

Gesture Controlled Hexapod

Archit Sharma, Sai Pranay, Samyu Kamtam

Abstract

This project focuses on the creation and execution of a hexapod, a robotic system with six legs that is operated by finger movements using flex sensors and an accelerometer. The Basic Stamp 2 microcontroller powers the system, which analyzes data from two sensors to understand the user's hand gestures and actions. Flex sensors mounted on the user's fingers track the angles of bending, while the accelerometer monitors changes in hand orientation, facilitating accurate control of the robot's motions. The hexapod is programmed to execute a series of predetermined dance routines in reaction to user commands, allowing for interaction through natural hand gestures. This project showcases the possibilities of intuitively controlling robotic systems via wearable sensors, effectively integrating motion and gesture-based inputs for lively and coordinated movements.

Table of Contents

Abstract	1
Table of Contents	1
Introduction	1
Related work	2
Design and working principle	2
Mathematics	3
Hexapod sequences	3
Encoding State values for hexapod sequence	4
Prototyping	5
Cost analysis for mass production	5
The estimated cost of this solution is on par with advanced wheelchairs in today's market, but our solution provides more features.	5

Results	5
Final set up:	6
Next Steps	6
Conclusions	6
References	6
Appendix	7
Circuit diagram	7
Gesture to Hexapod mapping	7

Introduction

In recent times, the combination of wearable sensors and robotics has created new opportunities for seamless human-robot interaction. A particularly noteworthy method is utilizing natural hand gestures to control a robot, which can offer a more intuitive and user-friendly interface. This project is centered on the creation and execution of a hexapod, a six-legged robot that is operated by the movement of two fingers. The robot is driven by a Basic Stamp 2 microcontroller and employs a blend of flex sensors and an accelerometer to track the motion and position of the fingers. The flex sensors are responsible for detecting the bending of the fingers, while the accelerometer captures shifts in hand orientation, enabling users to guide the robot's actions with great accuracy.

Hexapods are widely used in a variety of areas like precision positioning, motion simulation, and navigation in complex terrains. Hexapods present great stability due to their six-legged structure and move with high precision. However, their complex design and control systems mean high manufacturing costs which increase further with precision requirements.

The hexapod can execute a series of predefined dance movements in response to the input from the two fingers. This straightforward yet efficient control system

showcases the potential of employing minimal sensors to develop an engaging and interactive robotic experience. By relying on just two fingers for control, the system provides an intuitive method to direct the robot, making it straightforward for users to engage with and command the bot in real-time. The project underscores the expanding possibilities of merging sensor technologies and robotics to create gesture-based systems for both entertainment and practical uses.

Related work

Gesture controlled robots have gained significant popularity in today's world. Our research shows the different ways of coupling human machine interaction for intuitive control and smooth movement. Here we review the different works on gesture controlled hexapods along with their benefits and drawbacks.

Gesture-Controlled Hexapods have been designed using ATOM ESP32 Pico and IMU modules to detect and interpret hand movements. A user-friendly interface is provided making the human machine interaction naturally acquirable and hence reducing the users learning curve. In addition, since the design is wireless, user mobility and operational flexibility is improved. However, this method has reported that the technology is susceptible to signal interference and so makes it difficult to achieve accurate gesture recognition. This inturn impacts system reliability and control.

Intel's CEO demonstrated a hexapod that could be controlled using wearable devices. These wearable devices use accelerometers and gyroscopes to provide an ergonomic interface for real-time user interaction. Nevertheless, real-time performance was compromised due to latency in gesture recognition. The gesture

set was also limited, restricting the range of functionalities.

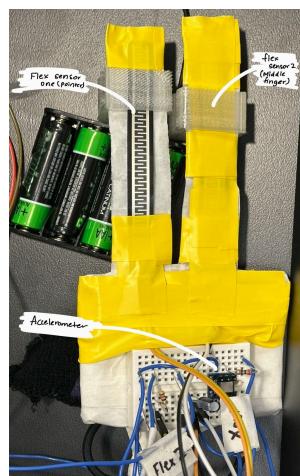
Other advancements in hexapod control use new algorithms for improved stability and autonomous behaviour. However the concern of energy consumption and high processing power remains.

