Monitoring and Control of Cardiovascular responses by using portable devices

By

Hamzah Alqudah

Submitted in Partial Fulfillment of the Requirements

for the Degree of Master by research

in the Faculty of Engineering and Information Technology at

University of Technology Sydney, 2015

Sydney, Australia

Certificate of Original Authorship

I, Hamzah Alqudah, certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis, itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signature of Student:

Date:

Acknowledgments

I would like to extend my deepest gratitude to all who supported me during the last academic years and throughout the delivery of my thesis.

To begin with, I would like to extend a big thank you to Dr. Li Li and Dr. Steven Su for their ongoing help, support, advices, friendship and knowledge which helped me a lot to develop and achieve great success on both academic and personal levels.

I am most grateful to my colleagues Azzam Haddad for his kindness and assistance. I am indebted to the Centre for Health Technologies (CHT) - Faculty of Engineering and Information Technology-University of Technology Sydney to provide me with all necessary tools to accomplish my thesis.

Words fail me to express my appreciation and love to my late mother who believed in me and carried me till her last breath. I dedicate this work to her, may her soul rest in peace.

Finally, warm thanks you to my father, brothers, sister and my wife who has supported and helped me a lot during my study.

Table of Contents

List of Figures:

List of Tables:

List of Equations:

ABSTRACT

Interval training is an effective training protocol which helps strengthening and improving the athletes cardiovascular.

Heart rate (HR) and oxygen uptake (VO_2) are major indicators of human cardiovascular response to exercises, observing these two factors can help predict energy expenditure (EE) which is an important factor in improving cardiovascular health. HR and VO_2 measurements can also aid early detection of cardiac diseases. The measurements of oxygen uptake and heart rate during sport or life activities are of great interest for development of training programs and the study of their effects on elite athletes or for assessing the efficacy of a rehabilitation therapy.

A common method for evaluating the effects of endurance training is the monitoring of various respiratory parameters during exercise. One difficulty to achieve this goal during sport or different life activities is to use a reliable and valid portable system to measure the HR in a field setting. Such a portable apparatus may also be useful to determine the energy cost of many sport and real life activities.

In this thesis, a portable device from Texas Instruments has been used to measure the Heart Rate. The eZ430-Chronos watch was reprogrammed and customized to measure the heart rate and to respond accordingly to eliminate any risk while exercising and to develop the exerciser cardiovascular fitness.

Chapter 1

Introduction

Exercising plays a vital role in improving and protecting the human being health; many diseases can be avoided by just exercising regularly, those diseases such as cancer, blood pressure, heart problems and diabetes.

The training protocol consists of three major phases; that is a warm up, exercise and cools down.

Warm up prepares the body for more intense exercise by improving blood flow to the heart, increasing the muscle temperature and protecting against injury through improved flexibility of muscles [1]. Warm-up techniques can be broadly classified into two major categories: (i) passive warm up, or (ii) active warm up. Passive warm-up involves raising muscle temperature or core temperature by some external means (e.g. hot showers or baths, saunas, diathermy and heating pads). Active warm up involves exercise and is likely to induce greater metabolic and cardiovascular changes than passive warm up [2].

The active warm up is probably the most widely used warm up technique and increases overall body temperature through active movements of the major muscle groups [3]. Warm up reduces the risk of injury. In this study, the active warm up was used to increase the muscles temperature and to avoid the risk on injuries, the participants were asked to do an active warm up by walking slowly for five minutes prior to the main exercise.

The second phase in training protocol is the exercising; the main characteristics of this phase include the intensity, duration, frequency and mode of exercise [4].

Exercise intensity refers to how much energy is expended when exercising, and the heart rate is used to measure the intensity of any exercise.

HR_{max} or the Maximum heart rate is defined as the highest number of beats per minute (bpm) during maximum exertion. There are many formulas to calculate the HR_{max}; the easiest and most common method is by using this formula:

$$\boldsymbol{HR_{max} = 220 - Age} \quad \text{Equation 1}$$

Inbar [5] has suggested a general equation to estimate the HR_{max}:

$$\boldsymbol{HR_{max} = 205.8 - 0.685\ x\ (age)} \quad \text{Equation 2}$$

The HR_{max} is used to determine the exercise intensity, which is measured by finding the HR_{max} percentage during the exercise or training. The American Heart Association recommends that an inactive person exercise in ten to fifteen minute sessions with a lower target zone of 50% and an active person in at least thirty minute sessions with an upper target zone of 85%, While [6] has recommended an 80% of the maximum heart rate for unfit people and for those with respiratory or cardiac risks.

Cooling down is the last phase of the training protocol. It is a transition from exercise to resting. The exercise intensity at this stage has been proposed to match 40-50 % of maximum heart rate of the exerciser, which will help in decreasing the heart rate back to its resting level.

Cooling down can also be defined as the phase that bring the body back to its normal physiological level after fast, vigorous exercise or activity by gradually slowing the pace of activity or by doing gentle exercises or stretches [7].

1.1 The objectives of the study.

The primary objective of this study is to use a portable device to monitor and control the cardiovascular responses, and mainly to control the heart rate using a portable device, the eZ430-chronos watch from Texas Instruments has been chosen as a primary device.

The first step to achieve the target is by investigating the HR response while walking or running exercises.

Then we need to customize and reprogram the eZ430-Chronos watch to respond to different heart rate values during the exercise.

1.2 The methodology of the research

To meet the above mentioned objectives, the research efforts have examined the following:

1. A comprehensive literature review to identify the cardiovascular system fitness concept.

2. Study the benefits of interval training protocol.

3. A comprehensive study of eZ430-Chronos watch from TI to understand its functionality, programming and reprogramming using CCS software.

4. The eZ430-Chronos was programmed and customized to respond to different heart rate situations.

5. The eZ430-Chronos was tested on many volunteers in different types of exercises.

6. The study propose that this watch can be used to control the exercise machine such as the treadmill based on the requested heart rate values which can help in rehabilitation the patients.

1.3 Outline of the thesis.

This section gives an overview of the topics in this study and their relationship to each other. In chapter 2, we have given an overview of the human being cardiovascular system, what is it? Its parts and explained the heart rate, how to measure it, and the importance of heart rate.

In chapter 3, we have explained in details the interval training protocol, its definition, importance and categories. In chapter 4, we have given a detailed description of the eZ430-Chronos watch, its features, and the code and chest strap.

In chapter 5, a detailed description of the code composer studio along with the C++ language has been given. While, in chapter 6, we have explained the experiments using the watch both in outdoor and indoor environment. Chapter 7 discussed the conclusion and the future work and the wide possibilities of using the eZ430-Chronos watch.

The appendix is the eZ430-Chrnos watch software, which have been reprogrammed and customized to implement the proposed control system.

1.4 Thesis contributions.

This thesis aims to develop a new device and its associated software to facilitate the monitoring and regulation of the cardiorespiratory response of exerciser in free living conditions, which has the potential to supervise the exerciser to build a stronger cardiovascular fitness. The contributions of this thesis can be summarized as follows:

Firstly, this thesis proposed an innovative approach for the indirect monitoring and control of cardiorespiratory response, e.g. VO2, by using cheap and wearable HR sensors. It also developed a prototype of the proposed portable system to prove the feasibility of the proposed approach.

Secondly, by selecting a highly integrated wearable wireless watch, the EZ430-Chronos, this thesis implemented exercises monitoring in free living condition. Furthermore, to adapt the limited computation capacity of EZ430-Chronos, this thesis developed various innovative codes to implement both heart rate estimation and controller implementation

Thirdly, this thesis provided a new method for the control of exercise intensity by using audio stimulations. Specifically, the develop prototype generate “beep” alter under certain condition as audio notification for exercise rate adjustments. The audio system in the watch can supervise the exerciser to adjust the exercise intensity to reach the goal of training exercise.

The developed portable device and its associated algorithms can be used in various exercise monitoring and regulation related projects. For example, based on this portable system, a new effective interval training protocol has been proposed and implemented recently in Centre of Health Technologies, UTS.

Furthermore, the developed system can be utilized in the control of other automated exercise machines, such as treadmill or bicycle to help with the patient's rehabilitation program and exercise training.

Chapter 2

Cardiovascular fitness

2.1 Definition of cardiovascular fitness.

The cardiovascular system consists of the heart, blood vessels, and the blood. The cardiovascular fitness is defined as the ability of the heart and lungs to supply oxygen-rich blood to the working muscle tissues and the ability to use oxygen to produce energy for movement [8].

Cardiovascular fitness is measured as the amount of oxygen transported in the blood and pumped by the heart to the working muscles and as the efficiency of the muscles to use the oxygen.

The cardiorespiratory system is composed of the respiratory system and the cardiovascular system, the main function of the respiratory system (lungs) is to allow for oxygen breathing in and to exhale carbon dioxide. The cardiovascular system delivers oxygen and nutrients to the cells to help with the removal of carbon dioxide from the cells.

Cardiovascular fitness, also known as cardiorespiratory fitness is one of the most important components of physical fitness and its composed of two parts, the first one, is how efficient are the heart and lungs at delivering oxygen to the body, and second one is how efficient is the body at creating the energy that the muscles will need in order to contract.

Regular exercise can increase the cardiovascular fitness as the heart becomes more efficient at pumping oxygen-rich blood to working muscles and body tissues [9].

Increasing the cardiovascular fitness means increasing the capability of the heart and the rest of the cardiovascular system in supplying oxygen and energy to the body.

Cardiovascular fitness has many health benefits, such as decreasing the risk of cardiovascular diseases, blood pressure, stroke, and diabetes; also it lowers the cholesterol level in the blood, and increases the bone density.

2.2 Cardiovascular system structure.

The cardiovascular system consists of the heart, blood vessels, and the blood.

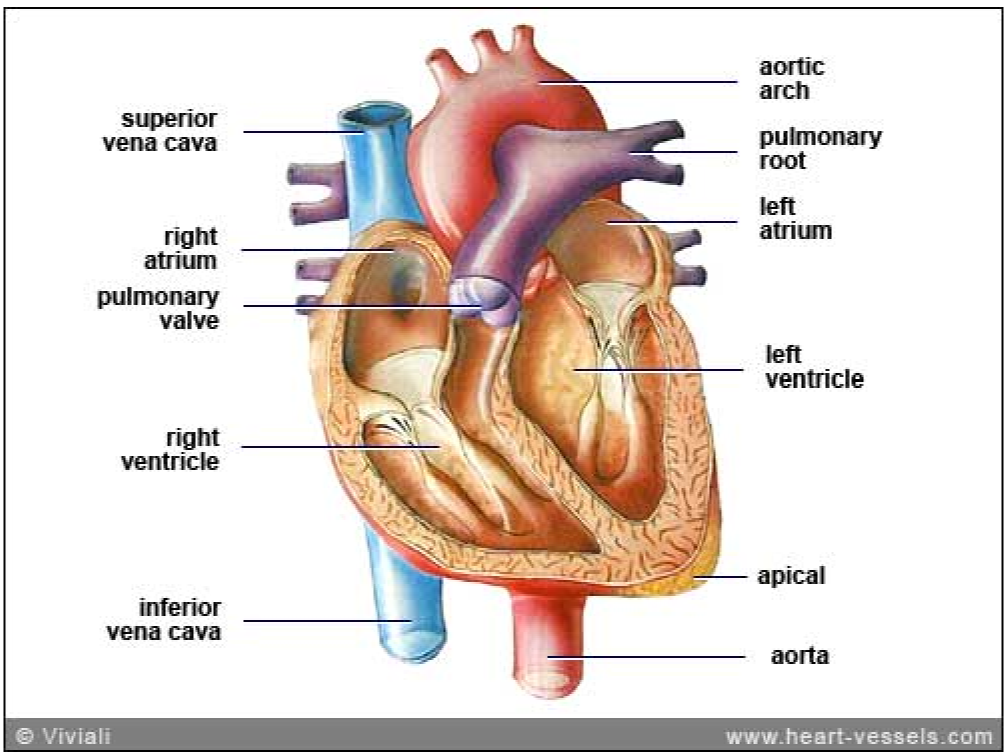


Figure 1: Heart and vessels

Heart is defined by [10] as a hollow, muscular organ that pumps blood through the body of a vertebrate animal by contracting and relaxing. In humans and other mammals, it has four chambers, consisting of two atria and two ventricles. The right side of the heart collects blood with low oxygen levels from the veins and pumps it to the lungs. The left side receives blood with high oxygen levels from the lungs and pumps it into the aorta, which carries it to the arteries of the body. The heart in other vertebrates functions similarly but often has fewer chambers.

