7CC003 Distributed and Mobile Computing Report

This report describes the work done on the 7CC003 practical work which includes research done with the aim of gathering information on the tools to be used, the projects design and method are explained as well as the implementation of said design and a critical evaluation of all work done for the module.

The task is to create a wireless home security system with arduinos and XBee wireless transceivers, the first arduino acts as the control device, the second arduino is the exterior security device, if movement is detected it activates a light, if a button is pressed it activates a buzzer on the first arduino, controls a servo that is activated by the first arduino. The third arduino is the interior security device and when movement is detected it alerts the first arduino with audio and text, it controls a pressure switch which when tripped it activates a buzzer on the first arduino and flashing lights on the third arduino. The second part of the task is to create a android application designed to integrate with the security system.

The aim of this project is to create a fully functioning wireless home security system as detailed in the introduction. The first objective is to successfully set up two arduinos that perform all necessary functions. The second objective is to add the third arduino that also carries out its tasks. The third objective is to integrate the android application with the security system.

The first reference by Severance, C. (2014) talks about the arduino microcontrollers history, the reasons behind its conception, its advantages such as its low cost, easy to work with etc, essentially a general overview on the arduino itself. This is useful as it gives background information on the hardware that is the focus of the project which helped form the solution for the task at hand thanks to the prior knowledge of the tools that would be used. This source is similar to the work of Teikari, *et al*. (2012) as both sources detail the arduino especially its advantages however the first source is quite general in its description and history of it while the second source goes into more detail by giving a practical use of the arduino microcontroller which while it is more useful and therefore is the better reference to help form the solution to the task, the first source still provides useful background information that is invaluable for the design stage.

The second reference by Teikari, *et al*. (2012) shows how the arduino microcontroller can be used to help vision research as a visual stimulator, this is useful as it gives a practical example of the arduino being put to good use in the scientific field and as such it justifies its choice by detailing the arduinos advantages for this purpose which includes how it is inexpensive and not difficult to use all of which helps in designing the solution the problem for this project. This source is similar to the work of Agudo, *et al*. (2014) as both sources show practical examples of the arduino in use as well as their advantages however the second source provides a more balanced viewpoint by also talking about the arduinos limitations compared to more expensive systems and therefore is the better source for improving the end product however the first source is still useful as it provides another viewpoint to compare practical examples with and improves the design of the solution for the task at hand.

The third reference by Agudo, *et al*. (2014) gives a practical example of how the arduino can be used as a colour sensor, this is quite useful as immediately this shows how useful the arduino can be which critically applies directly to this project with its use of the arduino, it also talks about the arduinos advantages such as how cheap it is and easy to use, however it cautions that although it performs the task quite well it is not up to the standards that professionals in this field would need. This source is similar to the work of Luiz, *et al*. (2014) as both sources detail the arduino in use the as well as its advantages however the first source also shows the drawbacks of this and therefore is more useful as being aware of the disadvantages is quite important however the second source still helps improve the overall design thanks to its practical example.

The fourth reference by Luiz, *et al*. (2014) gives an example of how the arduino can be used in the electromyography field as a surface electromyographic signal amplifier, this is useful as it provides another viewpoint of practical use of the arduino in a separate field detailing its advantages of low cost, reliability etc and provide their view that the system designed is suitable and can improve. This source is similar to the work of Severance, C. (2014) as both sources detail the arduino, the second source providing a general outlook on the arduino including the history while the first going into more detail with a practical example of the arduino in use which helps improve the design on a more functional level and as such is the better source but the second is still helpful as it provides a good mindset and general knowledge of the arduino hardware.

The hardware was designed the way it is after performing research in this area in order to create the best solution with the time allotted and the resources available, as well as it is suitable because it successfully accomplishes the task of simulating a wireless security system with arduinos, XBee's, buzzers, led's, buttons and a servo.

As shown in figure one the first arduino has a serial connection with an XBee using pins 2 and 3, it has a button connected using pin 11 to activate the servo on the second arduino, it has a buzzer connected using pin 13 which is activated when it receives a signal to do so from the second or third arduino. The second arduino has a serial connection with an XBee using pins 2 and 3, it has a button connected on pin 11 which activates the buzzer on device 1 to simulate a door bell, it has a servo connected on pin 9 which is activated when the button is pressed on the first arduino and so it rotates 0 to 180 degrees to simulate the opening of gates. The third arduino has a serial connection with an XBee using pins 2 and 3, it has a button connected on pin 8 to simulate a pressure sensor which activates the buzzer on device 1 as an alarm and flashing lights in the form of an led connected on pin 9.

