BAT BOT LAUNCHER

Group 39 - Abhishek Bhandari (anb4), Kousthubh Dixit (kmdixit2), Vyom Thakkar (vnt2)
ECE 445 Proposal- Spring 2020
TA : Jonathon Hoff

1. Introduction

1.1 Objective:

Problem statement: We are working on a project that was pitched by Jonathan Hoff. Jonathan and his research group developed a bio-inspired robotic flapping-wing bat robot that mimics the agility and efficiency of bats using silicone membrane wings. The bat bot launcher used presently does not have a mechanism to control the initial launch condition of the bat. It is important to control the initial condition of launch because the trajectory of the robot is highly dependent upon the initial launch condition.

Proposed solution:
We are developing the automated launcher release mechanism for the robot bat that will allow the user to control the timing delay associated with the launch as well as choose the trajectory that he/she requires the bat bot to take, and the launcher will launch the bat bot such that it takes the chosen trajectory.

We will use seven sensors that are vertically stacked such that each position corresponds to a specific wing position. This vertically stacked column of sensors will be placed on the side of the right wing of the bat bot. The lowest sensor will correspond to the lowest wing position and the highest sensor will correspond to the highest wing position possible. All the sensors in the middle will correspond to the respectively desired launch positions. The wings operate with a frequency of 8.5Hz. So, in $t = 1/f = 0.117647s$ the wing completes one oscillation. The wings move across an angle of 60 degrees from one extreme position to the other. Vertically they traverse 22 cm from one end to the other and back.

In the first flap of the wing, we will start recording data from our sensor output, and match all the timestamps of when the sensor outputs become high with the arrival of the wing at each of those positions. All these times would be relative to the top-most position of the wing. We will then negatively offset these launch times to compensate for the delay associated with the IR sensors (2ms) and the delay associated with the servo motor(0.04s), and trigger the launch of the bat bot based on the user input.

1.2 Background:

1) Bat bot: This has already been designed by our TA Jonathon Hoff and his research group. The motivation for the bat bot itself is that it is a superior alternative to quadcopters that have shorter sustainable aerial flight times, are noisier and use really high speed rotors, and additionally the bat bot mimics the flight motion and technique of bats which are arguably one of the most agile aerial species.

2) Automated Launcher: We are designing an automated launcher component for the bat bot because depending on the wing's orientation at the time of launch, it may take one of many different possible trajectories. By controlling and coordinating the launch of the bat at a specific wing orientation, the initial condition can be controlled. Additionally, another functionality that is desired is a controllable time delay that the user can choose to dictate when the launch takes place.

1.3 High-level requirements:

1) For a given frequency/flapping rate (8.5 Hz) the system must be able to accurately detect the position of the wings of the robot and trigger a response that launches the robot when signaled to by a controller, and the accuracy of the detection should be such that in a series of 10 launches there should be no more than an error margin of 15% of the flapping rate in the difference of the launch time between any two of the launches.

2) System must accurately trigger the launch of the robot after a user-specified period of time utilizing the switches on the controller.

3) The system must be seamlessly integrated with the launcher in order to avoid collisions and interference with the launch path of the robot.