Blood vessels is an elastic tube or passage in the body through which blood circulates; an artery, a vein, or a capillary [11].

Blood is a fluid consisting of plasma, blood cells, and platelets that is circulated by the heart through the vertebrate vascular system, carrying oxygen and nutrients to and waste materials away from all body tissues. [12]

2.3 Cardiovascular system health.

Cardiovascular health refers to the health of heart as well as the vessels, so it's very important to maintain the health of the heart by doing exercises regularly.

Heart rate gives an indication of the exercise intensity, which can be measured by finding out the percentage of the HR_{max}. The intensity of the exercise refers to the body activity during any physical activity. There are three levels of the exercise intensity, which are low, moderate or vigorous.

A lower resting heart rate indicates greater cardiovascular health. The typical resting heart rate in adults is 60-80 bpm [13], while a highly-conditioned athlete may have a resting heart rate of 40-60 beats per minute.

To maintain the health of the cardiovascular system, the cardiovascular fitness must be increased, which means increasing the capability of the heart and the rest of the cardiovascular system in supplying oxygen and energy to body. This can be achieved by doing different type of activities such as walking, running, cycling, stair climbing and swimming.

It's very important for anyone to have a good cardiovascular fitness, which will help in having a very healthy body free of many diseases such as diabetes, high blood pressure and cardiovascular diseases.

The criterion measure of cardio-respiratory fitness is maximal oxygen uptake ($VO2_{max}$) typically expressed in liters of O_2 consumed per minute ($L \cdot min^{-1}$) or milliliters of O_2 consumed per kilogram of body mass per minute ($mL \cdot kg{-}1 \cdot min^{-1}$). Accurate measurement of VO_{2max} usually requires expensive testing systems, treadmills or cycle ergometers, suitably trained personnel; and, maximal effort from the subject [14].

A wide variety of predictive sub-maximal exercise protocols are available for use; these include treadmill walking [15, 16] and cycle ergometry [17] tests. However, motorized treadmills and cycle ergometers are not always available in a clinical setting. In contrast, step tests that require limited equipment (i.e., step, metronome, heart rate monitor, and stop watch) represent an attractive modality for assessing cardio-respiratory fitness in clinic.

2.4 Heart Rate.

Heart Rate (HR) simply refers to the number of the heart beats per minute and its expresses as beats per minute (bpm).

The heart rate varies from one to one, and there are many factors which effect on the heart rate such as, the sex, age, fitness and the physical needs.

The average heart rate for adults is between 60 bpm to 100 bpm, and it's considered to be high when its more than 100 bpm and low when it's below 60 bpm.

Heart rate can be measured manually or by using heart monitor devices. The simplest method to measure the heart rate manually is by placing two fingers and checks the pulse, by counting the number of pulses or beats in 15 seconds and multiplying the number by 4, the beats per minutes can be calculated.

At the start of the 20th century, the Dutch physiologist Willem Einthoven developed the first electrocardiograph (ECG). With an ECG it is possible to make a graphic recording of the electric activity, which is present in the heart. The ECG is composed of three sections, a P wave, a QRS wave and a T wave.

The Holter-monitor was developed after the invention of the ECG; it is a portable ECG capable of making a continuous tape recording of an individual's ECG for 24 hours. [18]

In the 1980s, the first wireless HR monitoring device was developed, consisting of a transmitter and a receiver. The transmitter could be attached to the chest, and the receiver was a watch-like monitor worn on the wrist [19]. One example of such devices is the eZ430-Chronos watch from TI with its chest strap; this device has been used in this study to implement the interval training protocol.

The portable heart rate monitors which are widely used nowadays are considered to be very easy to use, cheap, and reliable. The good thing of those devices, that the athletes can adjust the exercise intensity based on the recommended heart rate, as we will discuss later in this study, how using a portable device such as the eZ430-Chronos watch will enable the exerciser to improve his cardiovascular fitness based on the recommended heart rate by using this watch.

Another fact to consider when using those devices, that they are very reliable, and so many studies have been done to compare the results of those devices with the traditional ones, and the result were same.

In our study, we have compared the results of the eZ430-Chronos watch with the results from the cosmed K4b2 and the results were identical, this result encourages us to use the eZ430-Chronos watch as a primary device to measure the heart rate either in outdoor or indoor environment.

There are many factors, which might effect on the heart rate such as age, fitness, gender and temperature.

Temperature is considered an external factor that effect on the heart rate, the heart rate in a high temperature environment will be higher than at low temperature environment of the same exercise intensity.

Heart rate is used to find out the exercise intensity, which is defined as the amount of energy in kilojoules per minute to perform a certain task (KJ/min), a study by [20] has classified the physical activity intensity based on heart rate reserve ($HR_{reserve}$) and maximum heart rate (HR_{max}) to express intensity. $HR_{reserve}$ is defined as the difference between the HR_{max} and the HR_{rest}:

$$\boldsymbol{HR_{reserve} = HR_{max} - HR_{rest}} \qquad \text{Equation 3}$$

HR_{max} or the Maximum heart rate is defined as the highest number of beats per minute (bpm) during maximum exertion. The equation of Inbar [5] is considered to be the most accurate and widely used to calculate the maximum heart rate:

$$HR_{max} = 205.8 - 0.685\ x\ (age)$$

HR_{rest} is the value of the HR while resting or not doing any exercise, the best time to find out the HR_{rest} is at wake up time. The lower the HR_{rest} is the more and great fitness health level. HR_{rest} can go as low as 30 bpm. One of the great athletes Miguel Indurain who is a five time winner of the Tour de France reported to have a resting heart rate of 28 bpm.

2.5 Conclusion.

The cardiovascular fitness is defined as the ability of the heart and lungs to supply oxygen-rich blood to the working muscle tissues and the ability to use oxygen to produce energy for movement.

Cardiovascular fitness is measured as the amount of oxygen transported in the blood and pumped by the heart to the working muscles and as the efficiency of the muscles to use the oxygen.

Regular exercise can increase the cardiovascular fitness as the heart becomes more efficient at pumping oxygen-rich blood to working muscles and body tissues. Increasing the cardiovascular fitness means increasing the capability of the heart and the rest of the cardiovascular system in supplying oxygen and energy to the body.

Cardiovascular fitness has many health benefits, such as decreasing the risk of cardiovascular diseases, blood pressure, stroke, and diabetes; also it lowers the cholesterol level in the blood, and increases the bone density.

The average heart rate for adults is between 60 bpm to 100 bpm, and it’s considered to be high when its more than 100 bpm and low when it’s below 60 bpm. There are many factors, which might effect on the heart rate such as age, fitness, gender and temperature.

Heart rate gives an indication of the exercise intensity, which can be measured by finding out the percentage of the HR_{max}. The intensity of the exercise refers to the body activity during any physical activity. There are three levels of the exercise intensity, which are low, moderate or vigorous.

To maintain the health of the cardiovascular system, the cardiovascular fitness must be increased, which means increasing the capability of the heart and the rest of the cardiovascular system in supplying oxygen and energy to body. This can be achieved by doing different type of activities such as walking, running, cycling, stair climbing and swimming.

It's very important for anyone to have a good cardiovascular fitness, which will help in having a very healthy body free of many diseases such as diabetes, high blood pressure and cardiovascular diseases.

The portable heart rate monitors which are widely used nowadays are considered to be very easy to use, cheap, and reliable. The good thing of those devices, that the athletes can adjust the exercise intensity based on the recommended heart rate.

Chapter 3

Interval Training Protocol

3.1 Definition of Interval Training Protocol.

Interval training is a workout in which you alternate periods of high-intensity exercise with low-intensity recovery periods. Also it is defined as a training protocol that requires participants to increase and decrease the workout intensity between exercise sessions and recovery periods. Interval training is a well-known exercise protocol that has been shown to strengthen and improve cardiovascular health. Moreover, it has been shown to help with weight loss, rehabilitation, general fitness, and the reduction of heart and pulmonary diseases. It utilizes the body's energy production system by activating both aerobic and anaerobic energy sources [21].

In interval training protocol the athletes will keep switching between high intensity exercise level and rest periods. In this way, the exerciser will be able to perform more exercises and hence achieve better results comparing to the continuous exercise.

Any training or workout such is running, swimming or cycling that involves high intensity training session with resting periods is called interval training protocol. The interval training protocol has proven to build up and strengthening the athletes cardiovascular system. We can usually notice that long distance runners are performing the interval training protocol as well as footballers.

3.2 Importance of interval training protocol.

Interval training has been found to be more effective than continuous training in stimulating fatty acid oxidation in muscle mitochondria [22].

Interval training is very important in decreasing the percentage of the body fat, by performing the workout for a specific time with a number of repetitions and having a rest or recovery periods the body will develop and strengthening the cardiovascular system and also will help in building stronger body.

Interval training protocol is considered nowadays as an integral part of the athletes training program specifically the runners.

The interval training protocol utilizes the body's both energy producing system; the aerobic and anaerobic. The basic difference between the aerobic and anaerobic system is that the aerobic system uses the oxygen to convert carbohydrates into energy, while the anaerobic system gets the energy from the stored carbohydrates in the muscles. The aerobic system is utilized while training for several minutes, on the other hand, the anaerobic system is utilized in a short training such as weight lifting.

The most basic form of the interval training is that a workout, in which the exerciser will walk on speed 5 km/h for two minutes and then run on speed 8 km/h for two minutes and continue in this way during the whole workout. However the exerciser can choose between the workout intensity based on his fitness and the goals to be achieved.

It has been proven throughout many studies and experiments that regular exercising plays a very important role in protecting the human beings from many diseases such as diabetes and heart problems, in addition to the positive moral of the exerciser in almost free of any cost.

Interval training protocol has been shown to improve cardiorespiratory fitness in a range of populations including those with coronary artery disease, congestive heart failure, middle age adults with metabolic syndrome and obese individuals [23][24][25][26], In many cases, the increase in cardiorespiratory fitness after interval training protocol was superior to after continuous moderate-intensity training [24][26].

Endothelial function, assessed using flow-mediated dilatation of the brachial artery, is improved to a greater extent following the interval training protocol compared with continuous moderate-intensity training [24][26], Other studies have documented beneficial changes in various components of resting blood pressure [27][28][29].

With each work interval, fatigue is caused by the depletion of stored fuel in the muscles, the accumulation of lactic acid in the muscles and blood, or the inadequate supply of oxygen to the active muscles. The repetition of the work intervals causes the onset of fatigue many times during a single workout. This type of training, therefore, improves the resistance to fatigue of the active muscles by repeatedly exposing them to high intensity exercise [30]. Which in turn, will help the exerciser to sustain a given exercise intensity for a longer period of time, increasing his or her endurance.

3.3 Interval training protocol categories.

As discussed earlier that the heart rate increases as the intensity of the exercise increases, and that the heart rate is the most common method to measure the exercise intensity, so to find out the exercise intensity, we need to calculate the target heart rate, which is a percentage of the maximum heart rate (HR_{max}).

The relationship between the heart rate and the exercise intensity is a linear relationship, where the heart rate will increases as the exercise intensity increases, and the interval training protocol has utilized this relationship to enhance and strengthening the fitness levels of the exerciser by either increasing or decreasing the heart rate.

There are four factors which effect on any interval training protocol, by manipulating those factors, different results can be achieved:

- Workout distance or duration.
- Workout intensity.
- Rest periods type and duration.
- Number of repetition for each workout.

There are three categories of the interval training protocol:

Short Interval:

Repeatedly raising and lowering the heart rate during training increases the amount of blood pumped during each heartbeat, which means more oxygen is delivered to your working muscles, according to [31].

In short interval, the exerciser switches between the high intensity exercise and rest period every ten seconds which will get the muscles fibers which in turn will increase the exerciser strength and speed.

The short interval is very useful for those activities with short bursts followed by walking or jogging. The training to rest ratio is 1:6 for beginners and 1:3 for advanced individuals.

