke percentage 1-99 so no actuator strain
  {
    int usec = 1000 + strokePercentage * ( 2000 - 1000 ) / 100.0 ;
    myServo.writeMicroseconds( usec );
  }
}
void SetStrokeMM(int strokeReq,int strokeMax)
{
  SetStrokePerc( ((float)strokeReq) / strokeMax );
}
//more setup/serialbegin
void setup()
{
  myServo.attach(PIN_SERVO);
  Serial.begin(9600);
}

void loop() {
 // reading the joystick

 VRy = analogRead(A1);
 delay(20);
 int State = map(VRy, 0, 1023, 0, 2); // sends a value 0 if under x axis, representing downshift, and 2 representing upshift. (correctly using the state method)
 if(State != lastState) // if statement to only process the state if it is different than the last state
 {

 if(State == 0) // if joystick is in the downshift position(ie, p = 0)
 {
  i-=d; // subtracts d from stroke percentage of actuator
   SetStrokePerc(i);// sets to the new stroke percentage
   delay(delayMS/2);// delay so you get a shift and not just continuous motion
 }
 if(State == 2)// if the joystick is below its mapped value of 55, then the actuator position is increased by d
 {
  i+=d; // adds d from stroke percentage of actuator
   SetStrokePerc(i); // sets to the new stroke percentage
   delay(delayMS/2); // delay so you get a shift and not just continuous motion
  }
  lastState = State;
   delay(10);
 }
 SetStrokePerc(i);
 //print values in serial to check everything while using
  Serial.print("\tVRry = ");
  Serial.print(VRy);
  Serial.print("\ti = ");
  Serial.println(i);
  Serial.print("\State = ");
  Serial.println(State);
  Serial.print("\LastState = ");
  Serial.println(lastState);

}
```