Programmable Remote Ear Alarm Clock

Ian Mowers

Computer Engineering, Union College

ECE497 – Capstone

ABSTRACT

In this project, we are looking to solve the issue of loud alarm clocks, which can cause noise problems for people who share rooms with others. To eliminate this issue, a simple ear-based system is proposed, which consists of a smartphone that signals a hearing-aid style receiver to vibrate. The vibration from the receiver will be enough to wake the user whilst not waking any others. It is intended to work within the perimeters of a typical bedroom (about 20ft maximum), allow the user to take different sleeping positions, and sleep undisturbed. In order to check the final prototype, the basic functionality of the device will be tested: the software must send a signal out to the receiver, and the receiver must have its vibration section vibrate.

Table of Contents

Index of Figures and Tables

INTRODUCTION

The main context for this project involves noise-related issues that people could have when waking up in the morning to alarm clocks. This in particular can affect others sleeping in the room. To combat this problem, the end goal is to design something more lightweight and portable that allows for a quieter way of waking up. The solution involves using current hearing-aid technology as a base for design, but further modifications to composition will be made to ensure comfort and feasibility of uninterrupted sleep while functioning.

Current alarm clocks for the most part have gone digital, but this technology still relies on the physical clock itself as well as a buzzer that could still be loud enough to wake users. Common and cheap digital alarm clocks furthermore have at most one or two buzzer settings, compared to white noise machines which have many more settings. Another inherent problem with these digital clocks is that after extensive use and remaining plugged in all the time, they begin to slow gain time. The current digital clock I use will, after about 2-3 months have gained about 4 minutes in time. I know it is easy to fix this issue just by resetting the clock, but I would like to have alarm clocks be consistent. The use of smartphones, which are connected by satellite to atomic clocks, will be most precise measurement of time, thus act as our clock.

Some may think that since smartphones already have so much capability in them, why should we make a separate device for the alarm? The phones still have to make some sort of noise to wake people up, and thus still can be a problem. People have also begun to use the phone under pillow and set the alarm to vibrate; seeming like an easy solution. But due to the active nature of phones, communicating with Wi-Fi other things at night (unless the user changes the phone settings, but I bet most people do not), studies are beginning to show that leaving such devices near your head at night could pose health problems. My project aims to eliminate the need for putting the smartphone under the pillow, but use it as part of a system to wake users.

PROBLEM DEFINITION

The basis of project comes from a common issue that some people face in their everyday life: waking up to a loud alarm clock. This may seem like a non-issue for most, but it is not for those that share rooms with other people. In my personal experience, I use an alarm clock every day to wake up in the morning, but I typically wake up before it goes off, for fear that I may wake my roommate. My most recent college roommate, on the contrary, was a heavy sleeper and needed to wake up to a loud alarm. Most days I would get up before him, but since we typically go to bed early, he needed to get up earlier than me sometimes. This resulted in occasions where I was woken up earlier than expected, being one of the many examples where alarms can cause problems for other sleepers.

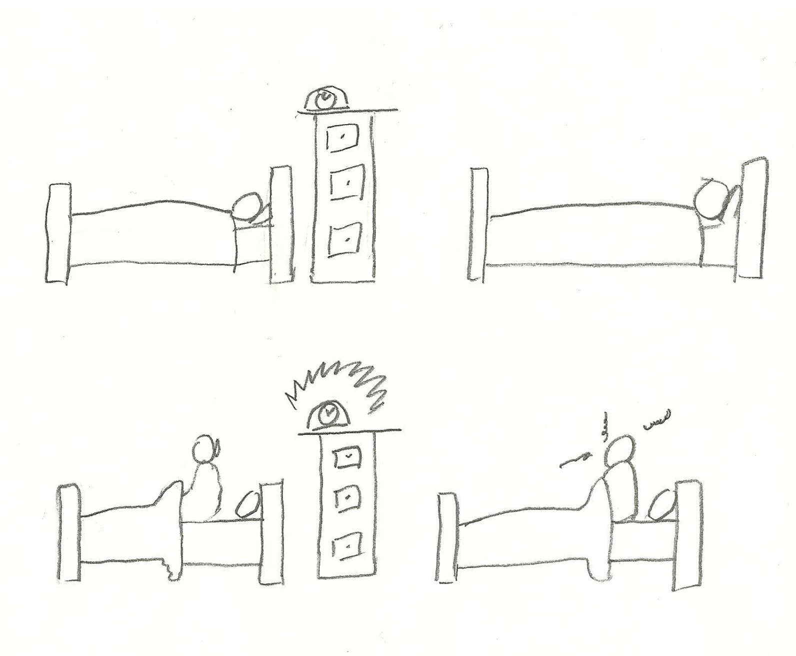

Figure 1: Artist's depiction of alarm clock problem. Note the frustration in the non-alarm clock user.