2. Design

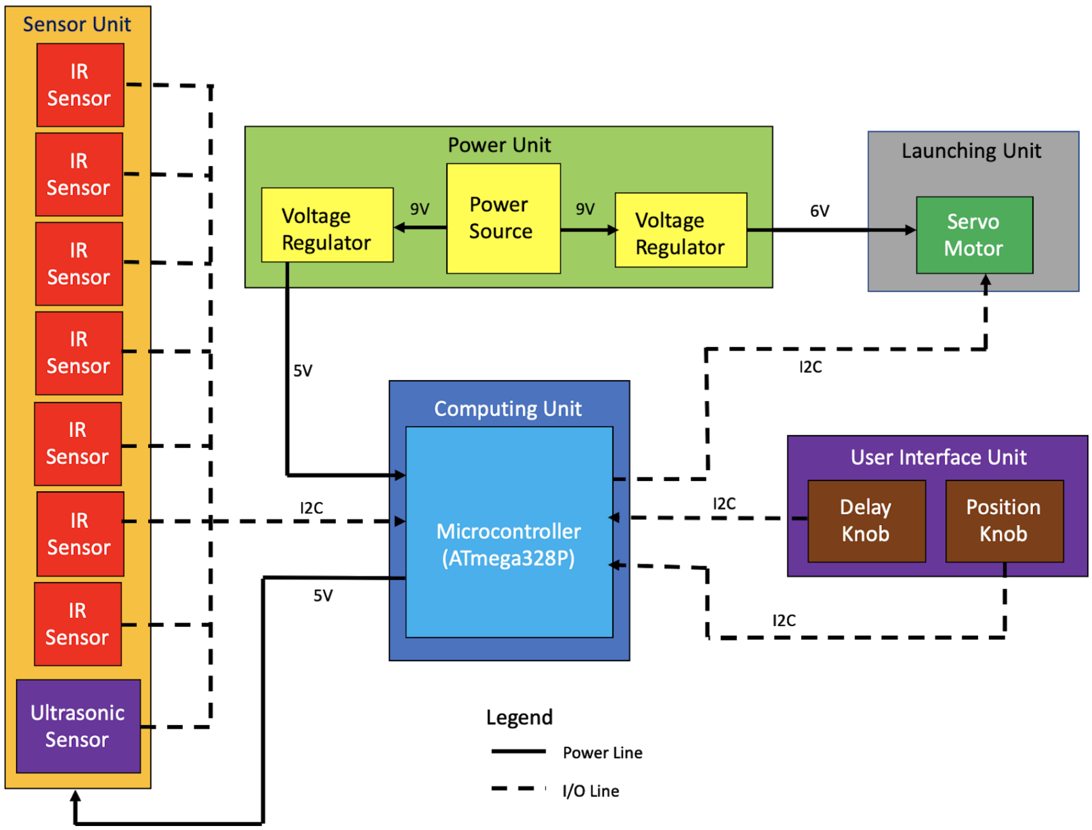

Figure 1. Block Diagram

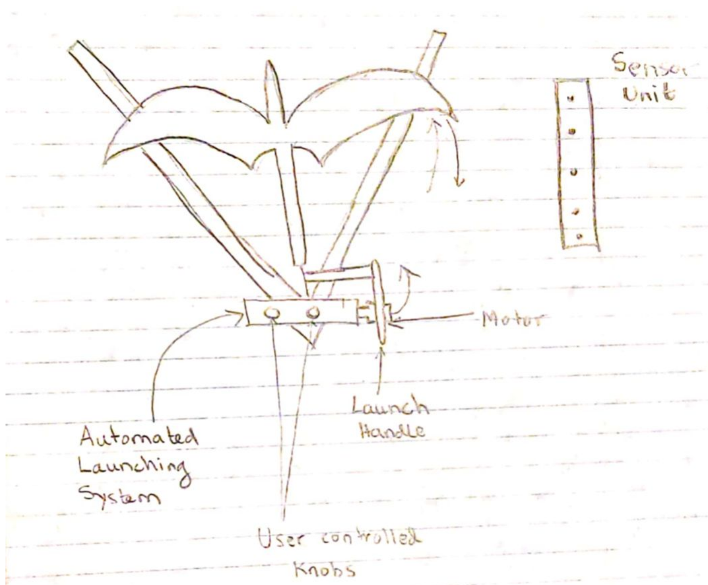

Figure 2. Physical Design

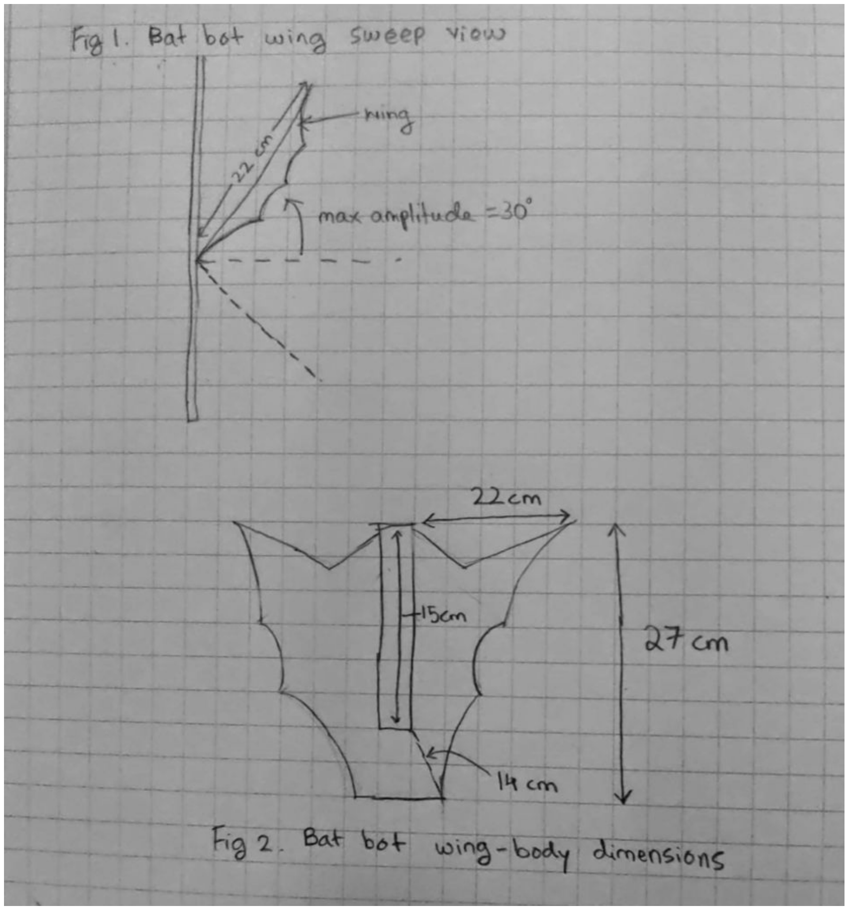

Fig 1. Bat bot wing sweep view

Fig 2. Bat bot wing-body dimensions

2.1 Power Unit:

The function of the power unit is to provide stable DC power to all of the electrical components in our system at their specified operating voltage. The voltage regulators will be responsible for bucking the voltage and maintaining it at stable output voltages of 5V and 6V. Each sensor will draw about 20mA, the micro-servo motor will draw about 150mA, the microcontroller also requires a drive current of 300 mA. and the lithium battery that should be able to provide up to a continuous current of 1000mA.

2.1.1 Power Supply:

The power supply that we are planning to use to power our system is supplied by 9 volt lithium batteries. This will then be regulated by two voltage regulators to provide the appropriate voltages of 6V and 5V to the micro-servo motor and the sensor components respectively.

2.1.2 Voltage Regulator:

The voltage regulators in our system would step-down input 9V voltage (supplied by batteries):
1) to a steady 5V which is what is needed to power our microcontroller , and
2) to supply servo motor with 6V.
We have two voltage regulators: one for the microcontroller and the other for the servo motor since these two components have different voltage requirements as mentioned above. We plan on using off-the-shelf linear voltage regulators in our system.

2.2 Computing Unit:

The high level function of the computing unit is to gather sensor output as well as user provided launch-specific inputs (delay and position) to trigger the release of the bat robot by activating the servo motor of the launching unit. The computing unit will be programmed with an algorithm that models wing position (state of the bat robot) by processing the sensor output. The computing unit communicates with the sensor unit, user interface unit as well as the launching unit using I2C lines.

2.2.1 Microcontroller:

We plan on using an ATmega328P microcontroller chip for our project. The purpose of the microcontroller will be to interface with the sensor unit as well as the user interface unit to eventually trigger the launching unit by activating the motor. The communication between the microcontroller and the sensor unit, trigger controls unit as well as the motor will be through General Purpose IO (GPIOs) pins on the microcontroller.

2.3 Sensor Unit:

The high-level function of the sensor unit will be to accurately model the periodic wing motion of the wings of the bat robot. We plan on using an array of vertically placed IR sensors in order to model the position of the bat wings. The IR sensors will relay the analog output of the intensity of the reflected signal to the microcontroller for processing.
The data collected from the 5 IR sensors (each placed such that it corresponds to a particular wing position) will be used to extrapolate the launch time for the other 5 positions. We will verify and validate mathematically the relationship between our projected theoretical values for time taken to move from position to position and the observed values.
We will use an ultrasonic sensor to validate the data generated by the IR sensors.
Each of these sensors draw about 20mA.

