.

Early Bird Alarm Clock
Design Document

Shouri Addepally
Sasin Gudipati
Siddharth Sharma

ECE 445

Fall 2022

Table of Contents

Introduction **3**
Problem: 3
Solution: 3
High Level Requirements: 3
Block Diagram: 4
Physical Design: 5
Subsystem Overview: 5
Subsystem Requirements: 8
Tolerance Analysis: 8
Cost and Schedule: **11**
Cost Analysis: 11
Schedule: 12
Ethics and Safety: **13**
References: **14**

Introduction

Problem:

As college students and as individuals soon entering the workforce, it is critical that we are able to wake up and be on time for lectures or meetings. To be on time, we tend to usually set alarms on our phones or on stationary alarm clocks. However, once it's time to wake up, our body defaults to hitting the snooze button for a few more minutes of rest which causes us to run late. While some may believe that oversleeping is not a major issue, two-thirds [of polled respondents] say snoozing their alarm at least once is part of their morning routine and that 72% snooze between one to four separate times on any given morning (Melore).

Solution:

Our group's project offers a potential solution to the problem of oversleeping by creating an "interactive" alarm clock which forces an individual to wake up and begin their day. At a high level overview, our mobile EarlyBird alarm clock will sound an alarm and will begin "running around" autonomously while also navigating around any objects which may lie in its random path. Upon the alarm being switched off, the user will be required to type in an answer to a hexadecimal addition problem displayed on the alarm clock. If a correct answer is inputted, the verification is complete and the alarm is turned off. If an incorrect answer is inputted, then a different problem will be shown on the alarm clock and the user will need to re-enter a valid answer to shut the alarm off.

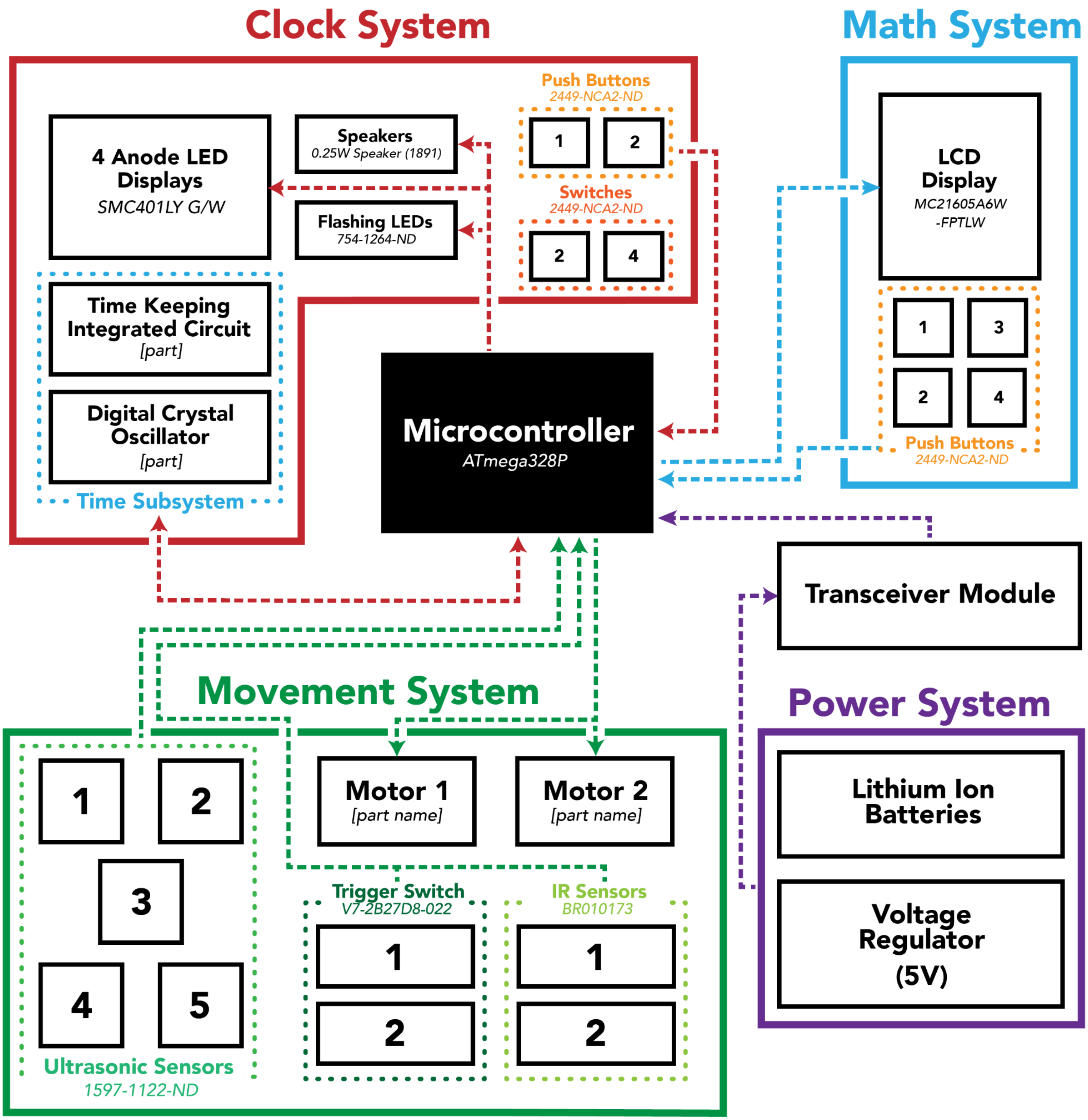
Our EarlyBird alarm clock will contain the following three components: alarm clock, driving system - motors + ultrasonic sensors, and a programmable hexadecimal addition problem needed to switch the alarm off for good. The alarm clock will show the time using and will also include function buttons which allow the user to set the correct time in hours and minutes. The alarm clock component will also include an AM/PM indicator as well as an alarm indicator. The driving system component consists of four motors which will be attached to four wheels respectively and servo motors to which the ultrasonic sensors will be attached. The driving system will also include autonomous obstacle detection and avoidance which is a key component as part of the alarm clock "running away". To give our clock a complete picture of its surroundings during autonomous control, we plan to use multiple ultrasonic sensors which will be attached to the front of the alarm clock and will constantly perform scans to give the clock a more accurate mapping of its surroundings. The final component is the hexadecimal addition problem which ensures that the alarm remains off if a correct answer is provided. The hex-add display will be on the opposite side of the time display and features an addition between two-bit hex values. If the answer inputted matches the correct answer, then the alarm clock will stay turned off. Otherwise, it will continue to ring as an invalid answer has been given.

