

# Territory Tracking and Restriction System



*Critical Design Review*

Top Dog Technologies

Denise “Merlin” Cuppett

John “Viper” Kaczmarek

Michael “Ice Man” Stewart

Chris “Jester” Wesp

Department of Computer Science

Texas A&M University

October 15, 2007

# Table of Contents

1	Introduction.....	3
2	Proposed design .....	4
2.1	Updates to the proposal design .....	4
2.2	System description.....	5
2.3	Complete module-wise specifications .....	5
3	Project management.....	7
3.1	Updated implementation schedule.....	13
3.2	Updated validation and testing procedures.....	15
3.3	Updated division of labor and responsibilities .....	15
4	Preliminary results .....	16

# **1 Introduction**

## **1.1 Problem Background**

Many people have trouble keeping their pets out of specific areas of their house. For instance, they may want to keep their pet off valuable furniture or keep it away from certain rooms and areas. When pet owners are absent from their houses, they are unable to control where their pets are allowed to go. The Territory Tracking and Restriction System will allow owners to track and control their pet's movement. When a pet enters user determined zones, the system will record that the pet entered the zone and deter the pet if it is an off limit zone. This is done by putting RF transmitters in off limit zones, and giving the pet a collar with a RF receiver. When the pet enters a zone, the receiver will receive the RF signal from the transmitter containing the zone id and deterrent settings. The receiver then responds according to the signal received, deterring if necessary. At any time, the owner can use a USB connection to get the data from the receiver to see what zones the pet has been entering and at what times.

## **1.2 Needs Statement**

There is a need to have a pet deterrent system that tracks pet movement throughout the house 24/7 by monitoring and documenting when a pet enters off-limit areas and deters the pet when needed.

## **1.3 Goals and Objectives**

The goal our group has set for this project is to create a network of receivers and transmitters that can record the general location of a pet and deter it from the off-limit areas.

Here is a list of objectives that need to be taken into consideration when designing our Pet Deterrent System:

- The prototype system must cost less than \$500 to be competitively priced based on the quality level it provides to the consumer.
- The commercial system should cost \$50 or less per receiver and \$30 or less per transmitter.
- The system must use a power source accessible to the public, such as a battery, and the power source must last at least 1 month without being replaced.
- The system must not harm animals or people.
- The system must function well in a typical indoor environment.
- The collars should be light, less than 1 pound, and comfortable for the pet.
- The system must be easy for the user to set up which is defined as the set up time taking less than 30 minutes.
- The system must be easy to use and adjust, any adult with basic computer knowledge should be efficient with the computer software after 1 week.
- The system should have a variable range that covers an area with a 1 foot radius to an area with a 20 foot radius.
- The system should document the zone and time when a pet violates a restricted location; should also record when a pet enters an allowable zone.
- The recorded information should be displayed to the user in an organized and understandable fashion.

## **1.4 Design Constraints**

The economic constraints of this system fall into two groups. First the prototype of the deterrent system should cost less than \$500. If the development of the project goes over \$500 then the project will go over budget. Secondly, this system is being designed with the intent of being a consumer product. The final system needs to be priced competitively. In order for the system to be affordable, the receiver should cost \$50 or less, and the transmitter should cost \$30 or less. Another commercial economic constraint is the power for the system. The power source needs to be something that is easily accessible and cheap to the consumer. Also, the power source for the system needs to be efficient, so that it does not have to be replaced often.

The physical constraints on the system are in the collar/receiver and transmitter design. The receiver needs to be lightweight (preferably less than one pound) and not hurt the pet. In addition to being lightweight, the receiver needs to be relatively small; a bulky receiver could hinder the pet's movement. The receiver also needs to not produce too much heat or it could aversively affect the pet; this means it is important that our voltage regulators do not produce too much heat. Further testing will be needed to determine what amount of excess heat is acceptable. The collar needs to be wireless when in operation, so the pet is not constrained by wires. The only exception to this is when the user is using USB to get the data off the receiver. The transmitter needs to have good battery life to prevent the deterrent zone from turning off. Also, the transmitter needs to be small enough, so that it will not get in the way of the user's daily life.

The physical setup of the receiver collar and transmitters needs to be easy for the user. Putting the collar on the dog and activating the receiver should be easy. Activating the transmitter and selecting the deterrent settings also needs to be easy for the user to accomplish with minimal instructions. To do this, the design needs to be simplified that the average consumer will not have any difficulties.

The software design needs to be user friendly and easy to setup for someone with basic computer knowledge. Also, the user interface should be simple and easy for the average user to be able to learn and use quickly.

The system needs to function in an indoor environment, so being able to work with obstructions such as furniture is needed.

Our prototype will not actually have a deterrent method due to possible safety regulations and constraints.

## **1.5 Validation and Testing Procedures**

The validation and testing will consist of a Range Test, Deterrent Test, Power Test, Software Accuracy Test, Accuracy Stress Test, and a Software Suite Test. The range test will make sure that the distance objectives were met satisfactorily. The Deterrent Test will make sure that the receiver accurately identifies an off limit zone. The Power test will help determine the power efficiency of our system. The Software Accuracy Test will make sure the software located in the hardware (receiver and collar) works. The Accuracy Stress and Software Suite Tests will determine if the software suite is working correctly.