2.4 User Interface Unit:

The user interface unit allows the user to specify the timing delay associated with the launch and the wing orientation from a list of seven preset values each. The user interface unit would consist of two control knobs each capable of enumerating seven different analog values corresponding to the seven different possibilities for each parameter.

2.5 Launching Unit:

The launching unit of our system interacts with the launching handle of the bat launcher (developed by Jonathan Hoff) to release the bat robot when triggered by the computing unit. The torque specifications are mentioned in the sub-system requirements.

2.5.1 Motor:

The function of the servo motor in our system would be to produce suitable torque to release the launching handle of the bat launcher when triggered to do so. It is suitably placed so as to not be an impediment to the launching path of the motor.

Table 1. Subsystem Requirements

Subsystem	Requirements	Verification
Power Unit	1) Must be able to provide regulated, constant DC voltage based on the requirements of the computing unit (5V), and the servo motor (6V) with a tolerance of +- 5% in supply voltage. 2) The supply current should not be more than 1000mA (peak current) at any time (maximum discharge). 3) Voltage regulator must be able to regulate voltage at 5V with 2% regulation.	1) Use a oscilloscope to verify DC voltage of Li-ion battery. We would also verify that the ripples in the voltage waveform conform to the supply tolerance using voltage probes. 2) We would use oscilloscope current probes to measure the supply current across the two terminals of the battery to check that it does not cross the threshold of 1 A. 3) We would connect a multimeter and oscilloscope across the linear regulator to ensure that there is a stable output voltage and we would also check whether the output voltage is within the 2% tolerance range. We could also use 10 uF electrolytic capacitors at both the input and output pins of the regulator to ensure better stability.
Computing Unit	1) The computing unit must correctly estimate wing position from sensor outputs to within 7.5% accuracy (half of 15% from the high level, as	1) 7.5% accuracy in terms of vertical distance corresponds to 1.65 cm. It takes the wing 4.4ms to move through this distance. The way we will test for

	7.5% error in both directions gives 15% error between two flights). 2) The total time of the microcontroller computation should be no more than 1/1000 of the time period of the completion of one flap.	accuracy is we will place one IR sensor at the height that corresponds to our launch position, and two others: one 1.65 cm above and one 1.65 cm below. Based on the timestamp associated with the launch signal, we will check the outputs of the three IR sensors. Based on their outputs, we would know if the wing had exceeded 1.65 cm at the time of launch. Note: So for a given position, the middle IR sensor will output 1. If one of the other two sensor's outputs changes to a 1 from a 0 that means the wing was managed to deviate by 1.65cm and hence did not pass the test. If the other two sensors change from a 1 to 0 or hold their value, that means that the wing orientation was within tolerance limits. 2) Since the microprocessor we have chosen operates at 16 MHz and our sensors and other components operate in the kHz and Hz range, one computation cycle must take less time than the time needed to complete one entire flap of the wing.
Sensor Unit	1) Possess the ability to produce a digital high/low output corresponding to the wing motion at a detectable range of 1m with a drive current of 20 mA. 2) It should be possible to come up with a one-to-one correspondence between a point in the sensor output to a particular wing motion. 3) The sampling frequency of the sensors (kHz) must be of an order of 3 higher than the frequency of the wing motion. (Hz) to allow for accurate calibration and error minimization. 4) The sensing unit should be placed in a position that does not hinder the bat robot launch path.	1) In order to verify the functionality of the IR sensors, we would connect pin 1 to a LED and a pull-up resistor, pin 2 to ground and pin 3 to a power source that provides 5V to the sensor. We would then place an object place an object at 0.5m from the sensor and check whether or not the LED lights up. We would then remove the object and place it at a distance of 1.5m from the sensor and then observe whether or not the LED lights up. If the LED does light up we know that the sensor is detecting an object at 1.5m, which means that it would also detect an object at 1m as well. 2)We plan to place seven sensors corresponding to each position. We will then check when the output signals are high and when they are low. We will compare this with the theoretical time taken for the wing to move through that