Intermediate Interval:

As we have discussed earlier in this chapter that the interval training protocol utilizes the aerobic and anaerobic energy system. In intermediate interval training protocol where the high intensity exercise duration starts from 30 seconds up to two minutes, the anaerobic energy system is targeted, which might cause muscle fatigue and discomfort.

Training within this level allows the body to be able to withstand the effects of lactic acid buildup. This can also carry over into everyday activities like carrying heavy bags of groceries or dragging heavy moving boxes from one place to another. The biggest benefits you will gain from this type of training are fast twitch muscle development and improving your resistance to muscle fatigue. In this type, the Work to Rest Ratio is 1:2 for all individuals.

Long Interval:

This type of interval training incorporates longer work periods and targets the Aerobic energy system. This energy system utilizes carbohydrates and fats to fuel the body. During high intensity aerobic exercise, the majority of the energy comes from carbohydrates. Steady, low intensity work mainly uses fats and protein. Continuous running for longer than 2 minutes increases your endurance and also improves the body's ability to transport and use oxygen. Some cardiovascular benefits of aerobic training are decreased risk for heart disease, high blood pressure and cholesterol, and it increases the heart's ability to pump blood and oxygen throughout the body, according to [31]. The focus of this type of training is to run at a pace that is a specific percentage of your maximum heart rate. Depending on your fitness level, any range from 140 to 180 beats per minute should be an appropriate training stimulus. In this type the Work to Rest Ratio is 1:1 for all individuals.

3.4 Conclusion

Interval training is a workout in which you alternate periods of high-intensity exercise with low-intensity recovery periods. Also it is defined as a training protocol that requires participants to increase and decrease the workout intensity between exercise sessions and recovery periods.

In interval training protocol the athletes will keep switching between high intensity exercise level and rest periods. In this way, the exerciser will be able to perform more exercises and hence achieve better results comparing to the continuous exercise.

The interval training protocol has proven to build up and strengthening the athletes cardiovascular system. We can usually notice that long distance runners are performing the interval training protocol as well as footballers.

Interval training protocol is considered nowadays as an integral part of the athletes training program specifically the runners.

The most basic form of the interval training is that a workout, in which the exerciser will walk on speed 5 km/h for two minutes and then run on speed 8 km/h for two minutes and continue in this way during the whole workout. However the exerciser can choose between the workout intensity based on his fitness and the goals to be achieved.

The relationship between the heart rate and the exercise intensity is a linear relationship, where the heart rate will increases as the exercise intensity increases, and the interval training protocol has utilized this relationship to enhance and strengthening the fitness levels of the exerciser by either increasing or decreasing the heart rate.

Chapter 4

eZ430-Chronos watch

4.1 Introduction.

In this chapter, the eZ430-Chronos watch which has been used in this study will be examined and described in details to cover its features, capabilities and the its software.

4.2 Description.

The eZ430-Chronos watch from Texas Instruments is a highly integrated, wearable wireless development system that provides a complete reference design for developers to create wireless smart watch applications. It may be used as a reference platform for watch systems, a personal display for personal area network or as wireless sensor node for remote data collection. Based on the CC430F6137, the eZ430-Chronos is a complete CC430-based development system, featuring a 96 segment LCD display and provides an integrated pressure sensor and 3-axis accelerometer for motion sensitive control. The integrated wireless interface allows the eZ430-Chronos to act as a central hub for nearby wireless sensors such as pedometers and heart rate monitors. The eZ430-chronos watch may be disassembled to be reprogrammed with a custom application and includes an eZ430 USB programming interface.

The watch comes with a chest strap for heart rate monitoring from BM Innovations, it sends the measured values to eZ430-Chronos watch, which in turn will run our customized software.

Figure 2: eZ430-Chronos watch

4.3 Features.

The main features of the eZ430-Chronos are:

- Fully functional sports watch based on the CC430F613, MSP430 with integrated sub-1-GHz wireless transceiver.
- Watch can be reprogrammed for custom wireless applications.
- Highly integrated watch includes on-board three-axis accelerometer, pressure sensor, temperature sensor and voltage sensor.
- 96-segment LCD display driven directly by CC430.
- Can be paired wirelessly with heart rate monitors, pedometers, or other devices based on RF transceivers such as the CC430 or CC11xx series.
- Includes an eZ430-RF USB emulator that connects the eZ430-Chronos to a PC for real-time in-system programming and debugging.

4.4 eZ430-Chronos Code.

The default firmware of the watch module provides a broad set of features, such as time, date, alarm, stopwatch, altimeter, heart-rate monitor, calorie, vertical speed, and distance information. The internal accelerometer provides acceleration data on the wrist module LCD and allows control of a PC by transferring the sensor's measurements. The module can also be used to control PowerPoint or other PC software with its buttons.

The different features are either available in the top or bottom LCD line. There are three user modes available: Modes such as Time or Date, Secondary Functions that allow activation and deactivation of features , and Set Functions that allow changing settings (for example, setting the time or date or resetting the stopwatch).

Figure 3 and figure 4 describe the watch top and bottom menu structure. One of the most important features in this watch is the Heart Rate. As the name suggest, it measures the Heart Rate, but for this function to work, a chest belt needs to be worn and connected wirelessly to the watch, once the Heart Rate feature is active and connected the chest belt, other sub features will active such as burned calories. The heart rate monitoring uses BlueRobin protocol from BM innovations for its communication. However, a heart rate can be simulated without a chest belt by the eZ430-Chronos control center.

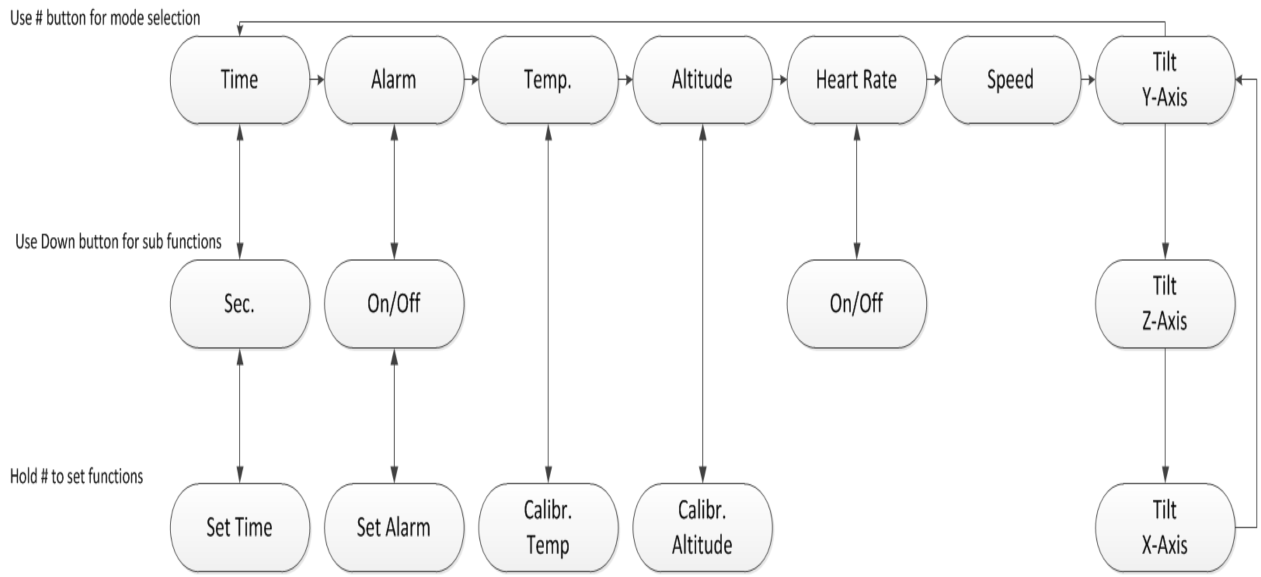


Figure 3: eZ430-Chronos Top Menu

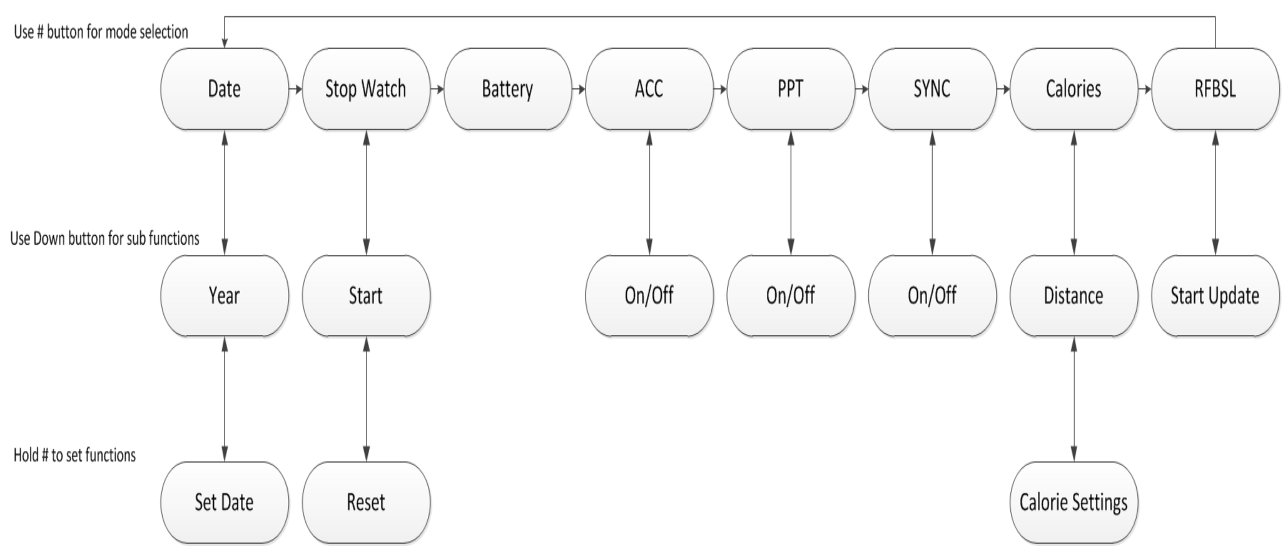


Figure 4: eZ430-Chronos Bottom Menu

The watch hardware will be initialized after a power up reset, the display memory is cleared and the radio is set to sleep mode. The main loop waits for wake up events, which can be either a scheduled event such as a clock interrupt or a button event as explained in figure 5.

If the LCD is updated with new data, the modules set a display flag, and LCD Line 1, Line 2, or both are updated by calling the menu specific display functions. If the button event is selected, then this will cause either mx_functions or sx_functions to be executed.