High Level Requirements:

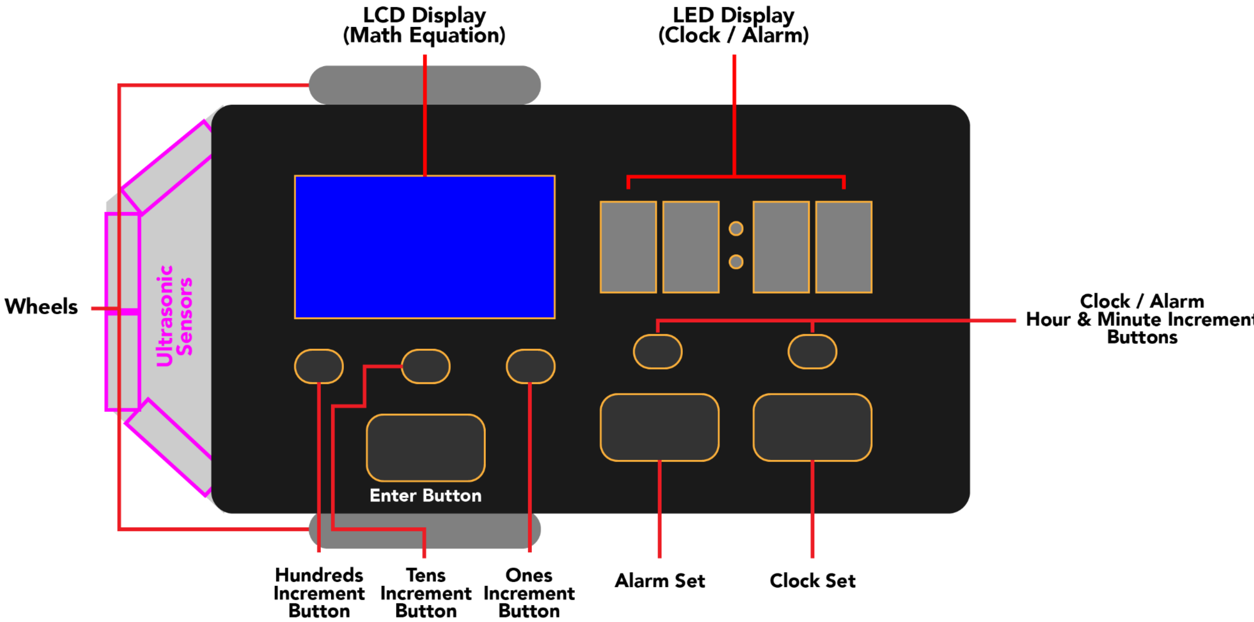
1. The device has the functionality of a normal alarm clock. It will have the ability to set the time, set an alarm, and produce an alarm sound.
2. The alarm clock can navigate the room and perform basic obstacle avoidance.
3. The device will display a math equation that must be solved before the alarm turns off.