		distance relative to the top position(base position) $t = y * ((1/8.5)/2) / 22 = (y * 0.00267s)$ 3) 4) The launch of the robot takes place seamlessly and does not come into contact with the sensor unit.
User Interface Unit	1) The user interface unit must allow the user to calibrate launch specifications like wing position and launch delay time as desired from a choice of 7 preset delay and wing position parameters. 2) The preset wing position parameters must cover different possibilities of launch trajectories (upwards and downwards).	1) We would start a stopwatch immediately after the user sets the desired wing position and launch delay time, and observe the amount of time taken for the bat to launch. We would then verify whether the user defined launch delay time matched the time we recorded with the stopwatch accounting for tolerances in wing positions and sensor delays. 2) In order to verify whether the bat follows the trajectories listed, we would just use an eye test to observe the direction in which the robot travels once it has been launched. We would then check whether the trajectory matched the way in which the robot theoretically must have travelled based on the user defined wing position. We can also corroborate this with our data from the computing unit verification.
Launching Unit	1) Must be capable of producing sufficient torque to launch the bat from the rest position by triggering the release of tension in the strings. 2)The servo motor must provide torque of the order of 0.01 N*m to flick the switch in order to release the bat bot. 3)The servo motor itself must not in any way obstruct the path of the bat bot during its launch.	1) We would pull the bat robot back to its initial resting position using the strings on the launcher. After the motor gets the signal from the microcontroller we must see to it that the motor rotates by 30 degrees to release the tension in the strings. We can check whether or not the motor received a high signal by connecting it in series with a LED, if the LED lights up it means that the motor received a high and should have triggered the launch of the robot. 2) The launch of the robot takes place seamlessly and does not come into contact with the servo motor.

2.6 Risk Analysis:

In the case of our launcher, the greatest risk to the success of our project is the failure of our sensor unit to produce data that can be used in a meaningful way (i.e noise). We have made sure that the sensor's sampling frequency is orders of magnitude higher than the frequency of motion of the wing, but despite that parameters such as the detection angle of each IR sensor is a limitation that we have to work around. We have used multiple IR sensors at equally spaced points such that they cover points spanning from each extremity in the wing's motion (Highest point to lowest point). Using mathematical modelling , we will make theoretical predictions of the time taken for the wing to move from position to position and calculate an error margin based on when each sensor outputs a high signal. After incorporating the error margin and linearly scaling it between positions, we will be able to make more accurate predictions of the time taken to move from a given position to another. This technique will incorporate the effects of air resistance as well. An assumption made here is that between 2 launch positions, the error margin (deviation from theoretical predictions) is varying linearly. If while calibrating our sensors, we find that the non-linearity in error introduced due to air-drag is creating an error in our predictions, we may have to refine our mathematical model.
Another factor that could affect the performance of our IR sensors is that during outdoor testing light, dust, moisture and so on might interfere with the performance of IR sensors. As a work-around, we are planning to focus our IR sensor beams using a tunnel-like contraption around the sensors.

2.7 Tolerance:

Propagation error: There is a certain response time taken by each sensor. The infrared sensors that we plan to use (IR Break Beam Sensor - 3mm LEDs) have a response time of 2ms. In the 2ms, the wings would have moved a certain amount of distance and we calculate that error using the following method.
The frequency of the wing flap is 8.5hz, which corresponds to a time period T = 1/(8.5) = 0.11764 seconds. In T/2 seconds, the wing moves from one vertical extreme position to the other. As the image shows, the max angle during a sweep is 30 degrees. The length of the wing is about 22 cm, and the vertical length through which the wing moves in T time is theoretically h = 2*22*sin(30) = 22cm. In 2ms, the wing would have moved down by y = 0.002*h/(T/2) = 0.748 cm. This is a significant amount. To correct for this, we intend to allow the wing to complete one complete flap in time T, and we will perform the computations of what launch timestamp is associated with each wing position (relative to a wing reference position) and then we will decrease the launch time stamp by the delay associated by the sensors to minimize the error. The servo motor that we intend to use takes 0.04 seconds to move through 30 degrees, which is the amount by which the motor needs to turn to initiate launch. This 0.04 second delay will also be corrected for in a similar manner as the error from the IR sensors above.

3. Ethics and Safety :

There are a few things that we need to take care of during our project. The main safety concern with launching the bat bot is the safety of the environment, the people surrounding the outdoor testing areas/facilities, and ourselves. We must take the necessary precautions to preserve the safety of the surroundings and public in accordance with the first and ninth IEEE code of conduct and ethics [1]. We have to ensure that our testing area is isolated from the public.