The desired end users or customers are those who sleep in a room with another person or multiple people; those under this kind of situation typically being college roommates, people who share apartments, or sleep in the same bed. Other potential customers include those who would prefer a quieter, less abrupt way of waking up. The main goal by fixing this issue is to ensure that only those who need to be woken up, will, and could also lead to a smoother wake-up process.

The desired behavior of this device will be spread out into two separate parts, a remote device that can send out a signal and the actual alarm device. The alarm will be a rechargeable in-ear device that stays on the ear while you sleep, while the alarm is set with the remote device. When the alarm goes off, a signal is sent from the remote device to the in-ear device and it goes off. I originally wanted the alarm itself to be a vibration, but other alarms are examined and decided upon in the design requirements below. The alarm will shut off automatically to preserve power and make it easier for user to have the vibration stop. Recharge-ability will depend on the power requirements for the ear device, since it will be active, but the device is intended to be rechargeable for useability and cost-efficiency. In the least, the device will need to be charged once a day, but maybe could last for a week before being recharged.

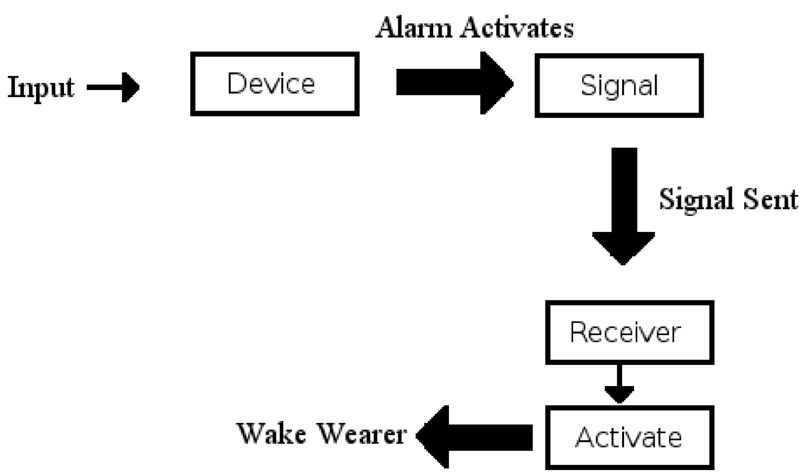

Figure 2: Basic block diagram of solution to problem

The user interface will allow the user to see the time, be able to set it, and maybe have some control over the strength of the vibration (which may affect how long the vibration will last). Overall, the remote signal device takes an input, the alarm time, or an alarm type in addition, then outputs a signal to the ear device. The input of the in-ear device will be the signal from the remote device, while the output is the alarm itself.

Design Requirements

	Behavior	**Constraints To Consider**
Signal Device	Signaling	Signal Standards
	Time Keeping	Required to design a time system, Cost
	Power	Need to design power source, cost
	Settings	Physical design of device, circuit design
Receiver	Receiving	Standards with transmitters, Health concerns
	Vibration	Circuit design, Power
	Power	Sourcing, Cost
	Recharge-ability	Cost, Effectiveness, Longevity
	Power Indication	Circuit design, physical design
	Physical Design	Health concerns, Power, Recharge-ability

To understand what choices in design are needed for the project, I first needed to look at the desired behaviors for the solution and figure out their constraints. The signal device's primary behaviors included how the signal was going to be sent out, keeping track of time, maintaining power in the system, and any additional settings that would be needed. These are constrained by a variety of issues, including standards, cost, design, and such, further described on the table above. The receiver has a mostly different set of behaviors required, including how the receiver will receive the signal, how the receiver will vibrate, supplying power and its ability to recharge, and physical design. Another, more periphery behavior is to include some indication that the power is low, since it will be a remote device. Constraints on the receiver range from cost to design to health and power, as further detailed in the table above.