Design

Block Diagram:



Physical Design:



Subsystem Overview:

1. **Alarm Clock Subsystem**

   The alarm clock subsystem is used to display the time and also is used to set/turn off alarms that have been set. In order to keep track of time properly, we have introduced a **time subsystem** which consists of a time-keeping IC and a digital crystal oscillator. The clock subsystem also contains 2 push buttons for setting the time and the alarms and these buttons establish a connection to the time subsystem. The clock subsystem also includes a switch which toggles between the alarm set or time/clock set feature.

Requirement	Verification
• The microcontroller should be able to iterate through the set of values (0-9) and display the values on the 7-Segment LED Display.	• We can test this by observing the LED displays and checking if the values are iterated through properly.
• The microcontroller should be able to utilize the time subsystem and keep track of seconds using the flashing LEDs with a minimal lag in +-0.05 ms.	• A test can be done by syncing the frequency of the flashing LEDs with another clock or set frequency and checking to see if the difference falls within our minimal lag bounds.
• The microcontroller should be able to take in push button inputs for time-setting capabilities, and switch input for determining time-set or alarm-set choice.	• Create/maintain an FSM which transitions from 0 to 1 on a switch toggle for time-set or alarm-set. The buttons should cycle through values 0-9 on the LED displays, where a 1 indicates a transition to the next numerical value. Encode this simple logic of these FSMs as part of our microcontroller capability to handle inputs and update the LED displays accordingly. Test by seeing transitions on

	LED displays and checking if the alarm's speakers go off at the inputted alarm time as well.
• The microcontroller can control the speakers to play an alarm at a specific tone and for a specific time of 15 seconds +- 1 second before shutting off for a 15 second +-1 second and then repeating.	• We can observe the frequency of the alarm clock or keep track of how long it is ringing for. We can pre-set values and check if the values we have kept track of match up with what we expect to see.

2. **Autonomous Movement Module**

The autonomous movement module is responsible for making the alarm clock move around without running into external objects. In order to generate a comprehensive mapping of its environment, we will be mounting multiple ultrasonic sensors which will all be interfaced with the ATMEGA328P microcontroller. This module also contains the two motors which will each power a wheel.

Requirement	Verification
• There are five ultrasonic sensors at the front of the alarm clock. The microcontroller must be able to identify which sensors have objects in front of them	• As a unit test, we will connect the sensor array to the microcontroller, and connect the microcontroller to 5 LEDs, each of which represents one sensor in the array. We will then place objects within the sensors' detection range, and the microcontroller indicates which sensors detected objects by illuminating the corresponding LED.
• The microcontroller can engage the motors to turn the chassis 45 degrees right, 45 degrees left, and 90 degrees right/left. Each turn should take place in less than 1 second +/- 0.25 seconds	• The unit test will involve writing a program that drives the motors to perform turns in the following order: ○ Right, 45 degrees ○ Left, 45 degrees ○ Right, 90 degrees ○ Left, 90 degrees • For this test, the motors need and wheels need to be mounted on the chassis. We will start the program and verify that the vehicle is turning in the aforementioned order in under 1 second for each individual turn.
• The microcontroller can engage the motors to move the chassis in reverse for 15 +/- 0.5 cm in under 0.5 +/- 0.25 seconds	• Once again for this test, the motors and the wheels need to be mounted on the chassis. Next, using the wheel circumference and the motors rotations per second, we will calculate the amount of time we need to run the motors in reverse to move the chassis 15 cm backwards. Once the program is loaded onto the microcontroller, we will initiate the program and simultaneously start a stopwatch. At 0.75 seconds, we will stop the stopwatch and measure the chassis displacement.