The Infrared Sensors that we will be using in our project produce low-levels of infrared radiation that has negative impacts on the human eye in the case of extensive exposure. Manufacturers of IR sensors are required to adhere to the IEC-62471 Standards which relate to Eye Safety [3]. When we purchase IR sensors for our project we will make sure that they are produced by credible manufacturers. While testing our project, we will also make sure that these IR sensors are not directly firing on human eyes.

Since we will be using lithium batteries in order to power our project, a hazard that is often associated with them is called “thermal runaway” which results in overheating and battery failure [4]. Thermal runaway often causes the battery to ignite. In the case of such an emergency we will deal with the fire using appropriate training that was introduced in the mandatory safety training online module.

While working on our project we will also make sure to not expose the system to water and extensive dust which can cause damage to our circuit or other associated electrical components which can often result in safety hazards.

Since we will be presenting results, data, progress and other important findings and observations from our project throughout the course of the semester, we will do so in an honest and reliable manner by keeping in mind point 3 of the IEEE code of ethics [1] and points 1.3 and 2.2 of the ACM code of ethics [2].

We vow to value inclusivity in our work, celebrate diversity and reject discrimination in all forms by upholding point 8 of the IEEE code of ethics [1] and point 1.4 of the ACM code of ethics [2].

Over the course of the semester we will also be going through some of the previous work that was done by others and we will be using findings from previous work to guide certain aspects of our project. We vow to cite and credit other people’s work in accordance with point 7 of the IEEE code of ethics [1] and point 1.5 of the ACM code of ethics [2].

Furthermore, as mentioned in the online Safety Training modules, whenever we work in the ECE 445 laboratory, each of us will ensure that we are not working alone in the laboratory and that there is at least someone else present.

References :

[1] “IEEE Code of Ethics,” *IEEE*. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 13-Feb-2020].

[2] “The Code affirms an obligation of computing professionals to use their skills for the benefit of society.,” *Code of Ethics*. [Online]. Available: https://www.acm.org/code-of-ethics. [Accessed: 13-Feb-2020].

[3] “Eye Safety for Proximity Sensing Using Infrared Light-emitting Diodes,” *Renesas*, 28-Apr-2016. [Online]. Available: https://www.renesas.com/us/en/doc/application-note/an1737.pdf. [Accessed: 13-Feb-2020].

[4] “The Hazards of Lithium Batteries,” *Vice Provost Tufts*. [Online]. Available: https://viceprovost.tufts.edu//ehs/files/The-Hazards-of-Lithium-Batteries.pdf. [Accessed: 13-Feb-2020].

[5] J. Barani, “Difference between sensor response time and sensor time constant τ (tau) 63.2%.,” *BARANI DESIGN Technologies*, 07-May-2019. [Online]. Available: https://www.baranidesign.com/faq-articles/2019/5/6/difference-between-sensor-response-time-and-sensor-time-constant-tau. [Accessed: 25-Feb-2020].

[6] P. by D. P. E. Team, “Servo Motor Controlled Wireless Light Switch,” *Device Plus*, 29-Sep-2017. [Online]. Available: https://www.deviceplus.com/how-tos/arduino-guide/servo-motor-controlled-wireless-light-switch/. [Accessed: 25-Feb-2020].

[7] Adafruit Industries, “Micro Servo - High Powered, High Torque Metal Gear,” *adafruit industries blog RSS*. [Online]. Available: https://www.adafruit.com/product/2307?gclid=Cj0KCQiApt_xBRDxARIsAAMUMu8DJ-uuFs-6tOg6P9fujQ1yQXtjG4Daq4FBgwxpiNMOfPCaNIVJCmcaAkHdEALw_wcB. [Accessed: 25-Feb-2020].

[8] Adafruit Industries, “IR Break Beam Sensor - 3mm LEDs,” *adafruit industries blog RSS*. [Online]. Available: https://www.adafruit.com/product/2167?gclid=Cj0KCQiA4sjyBRC5ARIsAEHsELEQvSkj3I_fo28Zwd1InnSDBPLw06ZITB79wKVqhUBJxGf0ArZXZI0aAmTIEALw_wcB. [Accessed: 25-Feb-2020].