- **Software part:**
  We have used the below code in the RPi to run the linear actuator

```python
# Importing the required libraries.
import RPi.GPIO as GPIO
from time import sleep
import vlc
import multiprocessing

# Set warnings as False.
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)

# Reading the alarm clock audio.
vlc_instance = vlc.Instance("--input-repeat=999")
player = vlc_instance.media_player_new()
song = vlc_instance.media_new("/home/pi/Music/audio.mp3")

player.set_media(song)
player.audio_set_volume(100)
player.play()

# GPIO pins for actuator positive and negative terminals
r2 = 40
r1 = 38

# Defining functions for setting up and working of the GPIO pins for the actuator
def set_input_pin(pin):
    GPIO.setup(pin, GPIO.IN)
def set_output_pin(pin):
    GPIO.setup(pin, GPIO.OUT)
    GPIO.output(pin, GPIO.LOW)
```

```
def high(pin):
    GPIO.output(pin, GPIO.HIGH)
def low(pin):
    GPIO.output(pin, GPIO.LOW)

# Function for the forward movement of the linear actuator.
def fwd(t):
    low(r2)
    low(r1)
    high(r2)
    sleep(t)
    low(r1)
    low(r2)

# Function for the backward movement of the linear actuator.
def rev(t):
    low(r2)
    low(r1)
    high(r1)
    sleep(t)
    low(r1)
    low(r2)

# Testing function
# for i in [r1,r2]:
#     GPIO.setup(i, GPIO.OUT)
#     GPIO.output(i, GPIO.LOW)

```

```
# GPIO setup for the alarm clock pin with Rpi
GPIO.setup(17,GPIO.IN, pull_up_down=GPIO.PUD_UP)
# GPIO.setup(17,GPIO.HIGH)

# Main Function for the complete working of the prototype.
while True:
    inpt=GPIO.input(17)
    if inpt== False:
        player.pause()
        #sleep(10)
        player.stop()
        GPIO.cleanup()
        GPIO.setmode(GPIO.BOARD)
        for i in [r1,r2]:
            GPIO.setup(i, GPIO.OUT)
            GPIO.output(i, GPIO.LOW)
        fwd(26)
        sleep(8)
        fwd(6)
        sleep(8)
        fwd(8)
        sleep(8)
        rev(41)
        break
```