## **2 Proposed design**

### **2.1 Updates to the proposal design**

The transmitter design has been altered. After further research into the capabilities of RF transmitter chips Top Dog Technologies has chosen to use TXM-315-LR from Linx instead of TRF7960 from TI or ADF7020 from Analog Devices. The Linx chip has a better variable range than the alternate choices and is easier to set up. Also, we have decided to use a simple PIC on the transmitter to control the transmitter

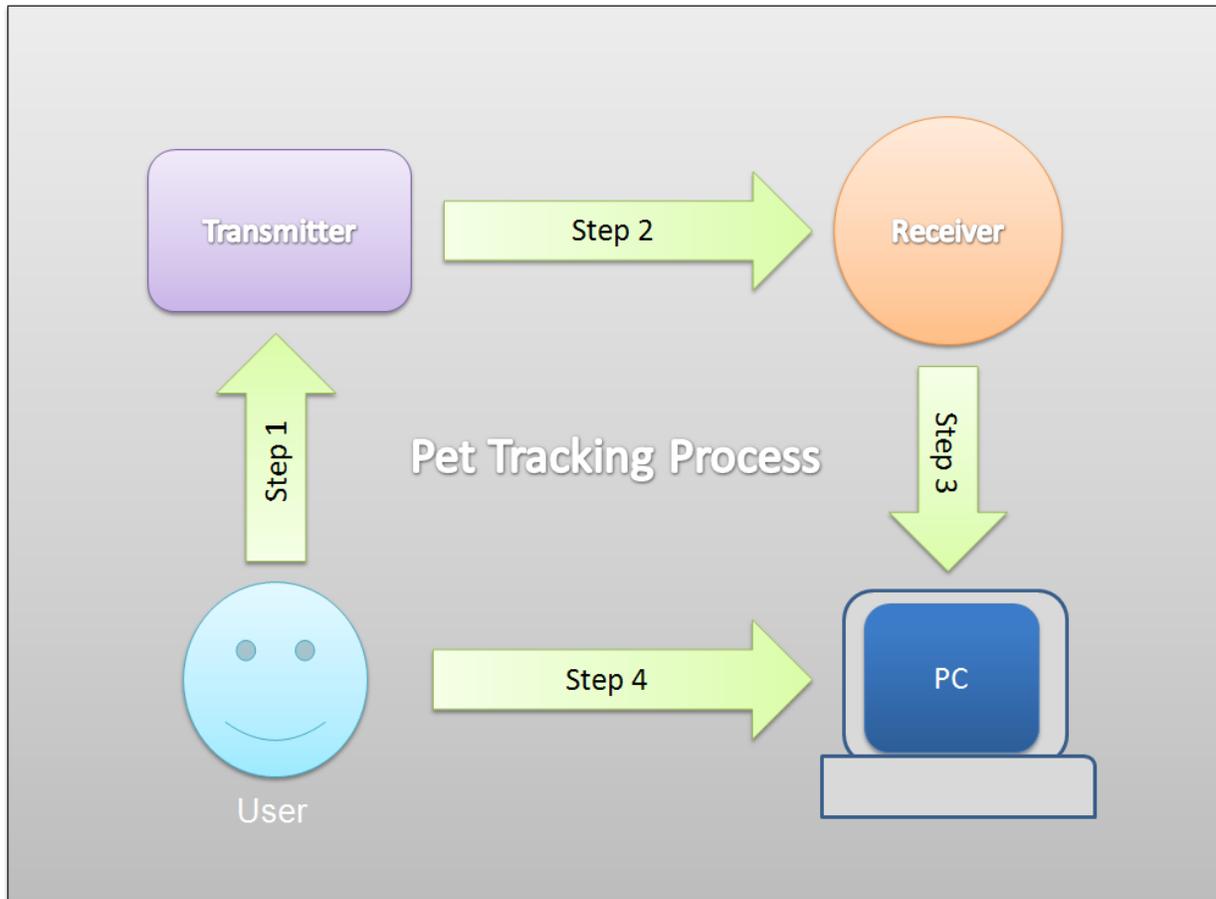
ID, deterrent setting and sending data to the transmitter chip. The change in the transmitter design does not make a large impact on the budget, though the budget has been revised to reflect the new parts.

Our original plan was to attain an Evaluation board for the CC1100 chip from Texas Instruments. This board had the chip, a microcontroller, and USB functionality. Basically we would have to do zero hardware work and only program the microcontroller to do what we wanted it to. That is, to evaluate the incoming signal, document it, and determine if a deterrent is needed. Due to poor support and the fact that they would not have any in stock until late October we had to choose a new chip, the Linx RXM-315-LR-S. Since we lost the microcontroller and USB capabilities of the Evaluation board we have added a PIC with a USB interface. Because of this change we will be spending more time on programming since more has to be done on the PIC to get everything to work properly together. We will also need to spend time debugging until we find a working circuit model since many things have to be adjusted to get the USB, memory module, etc. adjusted. The nice part about the new chip is the better range and the easier circuitry surrounding it. The CC1100 was an in-depth chip that required a lot of circuit adjustments just to change the transmission power. The Linx chip is a vast improvement because all it needs to vary the range is the input resistance, i.e. all you need to do is vary the voltage on input or place a potentiometer there. This is much simpler than the Ti chip not to mention the Linx chip has a better range and can get inside of closer distances. Though more work will be needed in the long run to get all the components of the receiver working, this is the optimal solution found thus far for this project.