• The microcontroller can engage the motors to move the chassis forward 1 +/- 0.1 meter in less than 3 +/- 0.25 seconds	• This test is identical to the previous test, except the chassis is moving forward for 1 meter, and the stopwatch will be stopped after 3.25 seconds.
• The microcontroller reverses the chassis 15 cm in 0.5 +/- 0.25 when the anterior touch sensors are triggered	• This unit test consists of a program that continuously moves the car forward. If the touch sensors detect an object, the microcontroller raises an interrupt which executes a subroutine to reverse the car. • The car will be facing a wall 1 meter away, and the program will be initiated. Upon touching the wall, the car should reverse between 14.75 and 15 cm in less than 0.75 seconds.
• The microcontroller can use the ultrasonic sensor input to drive the motors so that the chassis never makes contact with any obstacles.	• The unit test for this requirement consists of a program that drives the motors and monitors the output of the ultrasonic sensors. The car's default state is to move forward. Depending on the particular combination of ultrasonic sensors that detect an object, the program will choose either to turn the chassis 90/45 degrees in a certain direction, move forward, or reverse. • The obstacle avoidance program will be tested using the three test cases: 1. freestanding objects (such as a toy lying in the middle of the room) 2. Walls 3. Corners

3. Math Module Subsystem

The math module system is responsible for displaying a hexadecimal addition problem and verifying if the inputted answer is correct. If the answer is inputted incorrectly, then the alarm should not be turned off and should continue to ring until a valid answer is provided by the user. This subsystem is connected to the alarm clock subsystem - the result of the addition determines if the alarm should be switched off or not. The math module subsystem also contains push buttons which allow the user to input a 2 digit hex answer (one push button for tens digit, one for ones digit) and an LED display which allows the user to see what they've just inputted.

Requirement	Verification
• The microcontroller can generate two random two-digit hexadecimal operands and compute their sum	• Create a subroutine that randomly generates two operands and calculates their sum. The microcontroller will output these operands to the serial port. We can view the two operands and the sum by connecting the microcontroller to a computer. We will run the subroutine 50 times to ensure that operands are random and the sums correct.

• The user will use three buttons to input an answer and an 'enter' button to submit. The hundreds place, tens place, and ones place each have a dedicated input button. When an input button is pressed, the value in the corresponding place should increase by 1 until the value reaches 9, at which point the value should reset to 0.	• A unit test for this requirement consists of outputting the value of the answer punched in by the user to the serial port, and connecting the serial port to a computer where we should see the answer value changing in response to button presses.
• The microcontroller can identify when the user has inputted the answer that matches the calculated sum for the math equation	• This requirement will be tested by connecting the microcontroller to the computer using the serial port. We will run the subroutine that generates the math equation, and then use the push buttons to submit answers. The microcontroller will send a 1 to the serial port when an answer that matches the calculated sum is entered.

4. **Power Subsystem**

The power subsystem is responsible for supplying power for the rest of the aforementioned subsystems and this is how it is connected to the rest of the subsystems as well. It consists of lithium ion batteries which were chosen due to their rechargeability advantage, and a voltage regulator. The power subsystem is also connected to the transceiver module which is used to communicate how much power/voltage is supplied to the PIC microcontroller.

Requirement	Verification
• The power subsystem needs to provide 5V +/- 0.3V to the LCD, the ATMEGA328P, the HCSR04 ultrasonics, the motors, and the push buttons	• Use an oscilloscope to measure the output voltage of the voltage regulator and verify that it is between 4.7 to 5.3V.
• The power subsystem needs to provide 2V-2.6V to the speakers and the seven-segment LED display.	• Use an oscilloscope to measure the output voltage of the voltage regulator and verify that it is between 2 to 2.6V.