Design and working principle

We use flex sensors, and MX2125 Memsic accelerometer as the sensors and 9g micro servos as the actuator. The flex sensors provide a direct feedback to the BS2 based on bending of finger (index and middle) and can be integrated easily with the microcontroller using input mapping. Although, wear and tear can degrade their performance and it may need to be customly calibrated for individual users. The MX2125 was chosen to measure tilt and acceleration due to its reliability and sensitivity, but it required noise filtering and calibration for accurate gesture recognition.

Using this input from the flex sensor and Memsic accelerometer, the 9g micro servos are actuated. The servos are lightweight, compact and provide adequate torque.

The flex sensors are mounted on the index and middle finger positions on a hand mount while the accelerometer is placed on the dorsal surface of the hand. An input of the finger position and fist orientation is taken from the hand mount to the BS2 which maps the gesture to a predefined sequence of movements. This sequence is then sent to the servos on the hexapod and the hexapod performs the sequence until the next gesture is read by the microcontroller.



The gesture control hand mount is made to enhance user experience by leveraging natural hand movements for hexapod control,

reducing cognitive load, but it requires robust software algorithms for accurate gesture recognition.

The hexapod is a 12 DOF mechanism, having 2 servo motors (2 DOF) per leg. A third servo can be added to each leg making it an 18 DOF manipulator and achieving increased movement for each leg.

Mechanical Design - Hexapod



Mathematics

Servo states are stored in a vector $\text{servo}(i)$ in the form of quantised values from 0-15. The values are converted from 0-15 using formula(1) to PWM.

$$\text{servo position} * \frac{8534}{15} + 500 = \text{t_on for servo PMW} \dots \text{formula(1)}$$

Servo control

	Min value	Max value
θ	-90	90
θ_{on}	0	15
PWM	500	1000

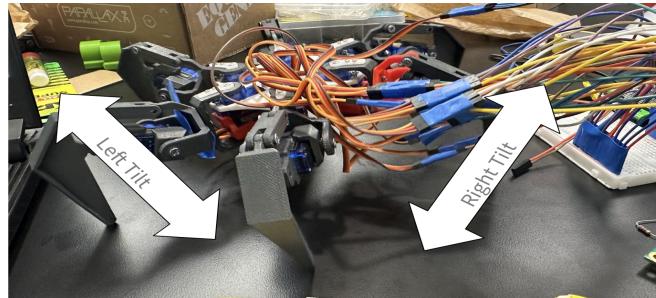
Speed control:

If $\text{speed} * \text{direction} = 0$ then $\theta = \theta + \text{speed}$

If $\text{speed} * \text{direction} = 1$ then $\theta = \theta - \text{speed}$

Speed control: Servo position is updated with speed where the direction of the speed is determined by the speed direction bit.