The aims and objectives are sufficient for this project as they successfully establish the core motivations behind the project as a whole and each aspect of it however they could be improved by going into further detail on the different aspects of the project such as the hardware considerations in order to better reflect on the complexity of the projects design choices and the effectiveness of the hardware involved. Another aspect would be the software side as this should have been focused on more in the aims and objectives due to its impact on the users experience therefore making this more prominent would be beneficial to the end products user friendliness.

The development process for the project had its strengths and weaknesses and as such it could be improved for the betterment of a similar task in the future, one strength of the development process would be the use of adequate alternative hardware due to the preferred hardware not being available as through the use of buttons replacing the various sensors an effective wireless security system is still able to be simulated. This is important as it shows the development process was flexible enough to allow for such changes in the hardware used, however this could be improved by performing research in the planning stage on alternative hardware for all of the projects hardware specifications in order to provide failsafe options and from that increase the development processes flexibility even further, should a similar project be encountered in the future this become part of its design and will be the better for it thanks to the lesson learned here.

A strength of the development process for this project would be the research performed early on enough to affect the end product produced. This is important as research was carried out on academic papers covering the arduino device in order to gain information on its uses, constraints and general information about it as this will allow for greater understanding of the hardware involved and positively influence the programming. Therefore the research was paramount to the successful creation of the security system however this could be improved by also researching into the software side of this project as that will inevitably better the end result.

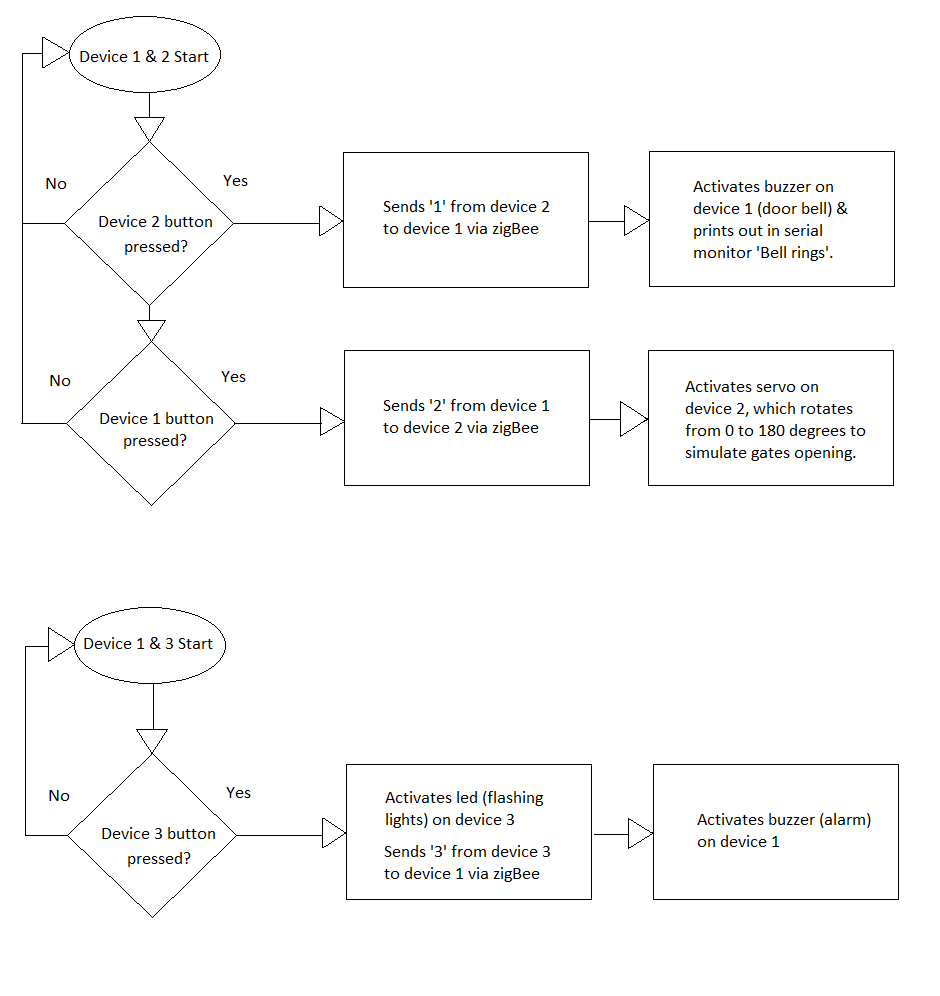
A weakness of the development process is that due to time constraints there is no android software to take advantage of the wireless security system, this is unfortunate however it provides a useful opportunity to learn from this so that it does not happen in a similar situation. Therefore in order to improve this for the future steps should be taken to map out the workload for a particular project in the design stage as this will result in better time management and ensure that all parts of the project are worked on as much as they are needed to which is an integral part of successfully completing a similar project such as this.