The software design for the receiver has not changed very much. Previously the design called for a self adjusting resolution that would automatically deal with memory constraints. Originally the program would store a timestamp in seconds and once the memory got to a certain level of fullness the data would be rewritten in a time resolution of minutes and then hours and finally days. Our new receiver design has a large amount of memory compared to previous designs. That means data loss is less of a risk but to be completely robust the software will also store a count for zone entries for each of the zones. The importance of high resolution data has been deemed less vital to a typical user. Therefore, data will have only two resolutions minutes and days. All data older than a week will have the lower resolution.

## **2.2 System description**

The system will be comprised of primarily three main components; the transmitters which create the zones of interest, the receivers which are in the pet collars, and the client program which is the user interface to monitor all the information. These components will interact in the following way



#### Step 1

The initial step in the process is for the user to define the zones of interest. This is done by placing the transmitters in areas in which they want to either track or deter. They must then define a range for the radius and select which pet collars will be set to the deter mode and which collars are set to only track. This step is refined by testing the ranges and making sure the system is functional.

#### Step 2

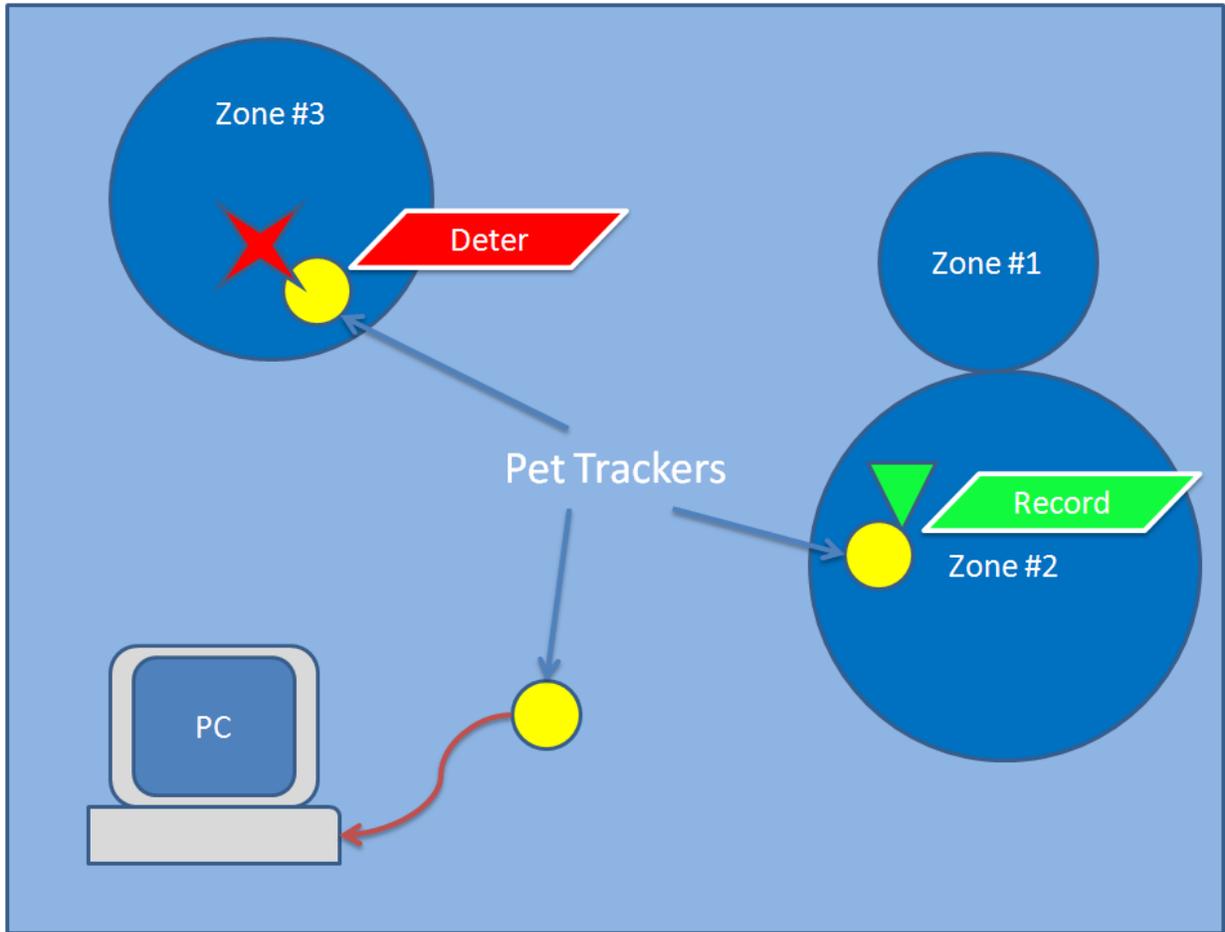
This is the primary step in the tracking process. The transmitter will continually broadcast a repeating signal stating its unique transmitter id and a code for which pets to deter and which to only track. Once a receiver comes in range it will process the message and respond accordingly. If the pet is not supposed to be in that zone it will trigger the deterrent mechanism. In either case however the zone id will be stored in the memory along with a time stamp

#### Step 3

In this step the user takes the pet collar and connects it to a home computer. All data is dumped to the PC and cleared automatically from the on board memory.