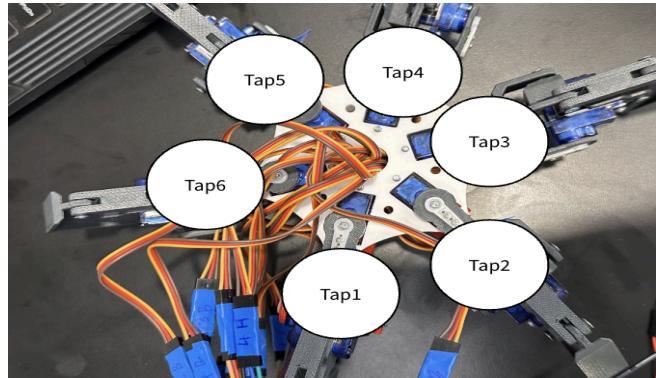
Hexapod sequences



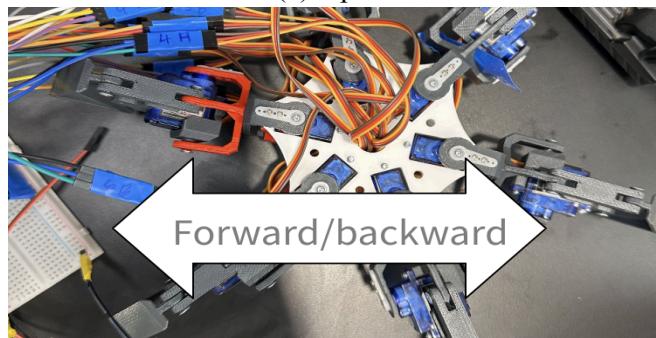
(a) Swagger



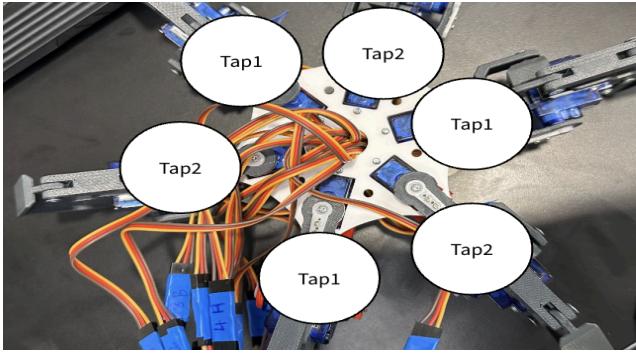
(b) Squat



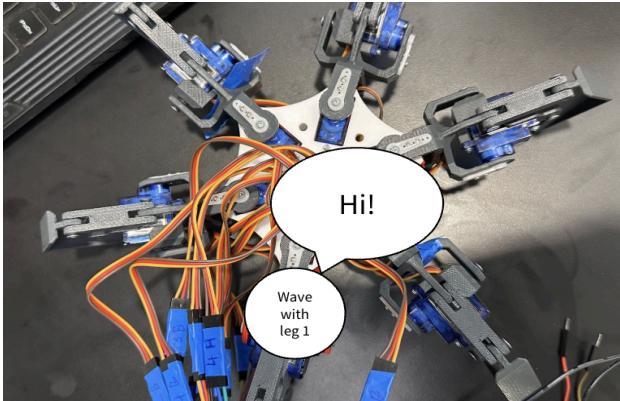
(c) Tap



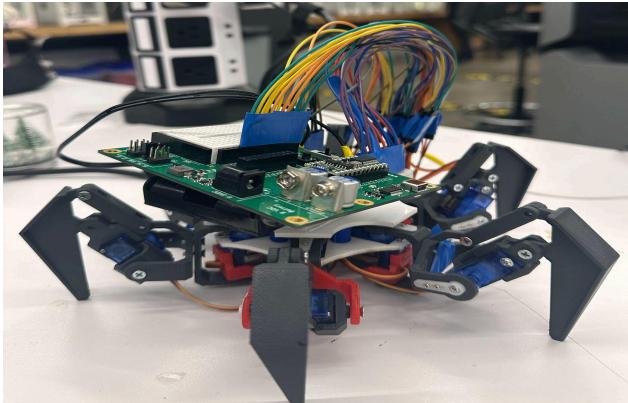
(d) Jiggy



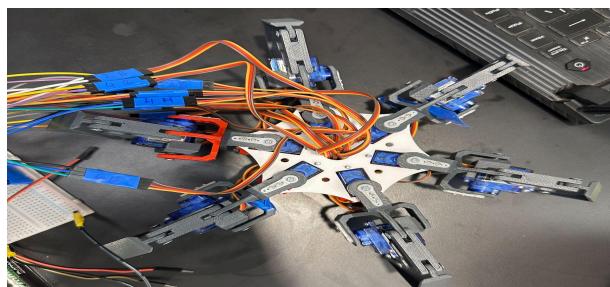
(e) Alternate Tap



(f) "Hi"



(g) Kill



(h) Rest

(i) Speed increase: when the gesture to increase speed is performed - speed of sequence is incremented by 3 times the current value.

(j) Speed reset: when the gesture to decrease speed is performed - speed of sequence is reset to original speed.

Pseudocode for algorithm

Algorithm 1 Gesture-Controlled Hexapod Movement

```

1: Initialize:
2: Define constants: frame_rate, rc_pause, speed_factor ← 1
3: Initialize variables:
4: theta[12] ← servo angles
5: speed[12] ← direction flags for servos
6: Sensor inputs: flex_index, flex_mid, mems_x, mems_y
7: Sensor results: flex_index_result, x_result, etc.
8: sense_state ← encoded sensor signal
9: Initialize flags: squat_flag, swagger_flag, etc.
10: while true do                                ▷ Main loop
11:   OFF
12:   if t20 = 0 then
13:     SENSE
14:   end if
15:   SEQUENCE
16:   MOVE
17:   Increment ticker t
18:   Pause(frame_rate)
19: end while