A weakness of the development process is that although it effectively uses alternative hardware to simulate a wireless security system it is still only able to simulate the motion and pressure sensing capability that is the prerequisites for this project. This could be improved by taking steps to ensure that the hardware requirements are met, this could be achieved by performing research into the area in order to find adequate alternative hardware that is available as this will allow for the successful creation of the end product without having to simulate missing hardware detracting from the systems capabilities.

Therefore in conclusion the project has had its successes and failures such as effective research and poor time management all of which have been taken into account as part of the projects critical evaluation in order to assess how well certain aspects of the project went and how this could be improved which is important as all of this can be learned from to repeat the advantages and avoid the disadvantages for a similar project in the future.

Appendix:

* Fig 1 - System flow chart



* Device 1 code -

// Device 1 - Remote control

#include <SoftwareSerial.h> // Imports the software serial library

SoftwareSerial zigBee(3,2); // Creates a serial connection between XBee and Arduino using pins 3 and 2

// Buzzer

int buzzSignal = 0; // Creates an integer variable buzzSignal and sets it to 0

// Button

const int buttonPin = 11; // The number of the pushbutton pin

int buttonState = 0; // Variable for reading the pushbutton status

void setup(){

Serial.begin(4800); // Begins hardware serial (PC<->Arduino)

zigBee.begin(4800); // Begins software serial (PC<->Arduino)

pinMode(13, OUTPUT); // Sets a pin for buzzer output

}

// Buzzer

void loop(){

// Buzzer code for door bell

if (zigBee.available()) { // If zigBee connection detected

buzzSignal = zigBee.read(); // Whatever is read from zigBee connection becomes buzzSignals value

if(buzzSignal == '1'){ // If buzzSignal = 1 do following

digitalWrite(13, LOW); // Buzzer is off

}

else{

buzz(13, 2500, 500); // Buzz on pin 13 at 2500Hz for 500ms

Serial.println("Bell rings"); // Prints out in serial monitor

delay(1000); // Wait for a second between buzzes

}

}

// Buzzer code for alarm

if (zigBee.available()) { // If zigBee connection detected

buzzSignal = zigBee.read();// Whatever is read from zigBee connection becomes buzzSignals value

if(buzzSignal == '3'){ // If buzzSignal = 3 do following

//digitalWrite(13, LOW); // Buzzer is off

}

else{

buzz(13, 2500, 500); // Buzz on pin 13 at 2500Hz for 500ms

Serial.println("Intruder Detected"); // Prints out in serial monitor

delay(1000); // Wait for a second between buzzes

}

}

//button

buttonState = digitalRead(buttonPin);

if (buttonState == HIGH) {

// If button pressed (motion sensor) send following to device 1

zigBee.print('2'); // Sends 2 over zigBee connection

Serial.println("Bell successfully rings"); // prints out in serial monitor

delay(500); // Wait for half a second

}

}

// Buzzers configuration

void buzz(int targetPin, long frequency, long length) {

long delayValue = 1000000/frequency/2; // Calculate the value of the delay between transitions

// One second's worth of microseconds, divided by the frequency, then split in half since there are two phases to each cycle

long numCycles = frequency \* length/ 1000; // Calculate the number of cycles for proper timing

// Multiply frequency, which is cycles per second, by the number of seconds to get the total number of cycles to produce

for (long i=0; i < numCycles; i++){ // For the calculated length of time

digitalWrite(targetPin,HIGH); // Write the buzzer pin high

delayMicroseconds(delayValue); // Wait for the calculated delay value

digitalWrite(targetPin,LOW); // Write the buzzer pin low

delayMicroseconds(delayValue); // Wait for the calculated delay value

}

}

* Device 2 code -

//Device 2 - exterior

#include <SoftwareSerial.h> // Imports software serial library

#include <Servo.h> // Imports servo library

SoftwareSerial zigBee(3,2); //Creates a serial connection between XBee and Arduino using pins 3 and 2