Subsystem Requirements:

1. *Alarm Clock*: wThe alarm clock consists of 3 subsystems: a "real-time clock", push buttons for setting the time and alarms, as well as a LED HEX display to show the time. The alarm clock runs on a 5V power supply. The microcontroller interfaces with all three subsystems of the alarm clock. The microcontroller will ***read*** input from the push buttons. The microcontroller will ***read/write*** information from the RTC. Based on input from the push buttons the microcontroller will ***write*** the time in the RTC subsystem and on the clock cycle the microcontroller will ***read*** the current time. The current time will be ***written*** onto the LED display by the microcontroller. Outside of basic function, an added feature will include a dimmed preview of the time using a photoresistor to control ambient lighting (Photoresistors).

2. *Autonomous Movement Module:* This subsystem satisfies the requirement to have autonomous movement. This will consist of multiple ultrasonic sensors that are placed evenly because they each roughly get less than 30 degrees of coverage (Peppers+Fuchs). The reason we are using multiple ultrasonic sensors is to make sure that no blindspot is being missed. These sensors will connect to the Microcontroller and send the data to it. From there, the data that is being sent to the Microcontroller is now distributed to the 4 motors and axles that turn the car if necessary.
3. *Math System:* The math system consists of an LCD Display controlled by the microcontroller. These displays show the Hex addition problem and in order to register input, there will be two push buttons - one that increments the "tens" digit (_x), and one that increments the "ones" digit (x_). It will loop from 0 to F on both and then start back over at 0 so there is no need to decrement buttons. To visually show this input, we are doing input LEDs as well so that the user can see the number while clicking the buttons.

Tolerance Analysis:

Our design makes use of 5 ultrasonic sensors which will be mounted on a semi-circle plate which protrudes from our alarm clock. These sensors are aimed to provide an accurate depiction of objects within a reasonable vicinity from the alarm clock, and in turn help for better object detection while the clock is running around. When using multiple ultrasonic sensors in the same system, there is an interference that happens when they are firing in similar ranges so it affects how it reads what is nearby. This interference is called cross-talk or noise (Inc, M. B.). Thus, we must place the ultrasonic sensors in spots which will minimize the cross-talk and also encompass the largest possible distance directly in front of the alarm clock. A plausible approach is to fire off sensors at different times so that they are not interfering with each other and to set up a timer so the microcontroller chooses sensors in an order extremely quick so it can tell what is in front of it. First, we can compute the time it takes for an ultrasonic sound to travel; this will help us also get a better understanding of the delay encountered in our staggered sensor approach.

$$\text{Dist}_{\text{Front of Clock}} = (\text{Speed of Sound} * \text{Time}) / 2$$

Given that we would like a reasonable detection distance of 1.5 cm for any outside obstacles for an extremely close object detection, we can use the above formula to compute the detection time as follows:

$$\text{Detection Time} = (2 * \text{Dist}_{\text{Front of Clock}}) / \text{Speed of Sound} = (2 * 1.5\text{cm})/(340 \text{ m/s}) = 88.24\mu\text{s}.$$

With the staggered firing of 5 ultrasonic sensors, there is a lag before each subsequent sensor after the first one is fired. If the sensors are fired in the same order and in cycles, the following initial lag times can be categorized into the following table:

	Sensor 1	Sensor 2	Sensor 3	Sensor 4	Sensor 5
Min Lag Time	0 µs	88.24 µs	176.47 µs	264.71 µs	352.94 µs
Max Lag Time	352.94 µs	352.94 µs	352.94 µs	352.94 µs	352.94 µs

Since Sensor 5 has the highest mean lag time, we can use this lag time as the detection time value in our first equation and derive the potential distance measured in this lag transmission time:

$$\text{Dist}_{\text{Front of Clock}} = (352.94\ \mu\text{s}) * (340 \text{ m/s}) / 2 = 0.0599 \text{ m} = \underline{5.99 \text{ cm}}$$

We can perform a simple calculation for the front-facing platform dimension for our alarm clock vehicle as follows, given the dimension of the Ultrasonic HC-SR04 Distance Sensor being 2.0 cm in width and the usage of 5 sensors.