Behavior	Device Style	Option	Advantages	Disadvantages
Signaling	Smartphone App	RFID	Direct from phone	Range, Standards
		Phone Signal	Use phone signal directly	Unsure about behavior and protocols, more work
	Custom-Made	Physical Receiver	Simple construction	Learn how to build it
Time Keeping	Smartphone App	Phone's Clock	Atomic/Satellite	-
		Alarm Clock App	Use existing app	Differing OSes, Redundant
	Custom-Made	Physical System	-	Learn how to build accurate clocks, power issue already described earlier
Power	Smartphone App	Phone Power	No separate charge	Relies on phone
	Custom-Made	Built-in	Own power supply	Need to construct separate source, time consuming
Settings	Smartphone App	-	Construct app to maintain settings	Needs app to function properly
	Custom-Made	-	Simple button based interface	Very limited and fixed design, choices will affect outer shell
Overall	Smartphone App	-	Smartphones nearly ubiquitous	Need to learn programming for OS, beholden to phone power, range, security
	Custom-Made	-	Separate controllable, portable device	Construction, Maintenance, Precision

For the Signal Device, I explored two main options to implement each desired behavior, while also exploring sub-options within, resulting in the table above. The two main ideas were to either create a smartphone application, using the smartphone as the power and signal transmitter, or design and construct a custom-made receiver that would also keep track of time. Advantages and disadvantages for both options were consider, while comparisons between sub-options on the signaling and time-keeping were explored.

For signaling, there were two options on the smartphone: utilizing the RFID system to communicate, or using the phone signal itself. For the custom-made transmitter, the device would include just the transmitter as a part of the circuit. Considering the advantages and disadvantages, although the RFID system could have range issues and require looking into transmission protocols, it appears to be simplest, cost-effective and easiest to learn to use.

Time-keeping was as major concern for the overall design. For the smartphone, we can just simply use the phone's own clock, which is connected wirelessly to satellites. Using this will potentially allow us the most precise time-keeping, compared to a custom-made clock system that would require knowledge of digital clock-making and creating something that would be prone to error like current alarm clocks.

Power for the transmitter could simply come from the smartphone itself, while a separate system would require the construction of a transformer to draw power from electrical outlets. As for the settings, all the settings could be controlled by the application software on the smartphone, thus making it easy to add more features, while putting it on a separate, molded device would affect the circuit design as well as limit future features.

Overall, smartphones have become nearly ubiquitous at this point in time, but creating a software application would require me to understand how to program in phone operating systems and also would be restricted by the phone's capability. For a custom-made transmitter, we could have a separate device that would then have all of its own capabilities and specifications, but I would be required to design all the specifications and ensure precision of the device itself, which could be a strenuous process.

Behavior	**Device Style**	**Option**	**Constraints**
Receiving	Hearing-Aid	Receiver on edge of shell	Size of receiver, Health hazards

	Mesh Earplug	Receiver on end, maybe as a chip	Health hazards
Vibration	Hearing-Aid	Vibrator goes in ear	Comfort, Health
	Mesh Earplug	Whole piece is vibrator	Design, Health
Power	Hearing-Aid	Button Batteries	Battery types, cost
	Mesh Earplug	Button Batteries	Same as above, design, cost
Recharge-ability	Hearing-Aid	Battery compartment	Inserting and removing batteries, see text for more
	Mesh Earplug	Battery compartment	Same reasoning as above, but much more problematic
Power Indication	Hearing-Aid	LED Light	Design, Manufacturing
	Mesh Earplug	LED Light	Design, Comfort, Manufacturing
Physical Design	Hearing-Aid	Around Ear	Health, Comfort
	Mesh Earplug	In-Ear Mesh	Health, Comfort, Functionality

Two main design ideas were considered for the receiver: hearing-aid style and a mesh earplug style. Aside from the physical design and the vibration, most of the options were the same, with power being constrained to button batteries and indication of low power being relegated to a small LED light. Primary constraints were determined by concerns. Since this is going to be an on ear device, a lot of decision-making was placed on health, design, and comfort.

Cost was mainly affected by the power. I checked on Amazon.com (a large scale supplier site) for slightly cheaper pricing and found that a single set of 60 single-use batteries for hearing-aids were about 15-25USD. These kinds of batteries typically last less than a week. Rechargeables turned out to be much more expensive, where a two-pack of rechargeable batteries were priced the same as a single use 60-pack. The difference here was that the total lifespan of these batteries on average were about a year, the average charge cycle typically being 12-18 hours, depending on size. With rechargeable batteries a charger would be needed, maybe a magnetic one like recharging stations for hearing-aid.

A primary health concern for the receiver is whether or not it should go inside the ear or outside the ear. Putting the vibrator inside the ear, similar to the in-ear portion of a hearing aid, could certainly cause wax buildup in the ear and lead to ear infections. Having the vibrator outside the could still work, as the human skull is right on the outside of the ear and can pick up vibrations and sounds. Utilizing this behavior of the skull will allow the user to have their ear open and avoid ear infections and problems in the future.