#### Step 4

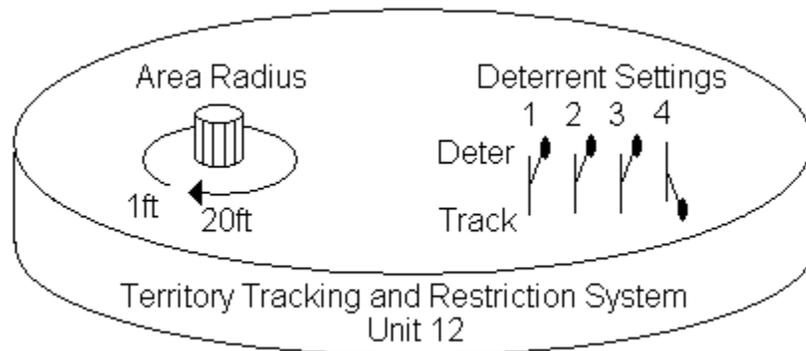
In the final step the user interacts with the client program to monitor the recorded data. They can choose to save the data for use later or simply discard it. Using this information they can make adjustments to the zones or simply return the pet collar to the animal and continue in a new iteration of step 2



### 2.3 Complete module-wise specifications

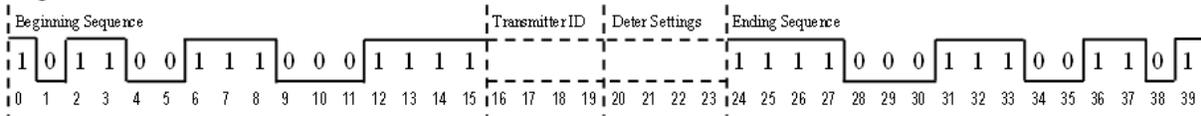
#### Transmitter

The transmitter has been designed to meet the objectives that specify that the product should be easy for the user to set up and adjust. Top Dog Technologies decided that the system should be usable with or without the software suite. A diagram of how the end project should look is found below. Each transmitter is given a unit number that represents the unique ID in decimal. Each collar is given a number 1 through 4 and the user may specify for each pet if the transmitter should track or deter by altering the appropriate switch in the deterrent settings section. The radius is adjustable by a knob, and should allow the user to create a circular zone with a radius between one and twenty feet.

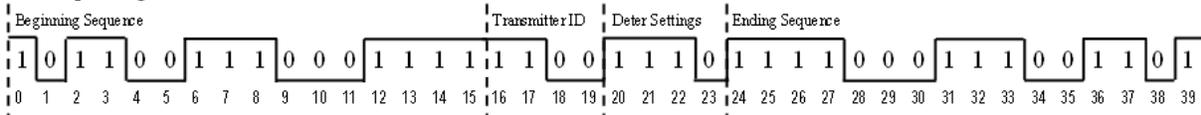




### Signal Transmitted



### Example Signal Transmitted



The user should be able to place a transmitter in his or her home, adjust the range with the knob and flip the switches to determine which pets are being deterred and which are being tracked. The transmitter then sends the appropriate signal, including the transmitter ID and deterrent settings, and when a receiver enters the zone the appropriate actions should be taken. The transmitter constantly sends a signal and then sleeps to save power. Here is a complete list of parts needed to create the transmitter that Top Dog Technologies has designed.

Part	Part Number	Vendor	Distributor	Price	Number per Transmitter
Transmitter	TXM-315-LR	Linx	Digi-key	\$7.46	1
PIC	PIC18F2455	Microchip	Microchip	Free Sample	1
Voltage Source				< \$5	1
Voltage Regulator				< \$5	1
Switches				< \$2	4
Potentiometer				< \$1	1
Resistor				< \$.50	1

### Receiver

The receiver module is what receives signals from the placed transmitters and deters the pet from the location, as well as stores information about the incident. Ideally, the module will consist of only a few ICs with the addition of a small battery. This should be a rather small unit upon completion and will easily fit on a pet's collar with ease and comfort, meeting our set objectives. The receiver will receive and process any information it gets from the designated transmitters. In order to process the data and eventually store it we will need a microcontroller, memory, and an interface. The method of deterrent for our application will be notification by an LED.

The receiver used is the RXM-315-LR-S from Linx Technologies which is the counterpart to the transmitter that is used in our setup. We will receive information across a 315 MHz signal. The -112 dBm sensitivity on this module is excellent for picking up the weak signals transmitted over short ranges on the transmitter. The module itself is rather simple since it only has sixteen total pins, nine of them being No Connections. Upon hookup, the only pins used are the Vcc, GND, and the DATA pin. Whenever the receiver picks up a signal emitted from the transmitter it will send it across the DATA pin to be processed in the PIC.

The microcontroller in our case is a PIC18F2455. It handles all of the information processing and is the core behind the receiver module. Once the receiver gets a signal from the transmitter it sends it from its DATA pin to the PIC's Rx pin, which receives data and stores it to a register. Currently, our bit setup is to send a 16-bit open identifier, the 4 bits needed for the transmitters, the 4 bits needed for the hot-encoded pet identifier, and 16 bits for a close identifier. The PIC then processes the data accordingly and judges if a deterrent is needed. If the bit for the particular pet is high, then it means it is an off-limit zone. If this happens, a signal is sent out of the VPO pin to the gate of a transistor which has an LED attached to it. When a signal is received the LED will blink.