```

Encoding State values for hexapod sequence

`flex_index_sign` = 0 or 1(not flexed, flexed)

`flex_mid_sign` = 0 or 1(not flexed, flexed)

`x_sign` = 0, 1 or 2(left tilt, neutral, right tilt)

`y_sign` = 0, 1, or 2(up, neutral, down)

`sense_state` = (`flex_index_sign` * 18) +
 (`flex_mid_sign` * 9) + (`x_sign` * 3) +
 (`y_sign`)

Prototyping

Component	Cost
BS2 with Board of education	\$100
9g Servo motors	\$45
3D print	\$14
Flex sensors	\$20
MX2125 accelerometer	\$15
Miscellaneous	\$15
Bill-of-material	
Total Prototype Cost	\$210

Cost analysis for mass production

- A. Direct Costs
 - a. Materials (structural, electronical, manufacturing)
 - b. Labour (Assembly line, quality control)
- B. Indirect costs (factory overhead, packaging, shipping, storage, marketing and distribution)
- C. Research and Development costs
- D. Tooling and equipment costs (3D printing, molds, custom tools)
- E. Compliance and Certification (regulatory approval and safety certifications)
- F. Manufacturing costs, operational costs, quality assurance

Cost component	Cost
Materials	\$2,000
Actuators and sensors	\$800
Manufacturing Labour	\$500
Tooling	\$100
Compliance and testing	\$300
Marketing and distribution	\$200
Shipping and Packaging	\$100
Overhead	\$300
Total	\$4300
Advanced Wheelchair market price	\$4365

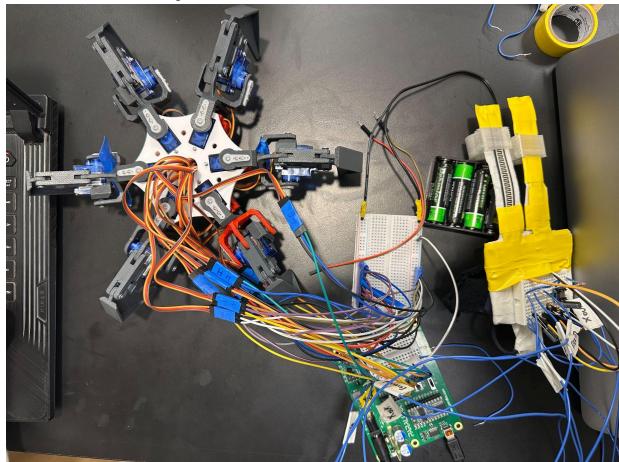
The estimated cost of this solution is on par with advanced wheelchairs in today's market, but our solution provides more features.

Results

We have successfully created the following motions and are able to trigger them using the gesture control hand:

1. Swagger
2. Squat
3. "Hi"
4. Jiggy
5. Tap
6. Alternate Tap
7. Speed increase
8. Speed reset
9. Kill button
10. Rest

Final set up:



References

<https://onlinelibrary.wiley.com/doi/full/10.1002/rob.22186>

<https://pmc.ncbi.nlm.nih.gov/articles/PMC9589936/>

<https://acrome.net/post/what-are-hexapod-robots-used-for>

<https://www.popsci.com/intels-ceo-unleashes-gesture-controlled-spiderbots/>

https://www.thingiverse.com/thing:1021540#google_vignette

Basic Analog and Digital Version 1.3

BASIC Stamp Syntax and Reference Manual Version 2.2

What's a Microcontroller? Version 3.0

Next Steps

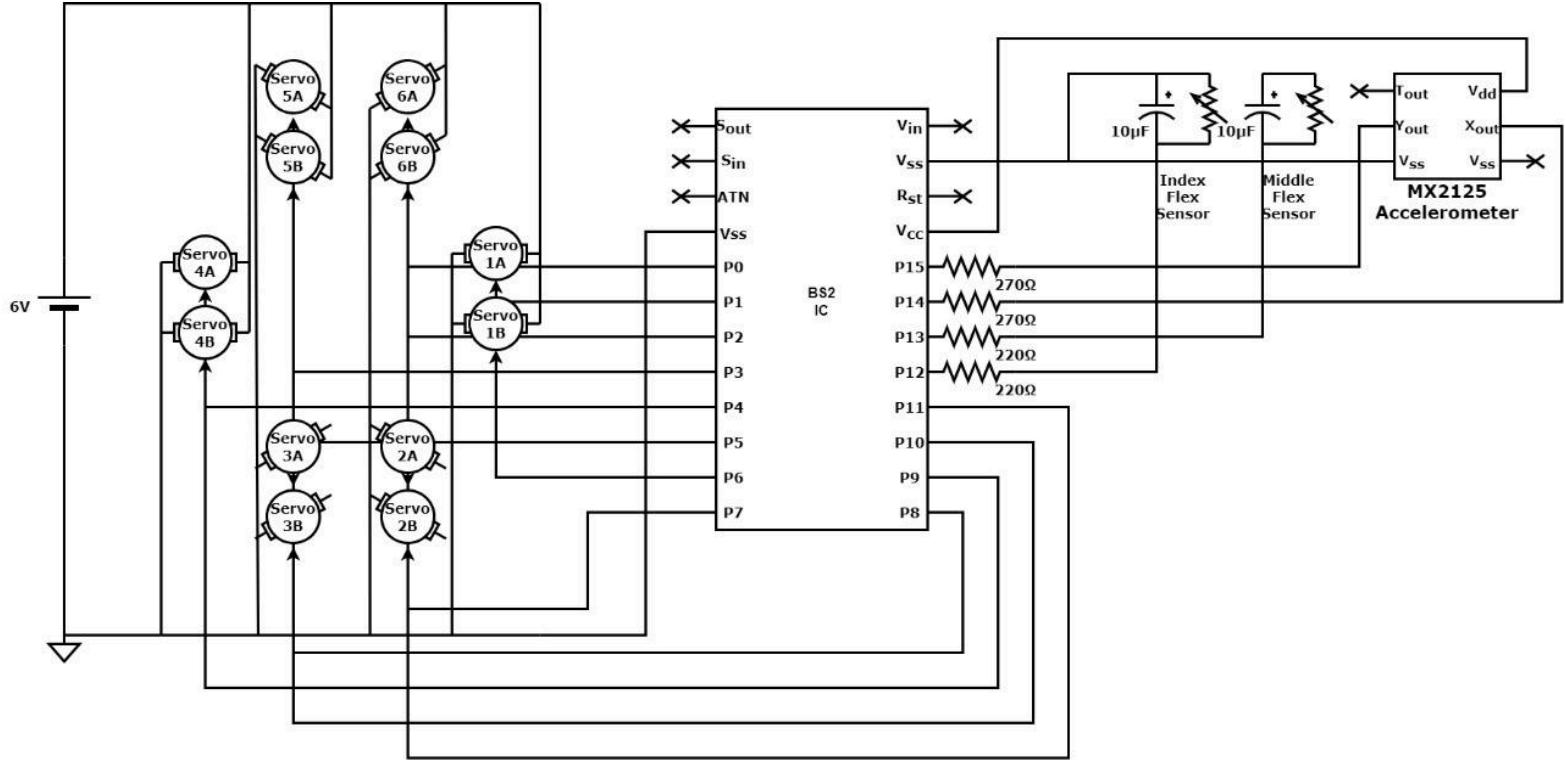
- Working walk and turn sequences using inverse kinematics and dynamics
- Make an 18 DOF hexapod with hand end effector for enhanced assistance
- Live size with strong actuators for load bearing

Conclusions

We urge to revolutionize assistive technology by enhancing mobility and self-expression for individuals with disabilities. Our solution combines stability and flexibility of the hexapod with a user friendly, gesture based control, reducing cognitive load. We faced challenges with respect to system complexity and energy efficiency.

Appendix

Circuit diagram



Gesture to Hexapod mapping

State name	Flex index	Flex middle	x acc	y acc	State value
Swagger	1	0	2	1	25
Squat	1	0	1	1	22
"Hi"	0	1	1	2	14
Jiggy	1	1	1	2	32
Tap	0	1	1	1	13
Alternate Tap	0	1	2	1	16
Speed increase	0	0	1	2	5
Speed decrease	0	0	1	0	3
Kill	1	1	0	1	28
Rest	0	0	1	1	4