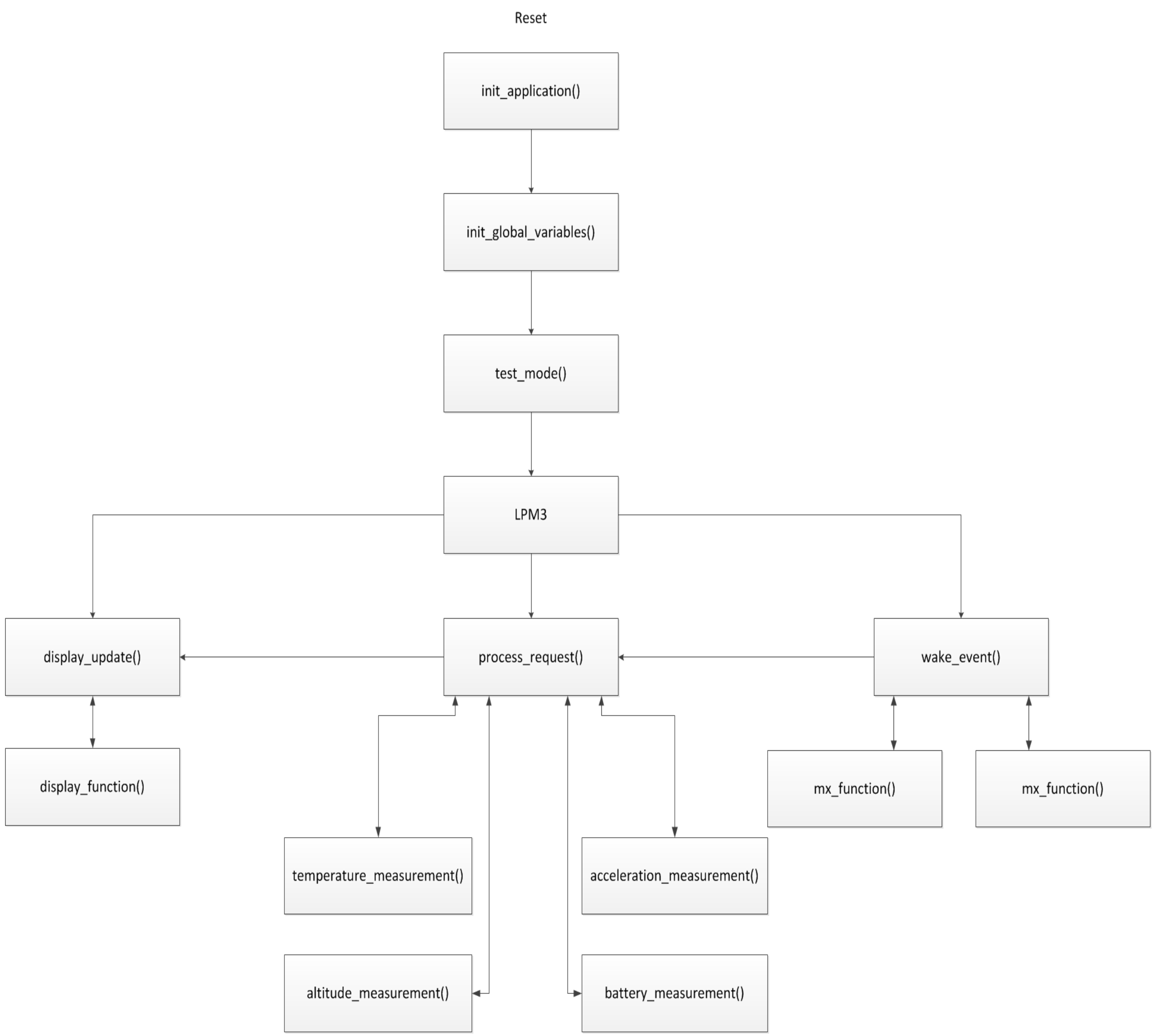


Figure 5: eZ430-Chronos software flow

4.5 eZ430-Chronos Control Center Software.

The eZ430-Chronos Control Center software provides several features demonstrating the wireless capabilities of the watch such as 3D acceleration graph with PC mouse control, wireless remote control, and wireless firmware update. The most important feature is the heart rate simulate. This feature has been used a lot while implementing the training protocol as it gives us an easy method to simulate the control system, and hence debug and update the software before testing the system in outdoor environment.

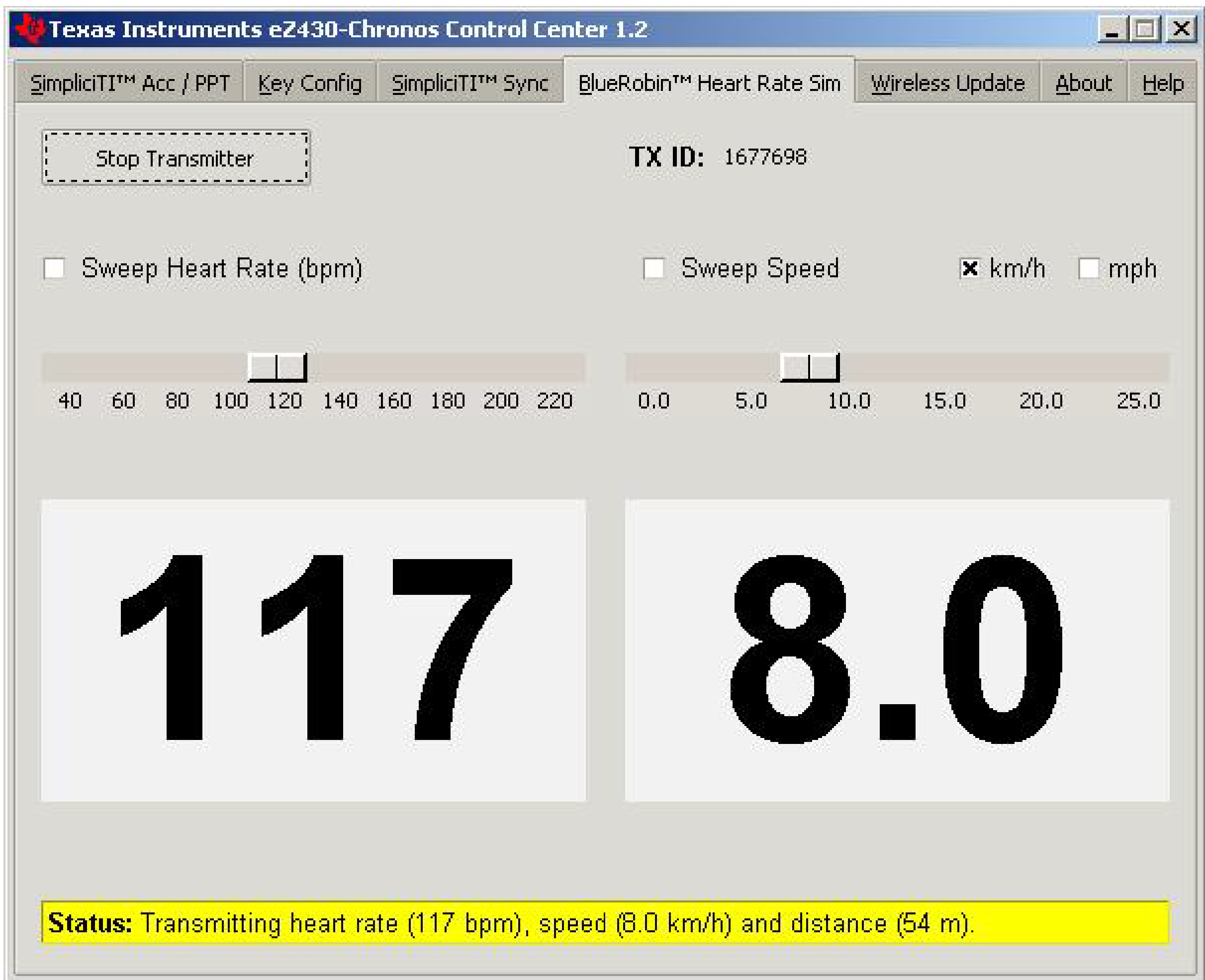


Figure 6: eZ430-Chronos Control Center Heart Rate Simulator

4.6 Chest strap.

The BM-CS5 chest strap from BM innovation has been used in this study to read the heart rate values and send them back to eZ430-Chronos watch. It uses the latest ultra-low power BlueRobin data transmission technology.

Each chest strap has a unique ID assigned that can be used to identify the user and pair the chest strap to a receiver. Built-in data collision prevention allows the chest strap to be used in multi-user environments, where a large number of chest straps transmit their data to a single receiver.

There are two models of this chest strap:

1- BM-CS5 868/915 MHz with up to 400m range.

2- BM-CS5 868/915 MHz with up to 10m range.

Figure 7: BMi Chest Strap

4.7 Conclusion.

The eZ430-Chronos watch from Texas Instruments is a highly integrated, wearable wireless development system that provides a complete reference design for developers to create wireless smart watch applications.

The watch comes with a chest strap for heart rate monitoring from BM Innovations, it sends the measured values to eZ430-Chronos watch, which in turn will run our customized software.

The eZ430-Chronos watch code was developed and customized using Code Compose Studio "CCS".

Chapter 5

Code Composer Studio

5.1 Introduction.

Code Composer Studio (CCS) v4 software is a very powerful programming tool from Texas Instruments that has been built on eclipse has been used in this study to program and customize the eZ430-Chronos watch which is based on the MSP430 microcontroller family from TI.

Code Composer Studio is an integrated development environment (IDE) that supports TI's Microcontroller and Embedded Processors portfolio. Code Composer Studio comprises a suite of tools used to develop and debug embedded applications. It includes an optimizing C/C++ compiler, source code editor, project build environment, debugger, profiler, and many other features. The intuitive IDE provides a single user interface taking you through each step of the application development flow. Familiar tools and interfaces allow users to get started faster than ever before. Code Composer Studio combines the advantages of the Eclipse software framework with advanced embedded debug capabilities from TI resulting in a compelling feature-rich development environment for embedded developers. [32]

CCS software which is based on Eclipse open source software framework was developed by TI to program and support their microcontrollers such as the MSP430 family.

5.2 Eclipse.

5.2.1 Definition of Eclipse.

Eclipse is a platform that has been designed from the ground up for building integrated web and application development tooling. By design, the platform does not provide a great deal of end user functionality by itself. The value of the platform is what it encourages: rapid development of integrated features based on a plug-in model [33].

Eclipse provides a common user interface (UI) model for working with tools. It is designed to run on multiple operating systems while providing robust integration with each underlying OS. Plug-ins can program to the Eclipse portable APIs and run unchanged on any of the supported operating systems.

At the core of Eclipse is architecture for dynamic discovery, loading, and running of plug-ins. The platform handles the logistics of finding and running the right code. The platform UI provides a standard user navigation model. Each plug-in can then focus on doing a small number of tasks well.

5.2.2 Eclipse Architecture.

The Eclipse platform defines an open architecture so that each plug-in development team can focus on their area of expertise. Let the repository experts build the back ends and the usability experts build the end user tools. If the platform is designed well, significant new features and levels of integration can be added without impact to other tools.

The Eclipse platform uses the model of a common workbench to integrate the tools from the end user's point of view. Tools that you develop can plug into the workbench using well defined hooks called extension points.

The platform itself is built in layers of plug-ins, each one defining extensions to the extension points of lower-level plug-ins, and in turn defining their own extension points for further customization. This extension model allows plug-in developers to add a variety of functionality to the basic tooling platform. The artifacts for each tool, such as files and other data, are coordinated by a common platform resource model.

The platform gives the users a common way to work with the tools, and provides integrated management of the resources they create with plug-ins.

Plug-in developers also gain from this architecture. The platform manages the complexity of different runtime environments, such as different operating systems or workgroup server environments. Plug-in developers can focus on their specific task instead of worrying about these integration issues.

5.2.3 Eclipse Structure.

The Eclipse platform is structured around the concept of plug-ins. Plug-ins are structured bundles of code and/or data that contribute functionality to the system. Functionality can be contributed in the form of code libraries (Java classes with public API), platform extensions, or even documentation. Plug-ins can define extension points, well-defined places where other plug-ins can add functionality.

Each subsystem in the platform is itself structured as a set of plug-ins that implements some key function. Some plug-ins add visible features to the platform using the extension model. Others supply class libraries that can be used to implement system extensions.

The Eclipse SDK includes the basic platform plus two major tools that are useful for plug-in development. The Java development tools (JDT) implement a full featured Java development environment. The Plug-in Developer Environment (PDE) adds specialized tools that streamline the development of plug-ins and extensions.

These tools not only serve a useful purpose, but also provide a great example of how new tools can be added to the platform by building plug-ins that extend the system.