$$\text{Width}_{\text{Sensor Platform}} = (\text{Width}_{\text{UltraSonic Sensor}}) * (\text{Number of Sensors}) = 2.0\text{ cm} * 5\text{ Sensors} = \underline{10\text{ cm}}$$

Therefore, the max area that can be scanned under the most optimal sensor approach given is computed as follows:

$$\text{Max Scannable Area} = \text{Dist}_{\text{Front of Clock}} * \text{Width}_{\text{Sensor Platform}} = 5.99\text{ cm} * 10\text{ cm} = \mathbf{59.99\ cm^2}$$

Now, we can move the efficiency of our ultrasonic sensor approach by calculating its scannable area and comparing it to the Max Scannable Area computed above.

For an individual ultrasonic HC-SR04 sensor, the measuring angle is 15 degrees, and this angle can be used to compute the hypotenuse of the triangle along with the reasonable detection distance of 5 cm given the capabilities of ultrasonic sensors. This hypotenuse is the max distance that is covered by the measuring angle of an ultrasonic sensor and is computed as follows:

$$\text{Dist}_{\text{Covered by Sensor}} = (\text{Dist}_{\text{Front of Clock}}) / \cos(\text{measuring angle}) = 5\text{ cm} / \cos(15) = \underline{5.17\text{ cm}}$$

We can also solve for the distance covered by the ultrasonic sensor's measuring angle which involves solving for the side opposite the measuring angle, and multiplying this value by 2 to account for the distance spanned across the entire rotation.

$$\text{Dist}_{\text{Measuring Angle}} = (2 * (5\text{ cm} * \tan(15))) = \underline{2.6298\text{ cm}}$$

Finally, we can compute the entire scanned area across the 4 sensors fired off during the max lag transmission as follows:

$$\text{Scannable Area}_{\text{Total}} = (\text{Scanned Area}_{\text{Sensor}}) * (\text{Number of sensors})$$

Thus, the Scanned $\text{Area}_{\text{Sensor}}$ and the Scanned $\text{Area}_{\text{Total}}$ are computed as follows:

$$\text{Scanned Area}_{\text{Sensor}} = \text{Dist}_{\text{Covered by Sensor}} * \text{Dist}_{\text{Measuring Angle}} = 5.17\text{ cm} * 2.629\text{ cm} = 13.596\text{ cm} * 4\text{ sensors} = \mathbf{54.384\ cm^2}$$

Now, the efficiency ratio for the ultrasonic sensor layout that is proposed can be computed using the Scanned Area computed across all the ultrasonic sensors fired off during the delayed transmission period, and the maximum Scannable Area computed across the same delayed period:

$$\text{Efficiency Ratio} = \text{Scanned Area}_{\text{Across All Sensors}} / \text{Max Scannable Area} = 54.384\text{ cm}^2 / 59.99\text{ cm}^2 = 0.906 = \underline{\mathbf{90.6\%}}$$

With the above efficiency ratio of 90.6%, we can confidently state that the approach of staggering four ultrasonic sensors before a sensor is fired once again is efficient and an optimal sensor setup for accurate object detection as the alarm clock is running away.

Cost and Schedule:

Cost Analysis:

As engineers working on this project, each individual team member would require a compensation of $45 an hour * 20 hours/week * 10 weeks = $9000. The total labor cost would come out to be $27,000 for labor. Our project would also roughly require 6-8 hours per week - 38 * 7 = $266. The total for all the parts is $32.50. Thus the final price would be $27,000 + $266 + $32.50 = $27,298.00

Component	Manufacturer	Part Number	Quantity (USD)	Price
Double AA Battery	Duracell	N/A	4	4.30
4 x AA Battery Holder	Adafruit	830	1	2.95
5V Voltage Regulator	onsemi	LM317MTG	1	.78
2V Voltage regulator	Texas Instruments	TPS7A0220PDQNR	1	.52

Schedule:

Week	*Project Deliverable*	*Individual Tasks*		
		Sasin	***Shouri***	***Siddharth***
9/5	Project Approval	Project Research		
9/12	Project Proposal			
9/19	Machine Shop Conversation	Parts Research		
9/26	Design Document	Visual Aids, PCB Design Ideas, Design Document Work	System Design Ideas - Signal Analysis, Design Document Work	Parts Research, Design Document Research, Physical Model Research
10/3	Design Review & Design Document Changes	Initial PCB Design, Physical Design, Math Module Development	Clock Module Development, Machine Shop Submission	Power System Research, Testing Sensor Signal Analysis (Arduino), *Begin Ordering Parts*
10/10	PCB Design (I), TE (I), Machine Shop FINAL	PCB Design, Assemble PCB, Program Microcontroller	PCB Design, Assemble PCB, Embedded Programming for Sensor Signals	PCB Design, Assemble PCB, Program Microcontroller
10/17	N/A	Assemble Clock	Assemble Clock	Assemble Clock
10/24	N/A	Testing, Log PCB Issues + Possible Expansions (Redesign if necessary)	Testing, Debug Microcontroller Programming	Testing, Chassis Assembly Issues
10/31	PCB Design (II)	Progress Reports, PCB Design, Assemble new PCB (if Necessary), Testing & Verification	Progress Reports, PCB Design, Assemble new PCB (if Necessary), Testing & Verification	Progress Reports, PCB Design, Assemble new PCB (if Necessary), Testing & Verification
11/7	N/A	Demo Preparation	Demo Preparation	Demo Preparation
11/14	Mock Demo			
11/21	*Thanksgiving Break*			
11/28	Final Demo			
12/5	Final Presentation, Paper, Design Notebook and TE (II)			

** Completed Weeks Shaded Gray*

Ethics and Safety:

In terms of ethics and safety, we are sure that there are minimal risks related to this project. However, our main focus is to focus on the health of the user. We are to abide by the IEEE Code of Ethics Section 7.8 to guarantee the wellbeing and safety of our users.

Physical Injury:

Legs:

The alarm clock's movement may cause harm to the user if it hits the user's legs at a high speed. We are monitoring our speed, and maintaining slow enough speeds in which the user should not get injured. We do not want this car going so fast that it hurts any living object or even causes inanimate objects to fall and break.

Ears:

The alarm clock noise should be kept at a respectable volume so it does not harm the user's ears, yet still wakes them up when sleeping.

Privacy and Safety:

Although our alarm clock is placed in the user's possession, there are no concerns about Privacy and Location tracking. This is because we are not using any equipment that keeps track of long-term data. This would include parts that have recording components, i.e. microphones, cameras, 3-D rendering sensors.

References:

Contributor, T. T. (2005, September 21). *What is debouncing? - definition from whatis.com*. WhatIs.com. Retrieved September 15, 2022, from https://www.techtarget.com/whatis/definition/debouncing#:~:text=Bouncing%20is%20the%20tendency%20of,or%20closing%20of%20a%20contact.

"IEEE Code of Ethics." *IEEE*, https://www.ieee.org/about/corporate/governance/p7-8.html.

Inc, M. B. (2021, September 9). *How to use multiple ultrasonic sensors: MaxBotix Inc*. MaxBotix Inc. Retrieved September 15, 2022, from https://www.maxbotix.com/tutorials1/031-using-multiple-ultrasonic-sensors.htm

Melore, C. (2022, April 29). *Snooze News: Average American sets 4 different alarms to wake up!* Snooze Button Social Life. Retrieved September 15, 2022, from https://studyfinds.org/snooze-button-social-life/

Microchip technology. (n.d.). Retrieved September 16, 2022, from https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/DataSheets/39582C.pdf

Pepperl+Fuchs. (2020, March 17). *Transporting printed circuit boards*. Printed Circuit Board Detection with Ultrasonic Sensors. Retrieved September 15, 2022, from https://www.pepperl-fuchs.com/usa/en/21434.htm

Photoresistors. Photoresistors - an overview | ScienceDirect Topics. (n.d.). Retrieved September 15, 2022, from https://www.sciencedirect.com/topics/engineering/photoresistors#:~:text=A%20photocell%20or%20photoresistor%20is,light%20will%20cause%20higher%20resistance.

Switch Debounce in digital circuits. GeeksforGeeks. (2019, November 20). Retrieved September 15, 2022, from https://www.geeksforgeeks.org/switch-debounce-in-digital-circuits/