System Portion	**Decision**
Signal Device	Smartphone: design an application for both iOS and Andriod devices, while tapping into the RFID capabilities that will communicate with the Receiver. Will be much easier to make.
Receiver	Hearing-aid style device: bendable around ear, with battery compartment and little LED light. Vibrator location and position still unknown. Range no more than 20 feet.

For the final decision, I have chosen to go with a smartphone as the signal device and code software to go with it. The software will be programmed to run on smartphones with either iOS or Andriod operating systems installed. This will allow us to utilize the RFID capabilities and simply the process for producing the receiver, since many other devices using RFID have been made. The receiver itself will be based on the hearing-aid, but it will not be exactly the same due to health concerns or restraints. Instead of having an in-ear piece, the receiver will be a bendable shaft that will go around the ear, with a battery hatch at the end where the signal receiver and LED light for low power will reside. The exact location of the vibrator, and whether or not it will be in one location or across the device, is still unknown, but it will be part of the receiver. The desired range for the receiver will be 20 feet max, to ensure that it does not communicate with other RFID devices.

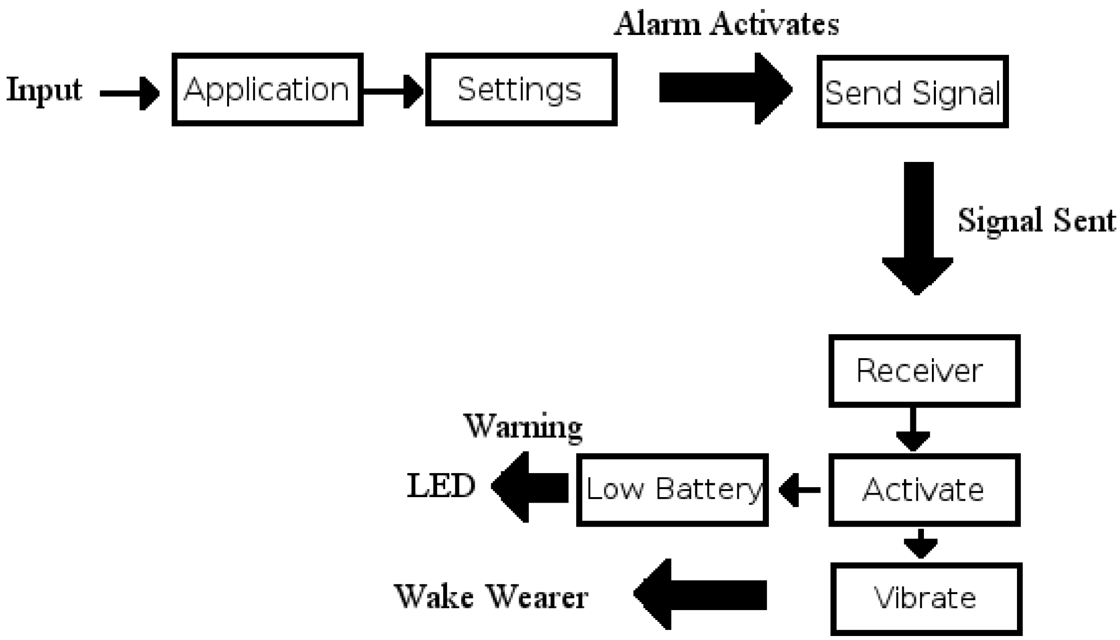

Figure 3: Final block diagram for solution to problem

TESTING PLAN

Once the prototype is built, several aspects of both the device and the receiver need to be tested. First and foremost, I need to test if the system works at all: setting the alarm time on the smartphone application, then waiting a bit for the alarm to go off; a vibration, indicating that the signal was sent to the receiver, will determine if the entire system is functioning. If the receiver does not vibrate, then first the signal will be checked, to see if there is a problem with the application itself. In the testing phase, the application should probably include some code to indicate that a signal was sent out, from there allowing us to check if the application was working properly and reduce the source of the problem to the receiver.