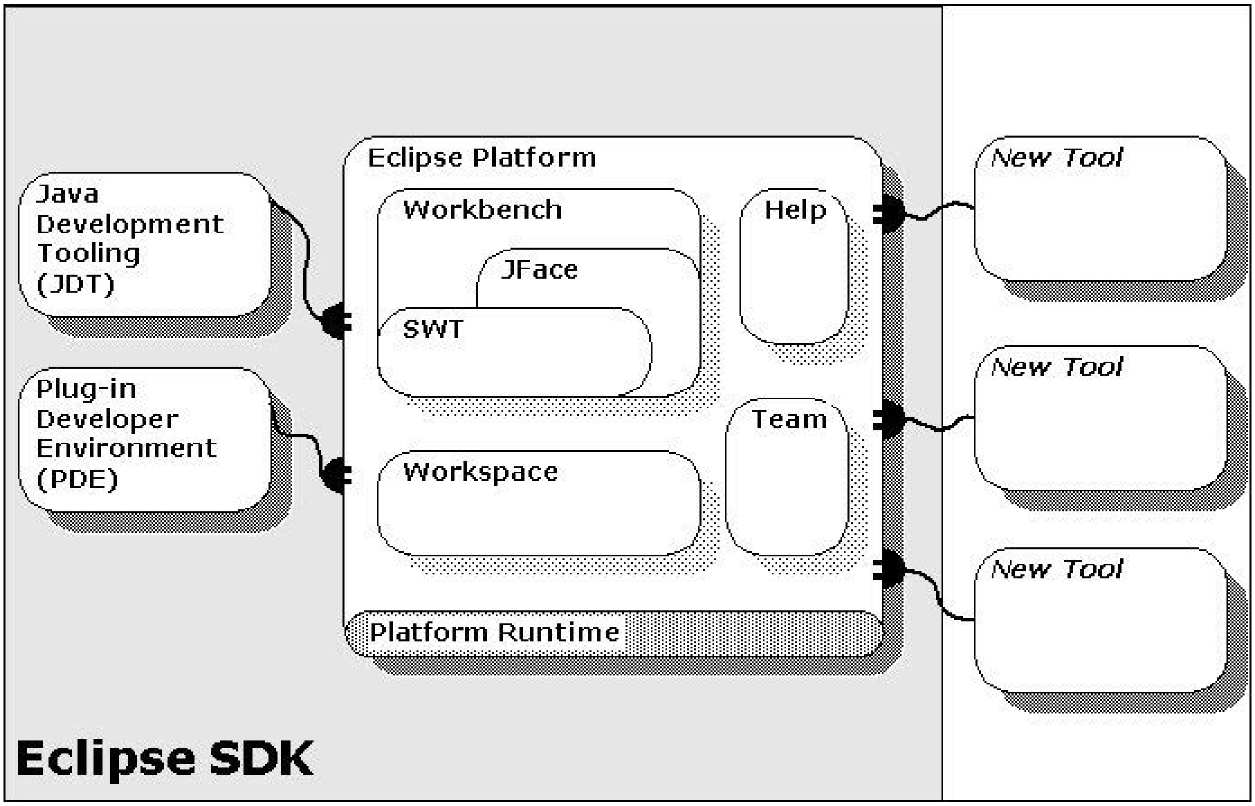


Figure 8: Eclipse SDK

5.3 C++ programming language.

In this chapter, we have explored the tools which have been used in this study, which are the eZ430-Chronos watch.

5.3.1 Introduction.

C++ is a high level language which is an extension of C language and has been developed by Bjarne Stroustrupis. C++ is a very powerful general purpose object oriented programming (OOP) language; C++ is an excellent option for microcontrollers programming.

C++ is a general purpose object oriented programming language. It is considered to be an intermediate level language, as it encapsulates both high and low level language features. Initially, the language was called 'C with classes' as it had all properties of C language with an additional concept of 'classes'. However, it was renamed to C++ in 1983.

C++ is one of the most popular languages primarily utilized with system/application software, drivers, client-server applications and embedded firmware.

The main highlight of C++ is a collection of pre-defined classes, which are data types that can be instantiated multiple times. The language also facilitates declaration of user defined classes. Classes can further accommodate member functions to implement specific functionality. Multiple objects of a particular class can be defined to implement the functions within the class. Objects can be defined as instances created at run time. These classes can also be inherited by other new classes which take in the public and protected functionalities by default.

C++ includes several operators such as comparison, arithmetic, bit manipulation, logical operators etc. One of the most attractive features of C++ is that it enables the overloading of certain operators such as addition.

A few of the essential concepts within C++ programming language include polymorphism, virtual and friend functions, templates, namespaces and pointers.

5.3.2 Features.

5.3.2.1 Object Oriented Programming (OOP):

OOP is defined as a programming paradigm that represents the concepts of "objects" that have data fields (attributes that describe the object) and associated procedures know as methods. Objects, which are usually instances of classes, are used to interact with one another to design applications and computer programs. [34][35]

The goals of object-oriented programming are:

- Increased understanding.
- Ease of maintenance.
- Ease of evolution.

5.3.2.2 Classes

A class is a user defined type or data structure that has data and functions. Its defined using keyword class followed by identifier, Body of class is defined inside curly brackets and terminated by semicolon at the end in similar way as structure.

The keyword private makes data and functions private and keyword public makes data and functions public. Private data and functions are accessible inside that class only whereas, public data and functions are accessible both inside and outside the class.

5.4 CCS Features.

The Code Composer Studio v4 main features are as following:

- Includes compilers for each of TI's device families, source code editor, project build environment, debugger, profiler, real-time operating system, and more.

- Intuitive, Eclipse-based IDE provides a single user interface that walks user step-by-step through application development flow.

- Familiar tools and interfaces help developers get started quickly, and add functionality with sophisticated productivity tools.

5.5 Programming using CCS4.

As we have mentioned earlier that the CCS4 is used as in IDE environment to customize our TI eZ430-Chronos watch, the watch was disassembled and connected through the USB programmer as shown in the following figure:

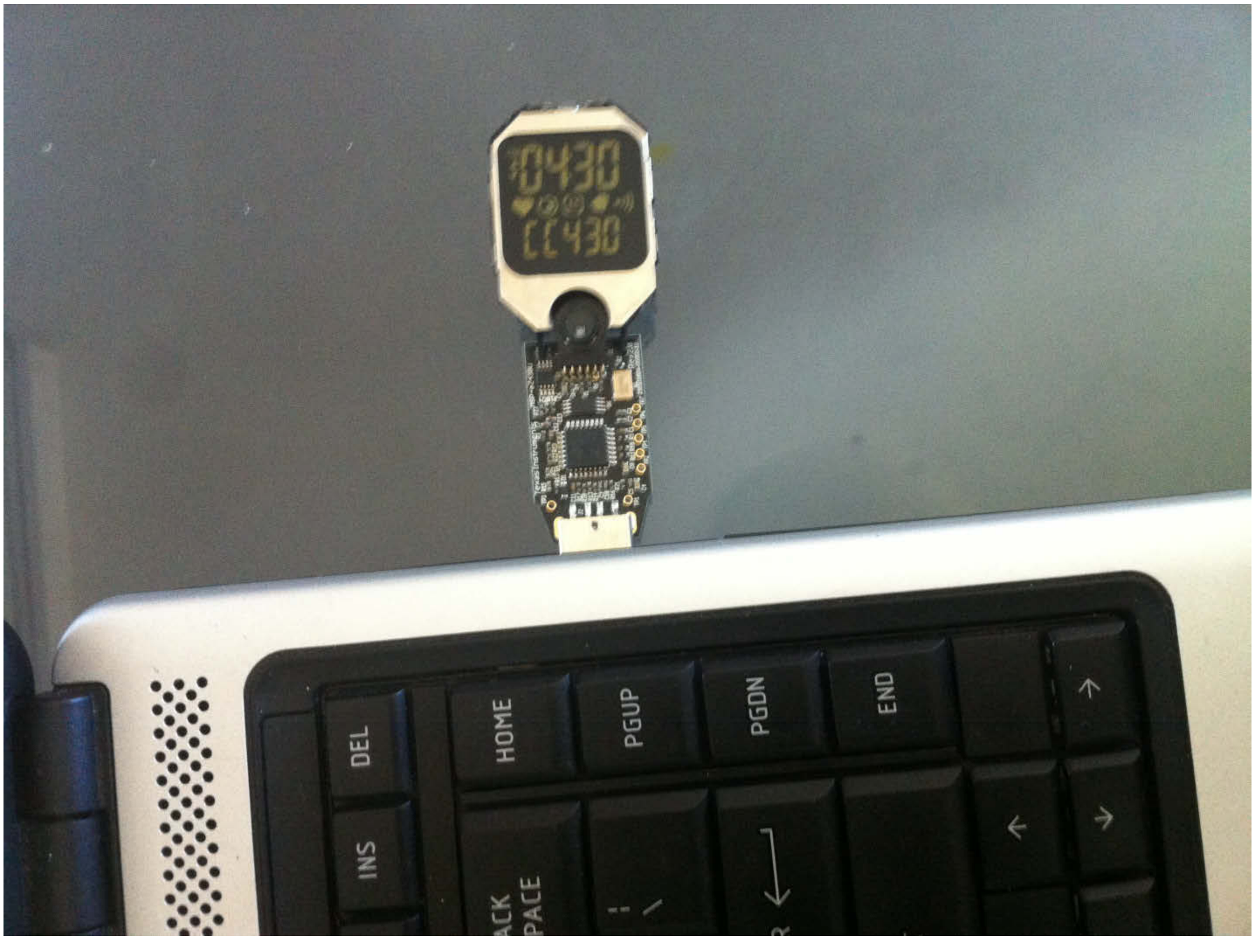


Figure 9: eZ430-Chronos connected to laptop for programming

The USB programmer drivers need to be installed on the windows environment to be able to connect it to CCS, once all this has been fixed; the TI eZ430-Chronos watch firmware was loaded into CCS for further customization as shown in the following figure:

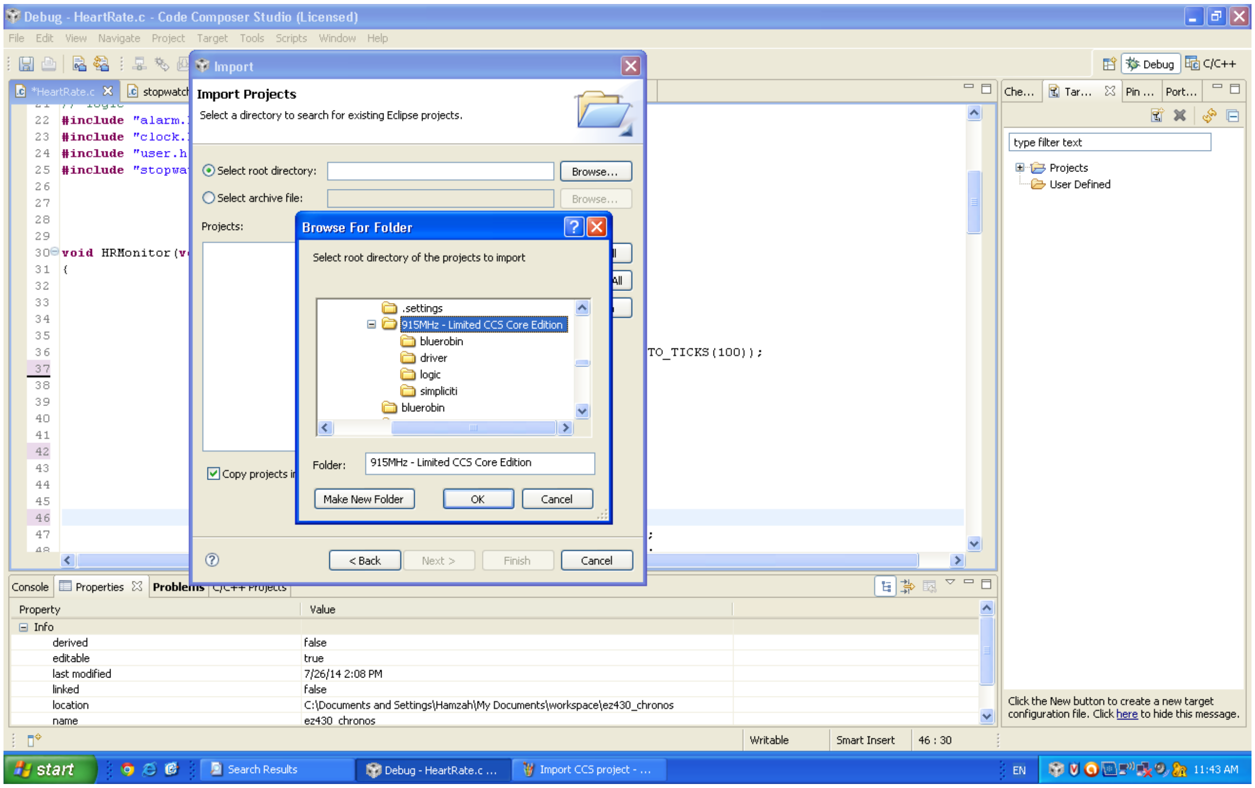


Figure 10: CCS – Import Project

The following figure shows the heart rate code in CCS which has been written in C++ language and later on were uploaded to the watch.