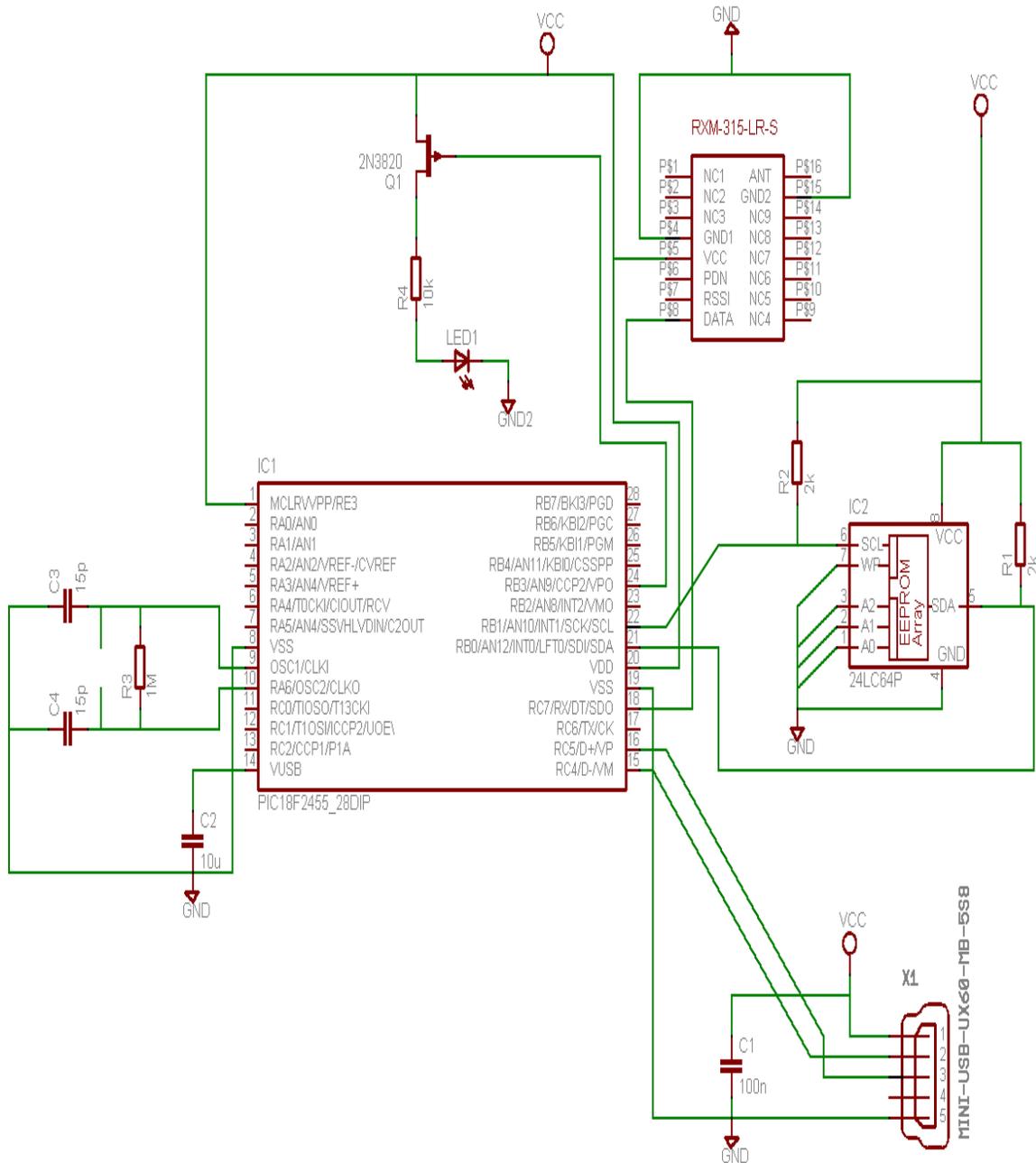
All instances of data that the PIC receives will be stored so that the user can follow the patterns of their pet(s). The information needed to be stored is the transmitter ID (which is four bits) and a short timestamp so the user can know when and where their pet has been. Since the PIC has only 256 bytes of memory to work with we needed an external memory module. For that we attained the Microchip 24LC64 memory module which has 64K memory that can be accessed serially. This means we only have to worry about two pins, SDA and SCL, which we can connect to the PIC respectively. Storing and retrieving the data will now work over the same line and the address pins are not needed.

The user interface to the stored data will be by USB. This is by far the most complicated aspect of the entire module because it requires a large amount of code and a few extra components to allow it to run correctly. The USB connector itself will be a type-B connector that has 4 DIP pins on it. Those pins are Vcc, GND, D+, and D-. The D+ and D- are the data lines for USB. These pins are connected to their counter D+ and D- on the PIC. In order to send signals correctly, the proper frequency must be obtained. To do this we need an external oscillator. Based on the recommendations from the tech doc we need a 16 MHz ceramic oscillator and two 22pF capacitors to make it run properly. The two sides of the oscillator are fed into OSC1 and OSC2 on the PIC. From here they are fed into the PIC's internal PLL, which multiplies the frequency to make it run properly for USB purposes.