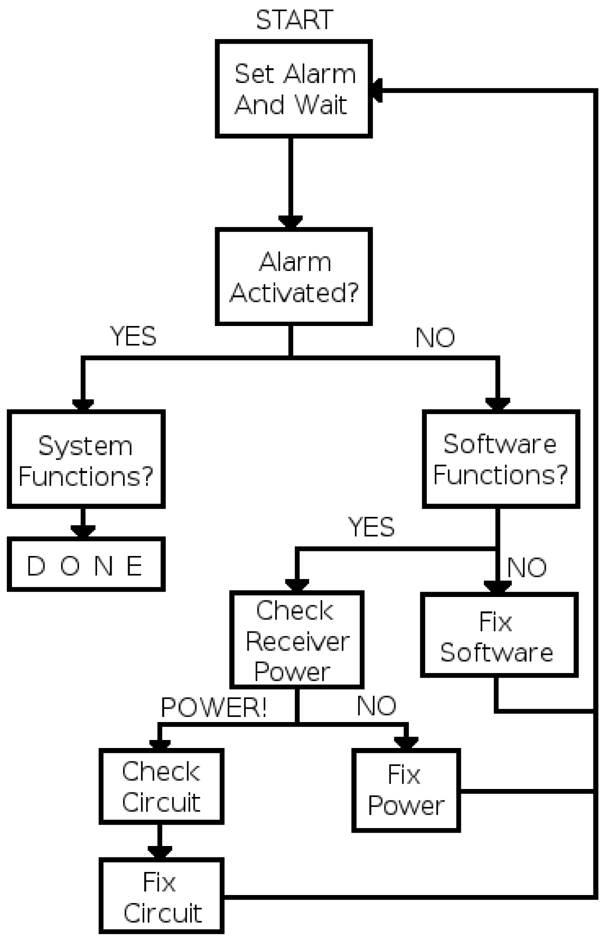

Figure 4: Test Flow for testing main function of prototype, signal device to receiver

At the level of the receiver, following the order of our behavior block diagram, there are two main things that need to be checked: the receiver power and the circuit itself. If there is no power to the receiver, then of course the receiver will not function. If the power is working properly, then it may be something more serious, like the actual circuitry. I will try to ensure that the circuitry within the receiver works before building the prototype.

After initial testing to see if the system functions at all, I will then begin testing variations on the functions of the system, such as the range between the smartphone and the receiver, the loudness of the vibration, user sleeping positions, etc. The receiver should still work within the desired range of 20 feet, and will be tested for maximum distance, which should not be too far from the device, if possible. The receiver will also be tested in terms of loudness, to see if the vibration is set at a good enough volume to wake the user and to also test the variability of the volume. Sleeping positions will be tested, to see if either sleeping on one's side or flat on one's back will work, hopefully both. Other factors may be tested in the process as well, but they probably will be better defined in the design stage.

Design Schedule

Week #	What Needs To Be Done
1	Finish learning programming of iOS and Andriod, any additional research
3	Basic schematics of receiver circuitry, begin simulating circuitry tests, basic schematics of receiver's physical design
4	Finish logic flow charts for software application, with some pseudocode
5	Looking at parts, have most of the circuitry design ready, protoboard testing
6	Software application code written, test and fix errors
8	Prepare parts ordering proposal, submit it
10	Receive parts

The basic outline of the design schedule is as listed in the above table. Over the course of the summer, I will be looking at materials and tutorials to learn about the ins-and-outs of programming in the iOS and Android operating systems. By the start of the term, I should have a grasp upon both systems and begin to start thinking about coding for the system. All additional research conducted over the summer that may cause changes to the design requirements or overall design will also need to be done by then.

With these in mind, I will give myself two weeks to begin working on schematics for the circuitry of the receiver and simulate those circuits to test for plausible designs. Based upon this, the shape and design of the hearing aid piece will be worked upon. During the course of those first few, the logical flow of the software application will be planned out, soon enough to be completely meted out by week five, with pseudocode prepared for the final writing of the software code. Before most of the code is written out, I will begin to look at parts for the receiver, run any last fixes on the circuitry design, and do physical circuit testing with a protoboard.

After writing the software code is finished, testing will be done to check for errors in programming and consistency. Further protoboarding will be done at the same time to ensure that the receiver will potentially work as intended. Over the next two weeks, a much fuller parts examination will be done and the ordering proposal will be prepared, after which it will be submitted. Parts will be ready by the end of the term.

Conclusion

Overall, the goal of this project is to construct an alternative, quieter way for people to wake up in the morning. It will be done through utilizing the capabilities of a smartphone, designing a software application to communicate with a custom-made receiver. This receiver will be hooked on the user's ear, naturally, and vibrate when it receives a signal to alarm. The resulting vibration alarm will wake the user by vibrating through the skull, alerting them to wake up. The design requirements have been determined as well as a preliminary schedule for the design process and testing algorithms. The final product will ensure that the user's alarm will not wake others present in a room while sleeping.