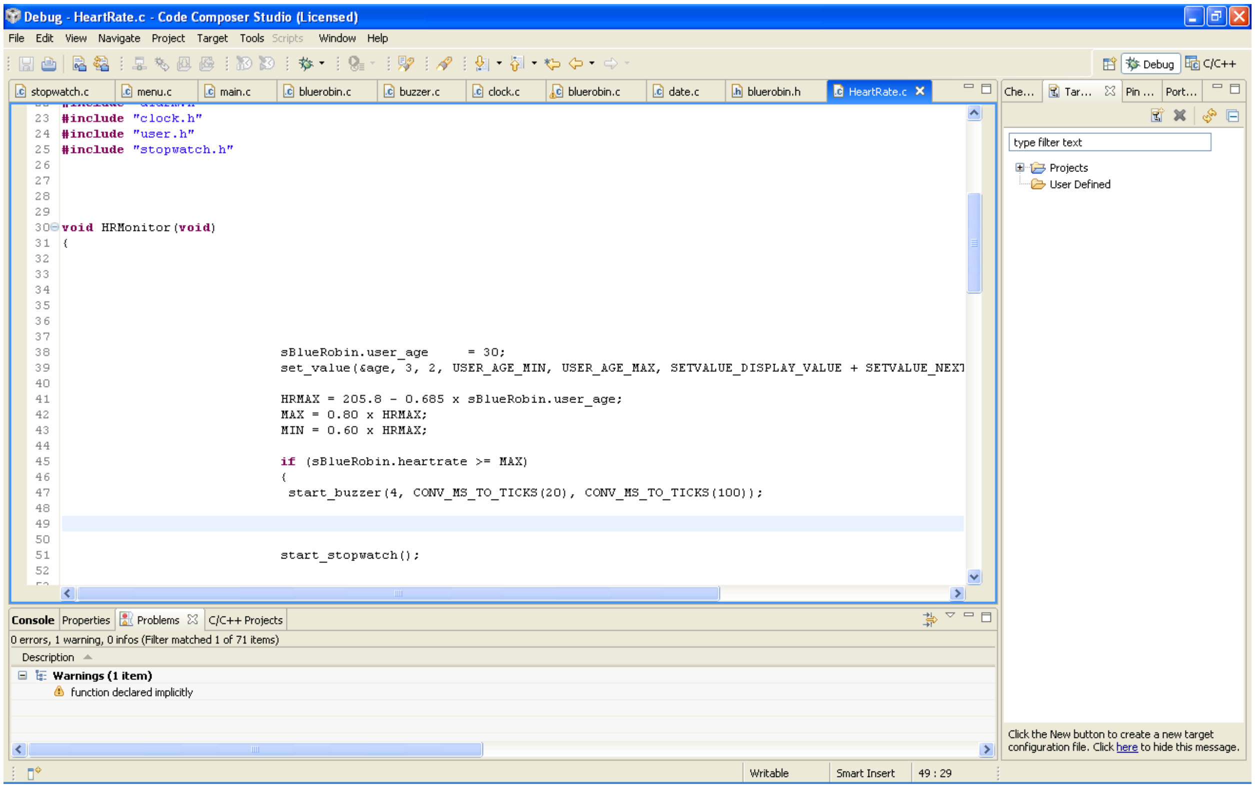


Figure 11: CCS – Heart Rate code

The whole firmware was then debugged in the CCS and uploaded to the eZ430-Chronos watch without any errors as shown in the following figure:

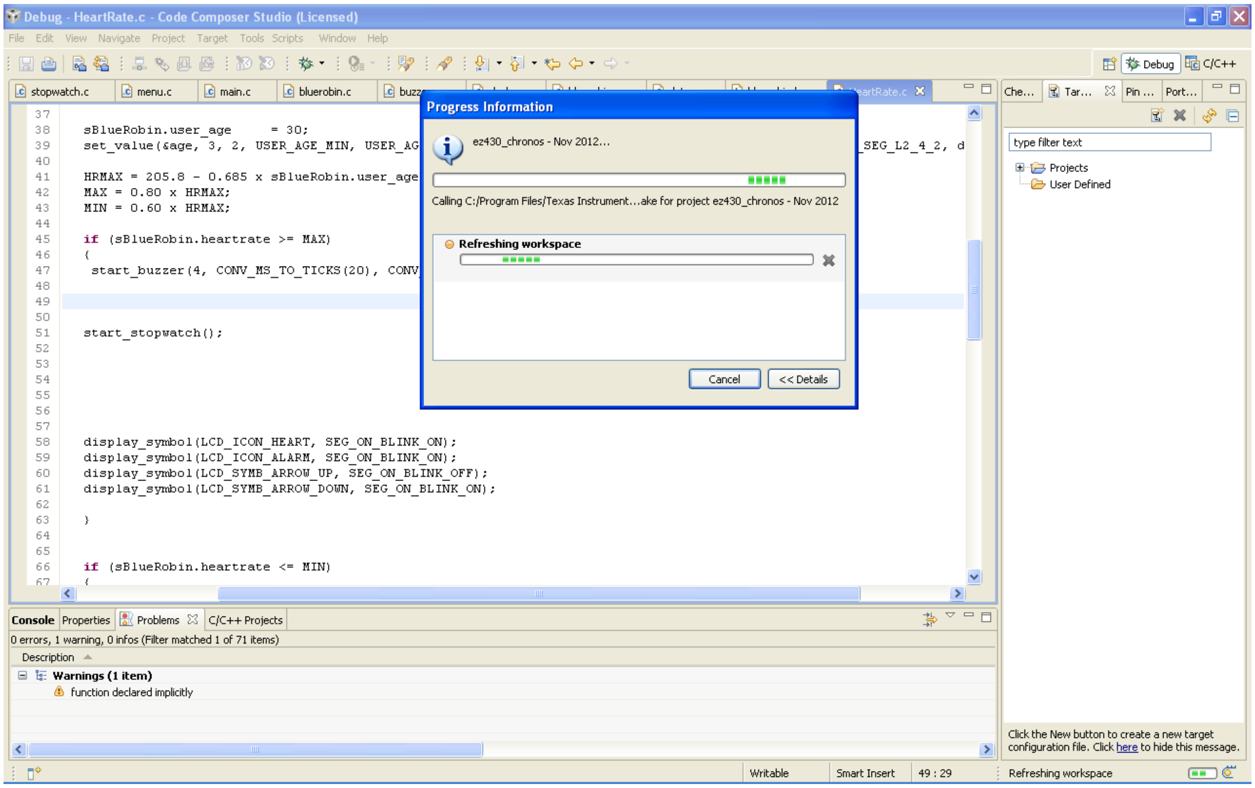


Figure 12: CCS – Debug Project

After loading the code into the eZ430-Chronos watch without any errors, the watch has been assembled again and is ready to be used by athletes for further testing and studies.

To test the watch, we need to wear the watch as well wear the chest strap, using the watch button, it will connect to the HR monitor, and the HeartRate.C will run to constantly monitor the heart rate and respond accordingly.

In the next chapter, we shall show how the watch was tested with different experiments and situations.

5.6 Conclusion

In this chapter, we have a brief description of the code composer studio which is the main programming IDE used to program the eZ430-Chronos watch.

The CCS is built on eclipse software and we have used C++ to program the eZ430-Chronos watch, also we have explored how to import projects in CCS4, how to debug them, and then how to upload the project into the watch using CCS4.

In the next chapter, we will experiment the watch with the new code on real life test both on outdoor environment and indoor environment.

Chapter 6

The Experiments and Subjects

6.1 Introduction.

The eZ430-Chronos watch is a very powerful development tool, which has integrated heart rate monitor. This watch has been used in this study to monitor the heart rate and control the athletes' behavior based on his heart rate value. We have tested the watch in indoor environment such as treadmill and in outdoor environment such as staircase, and we have noticed that the eZ430-Chronos watch is working just perfect in both conditions and it does help the athletes' to either increase or decrease the exercise intensity which will help the athletes' to increase his cardiovascular strengthening.

As we have mentioned earlier, that the interval training protocol in which the athletes alternate between the high-intensity exercise with the low-intensity exercise has proven to strengthen and improve the cardiovascular health, the athletes power and speed.

Any training or workout such is running, swimming or cycling that involves high intensity training session with resting periods is called interval training protocol. The interval training protocol has proven to build up and strengthening the athletes cardiovascular system. We can usually notice that long distance runners are performing the interval training protocol as well as footballers.

In this study, we have tested the eZ430-Chronos watch on indoor environment such as the treadmill and on outdoor environment such as the staircase.

6.2 Indoor environment experiment.

In the indoor environment, we have used the treadmill as a tool where the athletes will utilize the interval training protocol and run for a while, while wearing the eZ430-Chronos watch and the chest strap.

Figure 13: Running on treadmill while wearing the eZ430-Chronos watch.

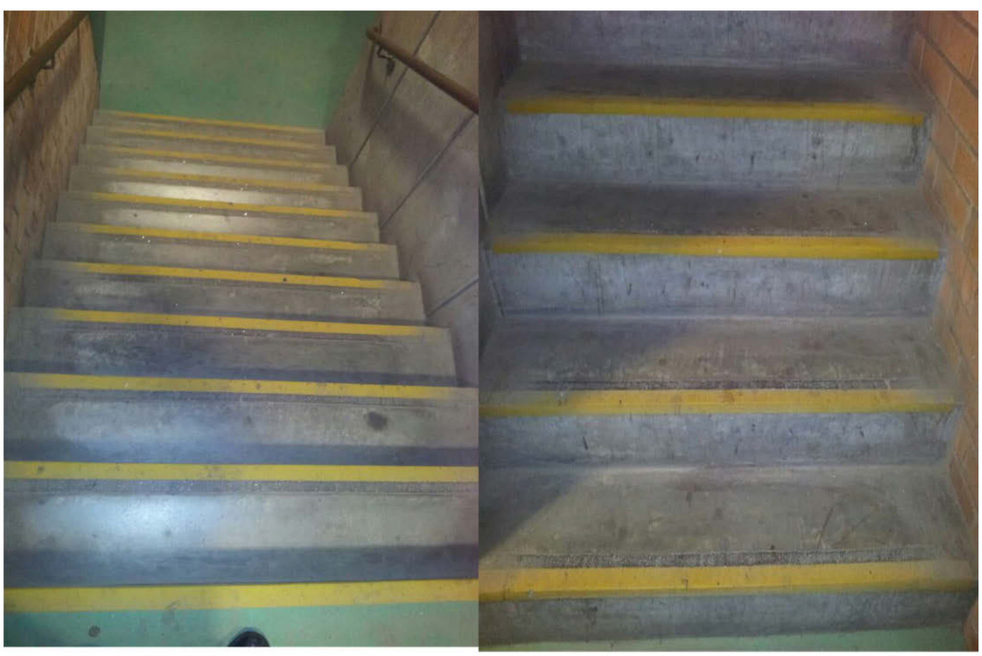
In this indoor environment, the exercisers were asked to walk on speed 5 km/h for a while, and then to start running at speed 8 km/h until they reach their HR_{max}, once they reach their HR_{max}, the watch will notify the exerciser by beeping and giving indicators on the display, so that the exerciser will slow down and start walking on speed 5 km/h until he reaches his HR_{min}.

In this way the exerciser is switching between the high density exercise which is the running on speed 8 km/h and the low density exercise which is walking on speed 5 km/h.

The experiment were performed at UTS building one emergency exit, and the participants were asked to climb the stairs from level 14 to level 26, where each level consists of two sets of stairs, with 11 stairs per set.

6.3 Outdoor environment experiment.

The UTS building one emergency exit has been chosen as an outdoor environment for the experiment, and the participants were asked to climb the stairs from level 14 to level 26, while wearing the eZ430-Chronos watch and the chest strap.



Each step is measured as 22 x 180 x 25 cm

Figure 14: UTS staircase.

In this outdoor environment, the exercisers were asked to walk on speed 5 km/h on the steps, and then to start climbing the stairs until they reach their HR_{max}, once they reach their HR_{max}, the watch will notify the exerciser by beeping and giving indicators on the display, so that the exerciser will slow down and start walking on the same step until he reaches his HR_{min}.