Here is a complete list of the parts needed to run the receiver:

<b>Part</b>	<b>Part Number</b>	<b>Vendor</b>	<b>Distributor</b>	<b>Price</b>	<b>Number per Reciever</b>
Receiver	RXM-315-LR	Linx	Digi-key	\$13.56	1
PIC	PIC18F2455	Microchip	Microchip	Free Sample	1
Memory Module	24LC64	Microchip	Microchip	Free Sample	1
USB Module	-	-	Digi-key	< \$5	1
Oscillator	-	-	Digi-key	< \$1	1
Capacitors	-	-	-	< \$1	4
LED	-	-	-	< \$1	1
Resistors	-	-	-	< \$1	4
MOSFET	-	-	-	< \$1	1

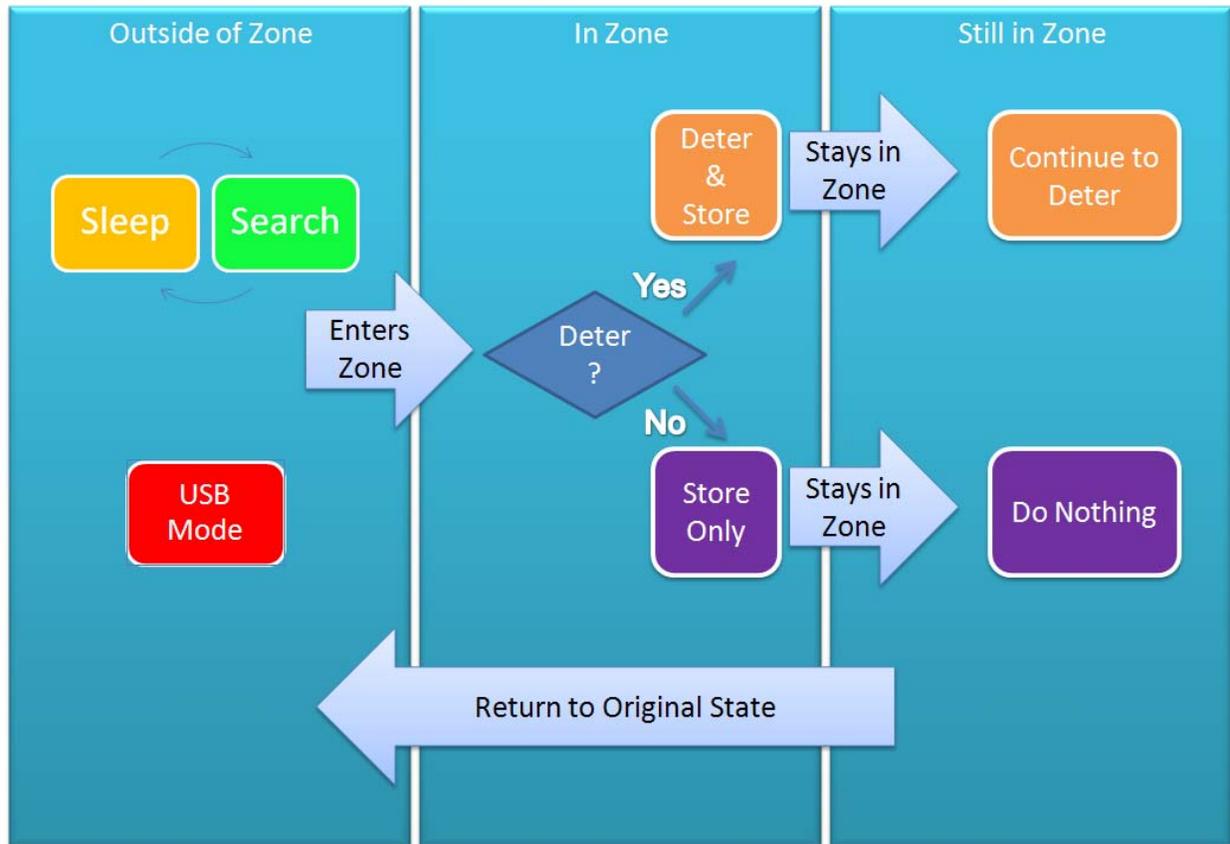
## The Receiver Design



## Receiver Software

The receiver continually listens for a transmitter's beginning sequence. Once it has this it records the tag and processes the deterrent data to see if the pet it is assigned to is allowed in the zone. If it is not a signal for the deterrent mechanism is sent and it will continue for one second intervals until the transmitter's signal is no longer detected. In either case the event is recorded in the on board memory. Data in the current weekly time window will be 6 bytes long (5 bytes for the Date/Time + 1 byte for the Tag

information). Data outside of this window will only be 4 bytes long (3 bytes for the Date + 1 byte for the Tag information). If the memory becomes filled (meaning the user has not cleared the data in a long time) old data will begin to be written over, but a counter will still continue to track the total number of events per transmitter. In the limit “all” data will be only 4 bytes per event. Our memory is 64kB which means the system could store almost 16,000 events. The program flow is modeled below in the graph.



### Computer Software

The software is going to be composed of a GUI (graphical user interface) written in Java that will allow the user to upload and view data from the receiver. The software will interface with the reader through USB. This will download all of the data containing the time and zones the pet entered since the data was last read. After the data is read in, it is erased from receiver, so new data can be saved. The software will be able to pick up the pet's ID from the data along with the timestamps to associate the timestamps with a specific pet.

Once the data is read into the software, it is automatically saved to disk and loaded into the software for display. The user can then view the data in multiple different ways. The timestamps can be viewed in graphs and pie charts, so the user can better see which zones the pet is entering. A tree view is also available to see the list of timestamps and pets. Also, the user can choose to load in more old data from saved files.