// Button

const int buttonPin = 11; // The number of the pushbutton pin

int buttonState = 0; // Variable for reading the pushbutton status

// Servo

Servo myservo; // Create servo object to control a servo

int servoSignal = 0; // Creates and sets integer servoSignal to 0

int pos = 0; // Creates and sets integer pos to 0 which stores the servo position

void setup(){

Serial.begin(4800); // Begins hardware serial (PC<->Arduino)

zigBee.begin(4800); //Begins software serial (PC<->Arduino)

pinMode(buttonPin, INPUT); // Initialize pushbutton pin as an input

myservo.attach(9); // Sets servo to pin 9

}

void loop(){

// Door bells button of the device one

buttonState = digitalRead(buttonPin);

if (buttonState == HIGH) { If button pressed (motion sensor) send following to device 1

zigBee.print('1'); // Sends 1 over zigBee connection

Serial.println("door bell rings"); // Prints out in serial monitor

delay(500); // Sets a delay of half a second

}

else{}

// Servo

if (zigBee.available()) { // If zigBee connection detected

servoSignal = zigBee.read(); // Whatever is read from zigBee connection becomes servoSignals value

if (servoSignal == '2'){ If servoSignal = 2 do following

myservo.write(pos); // Tells servo to go to position in variable 'pos'

delay(15); // Waits 15ms for the servo to reach the position

}

else{

for(pos = 0; pos < 180; pos += 1){ // Goes from 0 degrees to 180 degrees

myservo.write(pos); // Tell servo to go to position in variable 'pos'

delay(200); // increasing time by this formula 180 degrees \* 200 == 36000

}}}}

* Device 3 code -

//Device 3 - interior

#include <SoftwareSerial.h> // Imports software serial library

SoftwareSerial zigBee(3,2); //Creates a serial connection between XBee and Arduino using pins 3 and 2

// Button (pressure switch)

const int buttonPin = 8; // The number of the pushbutton pin

int buttonState = 0; // Variable for reading the pushbutton status

const int ledPin = 9; // the number of flashing light

void setup(){

Serial.begin(4800); // Begins hardware serial (PC<->Arduino)

zigBee.begin(4800); // Begins software serial (PC<->Arduino)

pinMode(buttonPin, INPUT); // Sets pushbutton pin as an input

pinMode(ledPin, OUTPUT); // Sets led pin as an output

}

void loop(){

buttonState = digitalRead(buttonPin); // Reads the state of the pushbutton value

if (buttonState == HIGH){ // If button is pressed do following

digitalWrite(ledPin, HIGH); // Sets the LED on

zigBee.print('3'); // Sends 3 over zigBee connection

Serial.println("vase has moved"); // Prints out in serial monitor

delay(1000); // Waits for one second

digitalWrite(ledPin, LOW); // Sets the LED off

delay(1000); // Waits for one second

digitalWrite(ledPin, HIGH); // Sets the LED on

delay(1000); // Waits for one second

digitalWrite(ledPin, LOW); // Sets the LED off

delay(1000); // Waits for one second

digitalWrite(ledPin, HIGH); // Sets the LED on

delay(1000); // Waits for one second

digitalWrite(ledPin, LOW); // Sets the LED off

delay(1000); // Waits for one second

digitalWrite(ledPin, HIGH); // Sets the LED on

delay(1000); // Waits for one second

digitalWrite(ledPin, LOW); // Sets the LED off

delay(1000);} // Waits for one second

else{}

}

Reference List:

Severance, C. (2014). Massimo Banzi: Building Arduino. *Computer*. [Online] **47**(1), pp.11 - 12 [Accessed 21 October 2014] Available at: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6750433>

Teikari, P., Najjar, R. P., Malkki, H., Knoblauch, K., Dumortier, D., Gronfier, C., & Cooper, H. M. (2012). An inexpensive Arduino-based LED stimulator system for vision research. *Journal of Neuroscience Methods*. [Online] **211**(2), pp.227 - 36 [Accessed 30 October 2014] Available at: <http://www.sciencedirect.com/science/article/pii/S0165027012003846>

Agudo, J. E., Pardo, P. J., Sánchez, H., Pérez, Á. L., & Suero, M. I. (2014). A low-cost real colour picker based on Arduino. *Sensors (Basel, Switzerland).* [Online] **14**(7), pp.11943 - 11956 [Accessed 27 October 2014] Available at: <http://www.mdpi.com/1424-8220/14/7/11943/htm>

Luiz, I., Moura, B. De, Carlos, L., Monteiro, D. S., & Soares, F. A. (2014). Low Cost Surface Electromyographic Signal Amplifier Based On Arduino Microcontroller. *International Journal of Electrical, Robotics, Electronics and Communications Engineering*. [Online] **8**(2), pp.310 - 314 [Accessed 17 October 2014] Available at: <http://www.waset.org/publications/9997556>