In this way the exerciser is switching between the high density exercise which is the climbing the stairs and the low density exercise which is walking on the steps.

The eZ430-Chronos watch requires the heart rate belt to be worn to measure the subjects' heart rate, so the exercisers were asked to wear the heart rate belt around the chest, and the eZ430 watch on their right hand.

6.4 The experiments.

6.4.1 Setup and preparation.

Five volunteers with different ages and different physical characteristics were participated in all experiments; all volunteers were free from any health issues.

The participant's physical characters are shown in the following table.

Subjects	Age (yr.)	Height (cm)	Weight (kg)
1	25	168	73
2	27	174	75
3	30	177	72
4	33	168	74
5	35	172	65

Table 1: Physical characteristics of the participants.

To guarantee an improvement of the subject's cardiovascular system fitness, the subject should participate at 80% of his HR_{max}. To find the HR_{max}, we have used Inbar formula as following:

$$HR_{max} = 205.8 - 0.685\ x\ (age)$$

$$HR_{max} = 205.8 - 0.685\ x\ (25)$$

$$= 188.7$$

$$\approx 189 \text{ bpm}$$

So, as we can see that for participate whose age is 25 years old, his HR_{max} is approx. 189 bpm, this means, that the participant must participate at 80% of his HR_{max}, which is 151 bpm.

Based on this value, the subject will continue running on treadmill or climbing the stairs until he reaches 80% of his HR_{max}.

So, while running or climbing the stairs, if the participant reaches 80% of his HR_{max}, he will start walking on the treadmill or walking on the same step he reached, and the watch will notify the participant that he reaches 80% of his HR_{max} by continuous beeping until the value drop down to less than 80% of his HR_{max}.

Using the same formula, we have calculated the HR_{max} and the HR_{min} for all participants as shown in the following table:

Subjects	**Age (yr)**	**HR_{max}**	**60% of HR_{max}**	**80% of HR_{max}**
1	25	189	113	151
2	27	187	112	150
3	30	185	111	148
4	33	183	110	147
5	35	181	109	145

Table 2: HRmax and HRmin.

6.4.2 Treadmill – Indoor environment.

The subject starts the exercise by walking for a specific time i.e. 60 seconds on speed 5 km/h, then the subject will start to run on speed 8 km/h for a specific time i.e. 60 seconds or until his heart rate reaches the maximum heart rate, once any of those two conditions is true, the subject will start walking again. The watch will beep to inform the subject, that he has reached either the minimum or the maximum heart rate value, there are two different beeps to differentiate between the maximum and minimum heart rate value, after that the subject was asked to rest on the chair for five minutes.

So, the experiment steps are as following:

- The participant wears the eZ430-Chronos watch, chest strap.
- The participant was asked to sit and rest for five minutes before doing any exercise.
- Then, the participant was asked to stand up on the treadmill, and wait for two minutes.
- The participant starts to walk on speed 5 km/h for 240 seconds.
- Then, the participant starts to run on speed 8 km/h.
- The participant will continue running until he reaches 80% of his $HR_{max.}$
- The eZ430-Chronos watch will start beeping to notify the athletes that he has reached his HR_{max}.
- Then, the participant will start to walk on speed 5 km/h until he reaches 60% of his HR_{max}.
- The eZ430-Chronos watch will start beeping to notify the athletes that he has reached 60% of his HR_{min}.
- At the end of the final walk, the subject will be seated on a chair and relax for five minutes.

The following flowchart explains the indoor experiment:

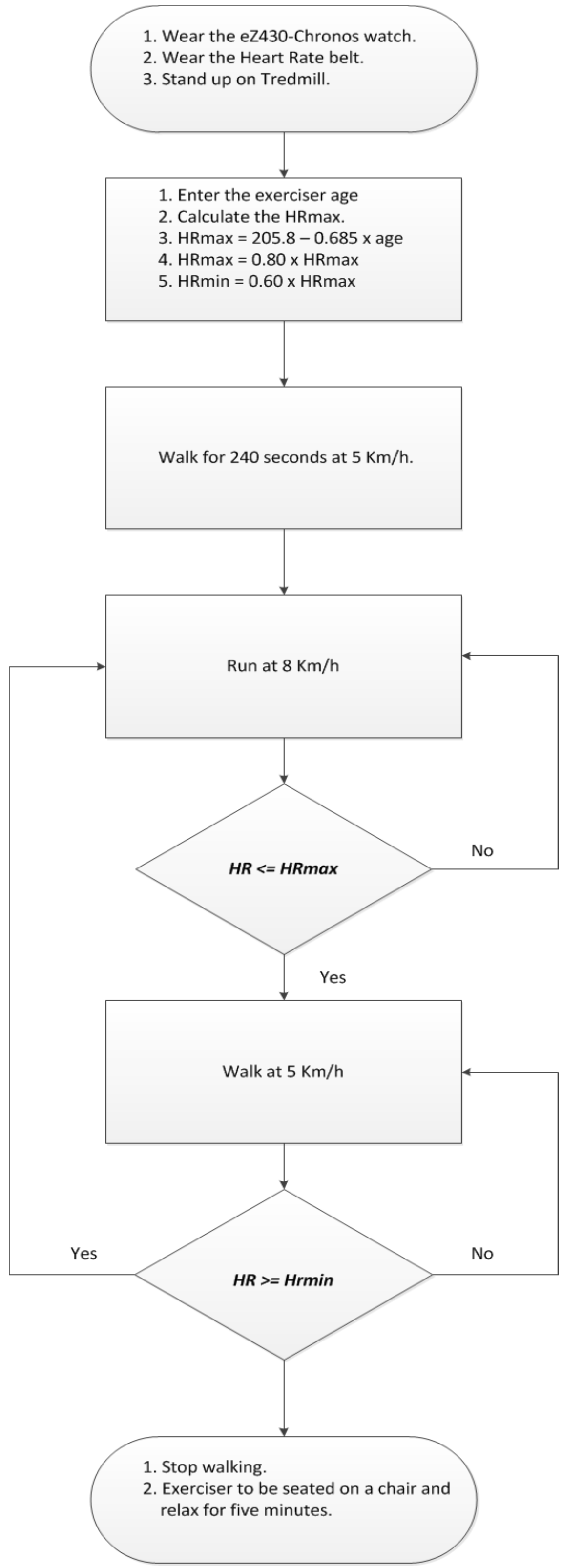


Figure 155: Indoor exercise flowchart

6.4.3 Staircase – Outdoor environment.

The subject starts the exercise by walking for a specific time i.e. 60 seconds, then the subject will start to climb the stairs for a specific time i.e. 60 seconds or until his heart rate reaches the maximum heart rate, once any of those two conditions is true, the subject will start walking again. The watch will beep to inform the subject, that he has reached either the minimum or the maximum heart rate value, there are two different beeps to differentiate between the maximum and minimum heart rate value, after that the subject was asked to rest on the chair for five minutes.

So, the experiment steps are as following:

- The participant wears the eZ430-Chronos watch, chest strap.
- The participant was asked to sit and rest for five minutes before doing any exercise.
- Then, the participant was asked to stand up next to the first set of stairs, and wait for two minutes.
- The participant starts to walk on the flat platform for 240 seconds.
- Then, the participant starts to climb the staircase.
- The participant will continue climbing until he reaches 80% of his HR_{max}.
- The eZ430-Chronos watch will start beeping to notify the athletes that he has reached 80% of his HR_{max}.
- Then, the participant will start to walk on the same step he reached until he reaches 60% of his HR_{max}.
- The eZ430-Chronos watch will start beeping to notify the athletes that he has reached 60% of his HR_{min}.
- At the end of the final walk, the subject will be seated on a chair and relax for five minutes.

The following flowchart explains the outdoor experiment:

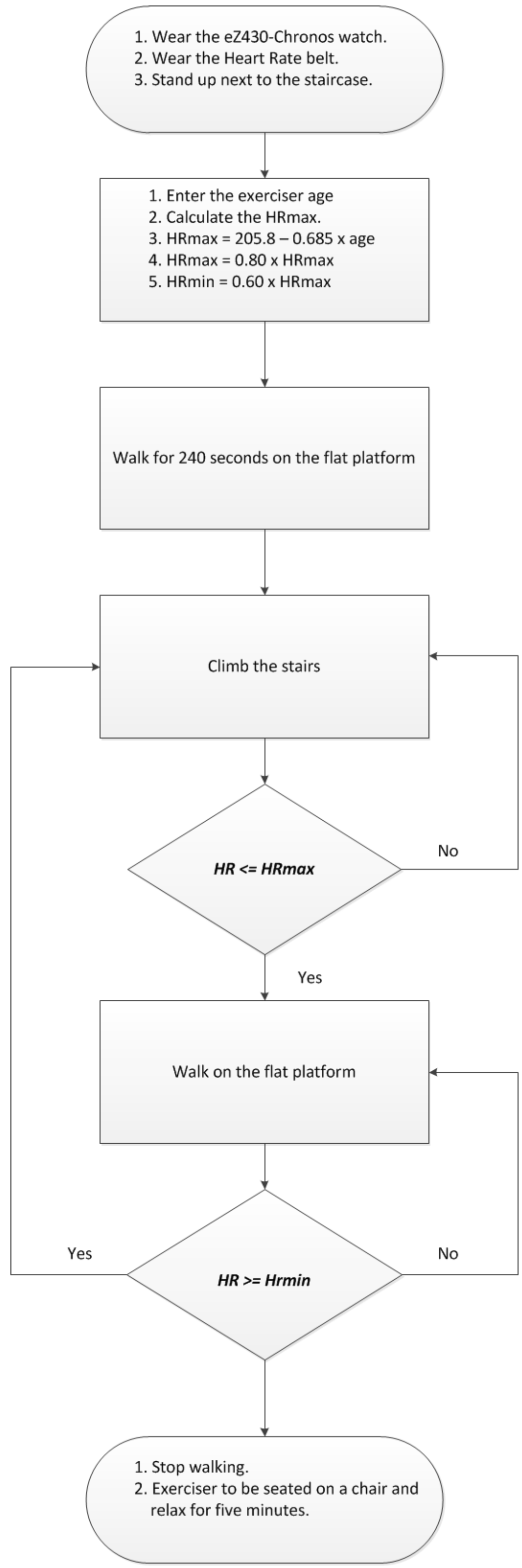


Figure 16: Outdoor exercise flowchart

6.4.4 The eZ430-Chronos watch response.

As we mentioned earlier, that the eZ430-Chronos watch will continuously monitor the heart rate and respond accordingly to help the athletes strengthening his cardiovascular system.

During the experiments, there will be three states of the eZ430-Chronos watch as described below:

1- *High heart rate.*

The heart rate is considered to be high when it reaches 80% of the HR_{max}, and once the exerciser reaches 80% of his HR_{max}, the eZ430-Chronos watch will do the following:

- Starts beeping – tone 1.
- Heart rate icon starts blinking.
- The up arrow becomes active.

The following picture of the eZ430-Chronos watch is its response during the high rate state:

Figure 17: eZ430-Chronos response during high heart rate session.

And the exerciser is highly recommended to respond to the watch and switch the exercise from high intensity to low intensity immediately to guarantee his safety and also to strengthening his cardiovascular system.

2- Low *heart rate.*

The heart rate is considered to be low when it reaches 60% of the HR_{max}, and once the exerciser reaches 60% of his HR_{max}, the eZ430-Chronos watch will do the following:

- Starts beeping – tone 2.
- Heart rate icon starts blinking.
- The down arrow becomes active.

The following picture of the eZ430-Chronos watch is its response during the low rate state:

Figure 18: eZ430-Chronos response during low heart rate session.

And the exerciser is highly recommended to respond to the watch and switch the exercise from low intensity to high intensity immediately to improve his cardiovascular system fitness.

3- Normal ***heart rate.***

The heart rate is considered to be normal when its value is more than 60% of the HR_{max}, and less than 80% of his HR_{max}, the eZ430-Chronos watch will do the following:

- Stop the beeping.
- Heart rate icon is steady.
- The up and down arrows are active.