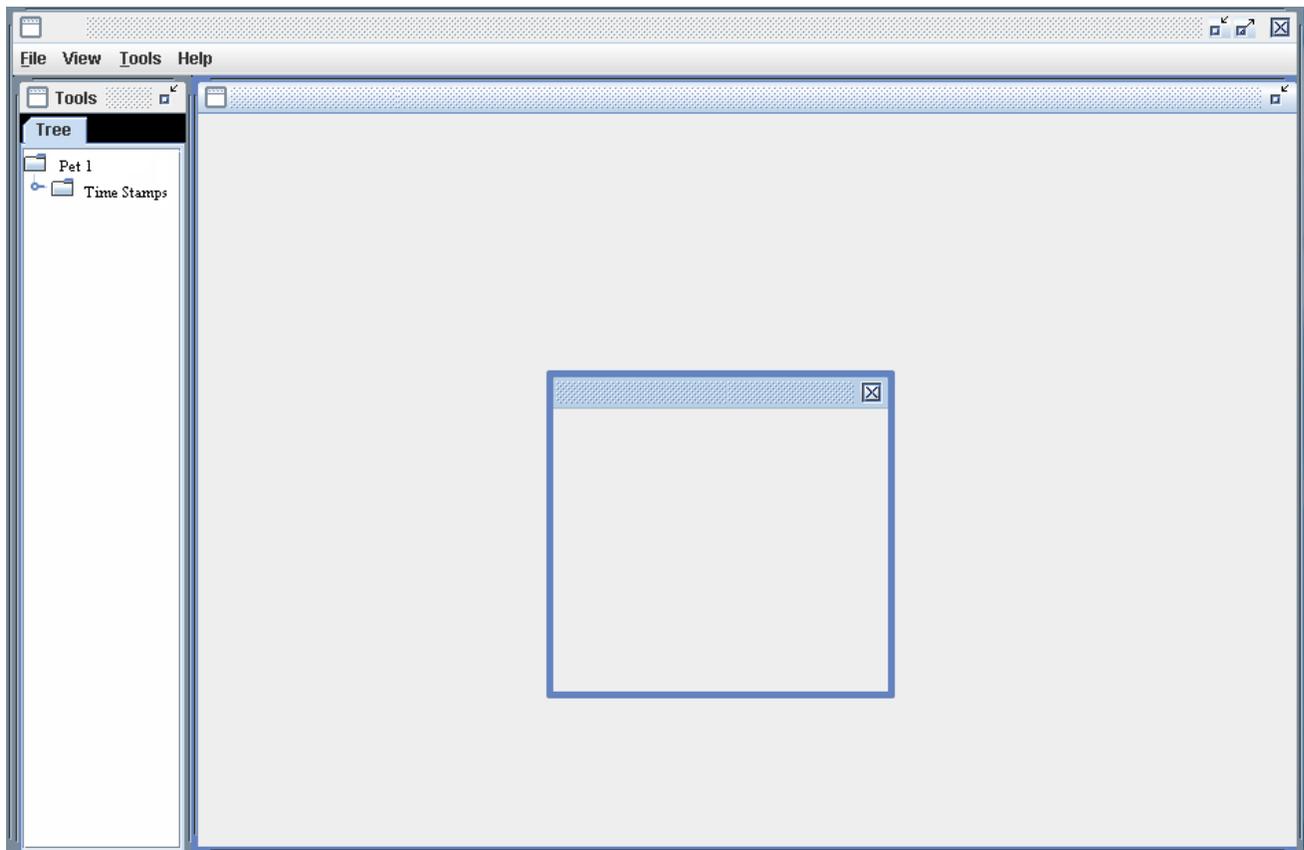
The main structure of the program:

Main Class – Initializes the program and starts the GUI

GUI Classes – Contains the graphical user interface implementations and event handling

Pet Class – Contains the data for one pet, including: pet id, timestamps, misc. info

Data Class – Contains the data storage and retrieval functionalities



Preliminary GUI design, still in development

This is still the preliminary design of the software and changes will probably occur at some point as features are added or changed.

### 3 Project management

Briefly review the roles and responsibilities of each team member (e.g., team leader, systems design, software design, hardware design, finance and purchases, testing, technical reporting, etc.) Provide any updates on the mechanisms that you are using to manage the project as a team, such as brainstorming sessions, keeping track of progress on every task, etc. Is your management approach working? If not, what types of improvement are you putting in action?

#### 3.1 Updated implementation schedule

The main dependencies in the schedule are between the testing, building, and perfecting of the receiver and transmitter designs. We are already set back one week, because the hardware came in late and we were unable to test the receiver and transmitter. Before we can actual build the receiver or transmitter systems, we have to test to see if we can send and receive with the transmitter and receiver chips. After those tests are completed, we can build the full receiver and transmitter systems. Once the systems are built and tested for correctness, we can further test and improve them.

The software suite does not have a lot of dependencies. The only part of the software that is dependent on something else is the USB data input. We cannot test the softwares ability to read in data from USB until we get the reciever working properly.

**The Updated Gantt chart:**

Weeks	7	8	9	10	11	12	13	14	15
<b>Tasks</b>									
Test Transmitter and Receiver Pair	█	█							
Create Software with Basic Functionality	█	█							
Test Storage System and Software Compatibility	█	█							
Build Transmitter Disk		█	█						
Build Collar Receiver and Deterrent Signal		█	█						
Build Storage System		█	█						
Perfect Transmitter Disk			█	█					
Perfect Collar Receiver and Deterrent Signal			█	█					
Perfect Collar Storage System			█	█					
Add Functionality to Software	█	█	█	█	█				
Focus on Parts of the Project that Failed Testing				█	█	█			
Test and Correct all Aspects of the Project						█	█	█	
Develop Final Report							█	█	█
Develop Final Presentation								█	█

### **3.2 Updated validation and testing procedures**

Top Dog Technologies still believes that the best way to test a design is to test it at various stages of the development. These early component tests will be completed so that issues may be solved before the entire system is tested. We will now discuss our tests that will prove the system is working as a whole.

#### **Range Test**

To begin with Top Dog Technologies is going to demonstrate the Territory Tracking and Restriction System with one transmitter unit and one receiver unit. The receiver unit on the collar is set to portray pet 1. The transmitter unit is set with switches to have the ID 0001. The transmitter is now turned on and the range adjusted with the knob to twenty feet. The deterrent setting for pet 1 is set to deter. The collar should be moved to enter the twenty foot radius and the LED on the collar should light up, indicating that the deterrent device is activated. This test must be repeated for different transmitter ranges, different entry angles and different obstacles for interference.

#### **Deterrent Test**

This test will begin with the receiver unit on the collar portraying pet 1 and the transmitter deterrent setting should deter pet 1 but track the other pets. The collar must enter the zone and the LED should light up. The transmitter deterrent setting should now be adjusted to track pet 1 and deter the other pets. The collar must enter the zone and the LED should not light up. This test must be repeated with the receiver unit on the collar portraying pet 2, 3 and 4.

#### **Power Test**

The power source in the transmitter and receiver should be checked before and after the devices have been running for a long period of time. This will help determine the efficiency of our system. The range should also be checked at different times to ensure that the radius does not diminish when the battery begins to drain.

#### **Software Accuracy Test**

After the hardware has been validated the software must be tested. To test the software the team members must keep a handwritten record of when the collar enters a zone and what the zone is set to. To begin this test the receiver unit on the collar should be set to portray pet 1. The transmitter should have the ID of 0001 and be set to deter pet 1. The collar should move into the transmitted zone multiple times while the transmitter is being adjusted. The ID on the transmitter should occasionally be changed. The transmitter deterrent setting should also switch between tracking and deterring for the pet. The collar should then be connected to the computer and the data represented with the software must be checked for accuracy. This test should be done with the receiver unit portraying different pets.

#### **Accuracy Stress Test**

After the accuracy of the software has been checked for a handful of entries, the collar will be taken in and out of a zone many times to portray the system running for a week. The collar will then be connected to the computer and the software suite will be tested for accuracy.

#### **Software Suite Test**

A tester that is not involved with the project should install and use the software suite. The tester should make sure that it installs cleanly and the suite overall is easy to use. Any features, options or buttons should be checked in random orders to make sure that the system does not crash. The data should be saved and then recovered to make sure that files are not corrupted. The suite should remain running for a long period of time to check for memory leaks.

All of these tests will verify that our product works and meets the need to have a pet deterrent system that tracks pet movement throughout the house 24/7 by monitoring and documenting when a pet enters off-

limit areas and deters the pet when needed. The extent that we reach this need is dependent on how well the product completes the tests.

### **3.3 Updated division of labor and responsibilities**

Denise Cuppett will be in charge of overall hardware design and system testing. Initially, while everyone else is building the hardware components Denise will oversee the process and keep track of all updates and issues that may occur. Specifically, in terms of actually constructing the project, Denise will work on building the transmitter module and programming the PIC that goes with it. Though we only have one PIC programmer, we have three demo boards to work with and an unlimited supply of MPLAB IDE software to work with. This will allow multiple PICs to be worked on at once and give Denise the freedom to work on it without interruption. Denise will oversee all the testing that goes on in our project as well. Any time a certain section of our system is complete it will undergo rigorous testing by its creator and all aspects of this will be administered by Denise.

#### **Denise's Deadlines**

Build Transmitter – October 29

Program Transmitter – November 5

Test Transmitter – November 14

John Kaczmarek, in part due to his software experience, will work on the bulk of the low-level PIC programming that goes into both the receiver and the transmitter. In turn, he will be specifically working on using the A/D converter, the USB coding, the EEPROM coding, and overall register access. Though Denise will generally be in charge of all the parts of the transmitter, John will help with the PIC part as to ease some of the workload off of that part. In terms of the whole project, John will manage the system design and all of the technical documentation that goes into. In the end, it will be John's responsibility to work on making sure our finished product is as close to professional and customer ready as possible.

#### **John's Deadlines**

Program Transmitter – November 5

Program Receiver – November 5

Documentation – November 19

System Overview Critique – November 19

Chris Wesp will manage and develop all of the software that will go into the computer interface. The software suite will be an elaborate program that allows the user to download all of the data from the receiver and view it in a way that manages and tracks the pet's movements. In the beginning stages Chris will help to test and setup the transmitter and receiver modules as to ensure they are capable of meeting the requirements for this assignment.

#### **Chris' Deadlines**

Transmitter/ Receiver setup – October 24

Have a working GUI – October 29

Implement Receiver into GUI – November 14

Test all software – November 19

Michael Stewart will be in charge of the team as a whole and all the financial responsibilities. It is his job to manage the team and keep track of everything that is going on at a certain time. Any time that a new part needs to be ordered he will be manage it into the budget and oversee the purchase. As far as the project goes into design, he will construct the hardware portion of the receiver module and work with John on programming it. When it is finished he will test it with Denise to ensure that everything is working properly.

#### **Michael's Deadlines**

Build Receiver – October 22

Program Receiver – November 5

Test Receiver – November 14

#### **4 Preliminary results**

Unfortunately the creation of the prototype Territory Tracking and Restriction System was delayed. We changed a few modules such as the transmitter and receiver, and did not receive the parts in time to test them. Since a large portion of the design is hardware, we are not able to test without the design solidified and the parts connected together.