The following picture of the eZ430-Chronos watch is its response during the normal rate state:

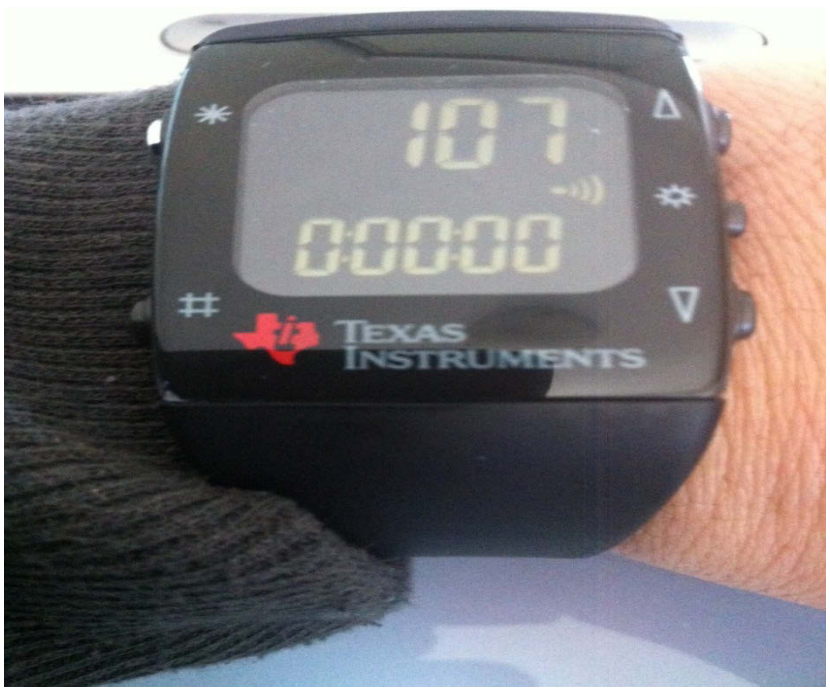


Figure 19: eZ430-Chronos response during normal heart rate session.

And the exerciser is recommended to increase the exercise intensity to improve his cardiovascular system fitness.

6.5. Conclusion.

Three volunteers were participated in this experiment both on indoor environment and on outdoor environment were the participants wear the eZ430-Chronos watch and the chest strap to monitor their heart rate continuously.

The eZ430-Chronos watch was programmed to respond according to real time measured value of the heart rate by beeping, which gives indicator to the exerciser either to increase or decrease the intensity of the exercise.

Increasing or decreasing the intensity of the exercise which means switching between high intensity and low intensity exercise and vice versa will help the athletes to improve his cardiovascular system fitness and protect him from many diseases.

Chapter 7

Conclusion and Future Work

7.1 Conclusion.

The eZ430-Chronos watch from Texas Instruments is a very powerful development tool, which has integrated heart rate monitor. This watch has been used in this study to monitor the heart rate and control the athletes' behavior based on his heart rate value.

We have tested the watch in indoor environment such as treadmill and in outdoor environment such as staircase, and we have noticed that the eZ430-Chronos watch is working just perfect in both conditions and it does help the athletes' to either increase or decrease the exercise intensity which will help the athletes' to increase his cardiovascular strengthening.

7.1 Future work.

The developed portable device and its associated algorithms can be used in various exercise monitoring and regulation related projects. For example, based on this portable system, a new effective interval training protocol has been proposed and implemented recently in Centre of Health Technologies, UTS.

Furthermore, the developed system can be utilized in the control of other automated exercise machines, such as treadmill or bicycle to help with the patient's rehabilitation program and exercise training.

Appendix I

The eZ430-Chronos code:

The eZ430-Chronos code consists of six main entities; they are the main.c, menu.c, buzzer.c, heartrate.c, bluerobin.c and stopwatch.c. The code for all parts is as following:

main.c

```
// *********************************************
// Initialization and control of application.
// *********************************************

// Include section

// system
#include "project.h"
#include <string.h>

// driver
#include "clock.h"
#include "dis.h"
#include "vti_as.h"
#include "vti_ps.h"
#include "radio.h"
#include "buzz.h"
#include "ports.h"
#include "timer.h"
#include "pmm.h"
#include "rf1a.h"

// logic
#include "menu.h"
#include "date.h"
#include "alarm.h"
#include "SW.h"
#include "battery.h"
#include "temperature.h"
#include "altitude.h"
#include "battery.h"
#include "acceleration.h"
#include "bluerobin.h"
#include "rfsimpliciti.h"
#include "simpliciti.h"
#include "test.h"
```

```
// Prototypes section
void init_application(void);
void init_global_variables(void);
void wakeup_event(void);
void process_requests(void);
void dis_update(void);
void idle_loop(void);
void configure_ports(void);
void read_calibration_values(void);



// Defines section


#define CALIBRATION_DATA_LENGTH          (13u)


// Global Variable section


volatile s_system_flags sys;


volatile s_request_flags request;


volatile s_message_flags message;

u8 rf_frequoffset;

// Function pointers for LINE1 and LINE2 display function
void (*fptr_lcd_l1)(u8 line, u8 update);
void (*fptr_lcd_l2)(u8 line, u8 update);


// Extern section

extern void start_simpliciti_sync(void);

extern u16 ps_read_register(u8 address, u8 mode);
extern u8 ps_write_register(u8 address, u8 data);


int main(void)
{
      volatile u8 ps;

      init_application();

      init_global_variables();

      test_mode();

      while(1)
      {
```

```c
	idle_loop();

	if (button.all_flags || sys.all_flags) wakeup_event();

	if (request.all_flags) process_requests();

	if (dis.all_flags) dis_update();
	}
}


void init_application(void)
{
	volatile unsigned char *ptr;


#ifdef USE_WATCHDOG
	WDTCTL = WDTPW + WDTIS__512K + WDTSSEL__ACLK;
#else
	WDTCTL = WDTPW + WDTHOLD;
#endif

	SetVCore(3);

	PMMCTL0_H  = 0xA5;
	PMMCTL0_L |= PMMHPMRE;
	PMMCTL0_H  = 0x00;

	P5SEL |= 0x03;
	UCSCTL6 &= ~XT1OFF;
	UCSCTL6 |= XCAP_3;

	UCSCTL3 = SELA__XT1CLK;
	UCSCTL4 = SELA__XT1CLK | SELS__DCOCLKDIV | SELM__DCOCLKDIV;

	_BIS_SR(SCG0);
	UCSCTL0 = 0x0000;
	UCSCTL1 = DCORSEL_5;
	UCSCTL2 = FLLD_1 + 0x16E;
	_BIC_SR(SCG0);

    __delay_cycles(250000);

	do
	{
	  UCSCTL7 &= ~(XT2OFFG + XT1LFOFFG + XT1HFOFFG + DCOFFG);
	      SFRIFG1 &= ~OFIFG;                      // Clear fault
flags
	} while ((SFRIFG1 & OFIFG));


	__disable_interrupt();

	PMAPPWD = 0x02D52;

	PMAPCTL = PMAPRECFG;
```

```
	ptr  = &P2MAP0;
	*(ptr+7) = PM_TA1CCR0A;
	P2OUT &= ~BIT7;
	P2DIR |= BIT7;

	ptr  = &P1MAP0;
	*(ptr+5) = PM_UCA0SOMI;
	*(ptr+6) = PM_UCA0SIMO;
	*(ptr+7) = PM_UCA0CLK;

	PMAPPWD = 0;
	__enable_interrupt();

	radio_reset();
	radio_powerdown();

	as_init();

	lcd_init();

	init_buttons();

	Timer0_Init();

	ps_init();
}


void init_global_variables(void)
{
	ptrMenu_L1 = &menu_L1_Time;
	ptrMenu_L2 = &menu_L2_Sync;

	fptr_lcd_l1 = ptrMenu_L1->dis_function;
	fptr_lcd_l2 = ptrMenu_L2->dis_function;

	button.all_flags  = 0;
	sys.all_flags           = 0;
	request.all_flags       = 0;
	dis.all_flags     = 0;
	message.all_flags = 0;

	dis.flag.full_update = 1;

#ifndef ISM_US
	sys.flag.use_metric_units = 1;
#endif

	read_calibration_values();

	reset_clock();

	reset_date();

	reset_alarm();

	reset_buzz();
```

```
	reset_SW();

	reset_altitude_measurement();

	reset_acceleration();

	reset_bluerobin();

	reset_rf();

	reset_temp_measurement();

	reset_batt_measurement();

	battery_measurement();
}


void wakeup_event(void)
{

	sys.flag.idle_timeout_enabled = 1;

	if (button.all_flags && sys.flag.lock_buttons)
	{
		if (!(BUTTON_NUM_IS_PRESSED && BUTTON_DOWN_IS_PRESSED))
		{
			message.flag.prepare     = 1;
			message.flag.type_locked = 1;
		}

		button.all_flags = 0;
	}

	else if (button.flag.star_long)
	{
		button.flag.star_long = 0;

		ptrMenu_L1->mx_function(LINE1);

		dis.flag.full_update = 1;
	}
	else if (button.flag.num_long)
	{
		button.flag.num_long = 0;

		ptrMenu_L2->mx_function(LINE2);

		dis.flag.full_update = 1;
	}
	else if (button.all_flags)
	{
		if(button.flag.star)
		{
			fptr_lcd_l1(LINE1, dis_LINE_CLEAR);
```

```
                ptrMenu_L1 = ptrMenu_L1->next;

                fptr_lcd_l1 = ptrMenu_L1->dis_function;

                dis.flag.line1_full_update = 1;

                button.flag.star = 0;
        }

        else if(button.flag.num)
        {
                fptr_lcd_l2(LINE2, dis_LINE_CLEAR);

                ptrMenu_L2 = ptrMenu_L2->next;

                fptr_lcd_l2 = ptrMenu_L2->dis_function;

                dis.flag.line2_full_update = 1;

                button.flag.num = 0;
        }

        else if(button.flag.up)
        {
                ptrMenu_L1->sx_function(LINE1);

                dis.flag.line1_full_update = 1;

                button.flag.up = 0;
        }

        else if(button.flag.down)
        {
                ptrMenu_L2->sx_function(LINE2);

                dis.flag.line2_full_update = 1;

                button.flag.down = 0;
        }
    }

    if (sys.all_flags)
    {
        if (sys.flag.idle_timeout)
        {
                sys.flag.idle_timeout = 0;

                clear_dis();

                dis.flag.full_update = 1;
        }
    }

    sys.flag.idle_timeout_enabled = 0;
}
```

```
void process_requests(void)
{
	if (request.flag.temperature_measurement)
temperature_measurement(FILTER_ON);

	if (request.flag.altitude_measurement)
do_altitude_measurement(FILTER_ON);

	if (request.flag.acceleration_measurement)
do_acceleration_measurement();

	if (request.flag.voltage_measurement) battery_measurement();

	if (request.flag.buzz) start_buzz(4, BUZZ_ON_TICKS,
BUZZ_OFF_TICKS);

	request.all_flags = 0;
}

void dis_update(void)
{
	u8 line;
	u8 string[8];

	if (dis.flag.full_update ||   dis.flag.line1_full_update)
	{
		clear_line(LINE1);
		fptr_lcd_l1(LINE1, dis_LINE_UPDATE_FULL);
	}
	else if (ptrMenu_L1->dis_update())
	{
		fptr_lcd_l1(LINE1, dis_LINE_UPDATE_PARTIAL);
	}


	if (message.flag.show)
	{
		line = LINE2;

if (message.flag.type_locked)            memcpy(string, "  LO?T", 6);

else if (message.flag.type_unlocked)     memcpy(string, "  OPEN", 6);

else if (message.flag.type_lobatt)       memcpy(string, "LOBATT", 6);

else if (message.flag.type_alarm_on)
		{
			memcpy(string, "  ON", 4);
			line = LINE1;
		}
		else if (message.flag.type_alarm_off)
		{
			memcpy(string, " OFF", 4);
			line = LINE1;
		}

		clear_line(line);
```

```
            fptr_lcd_l2(line, dis_LINE_CLEAR);

if (line == LINE2)      dis_chars(LCD_SEG_L2_5_0, string, SEG_ON);

else                    dis_chars(LCD_SEG_L1_3_0, string, SEG_ON);

            message.all_flags = 0;
            message.flag.erase = 1;
      }

      else if (dis.flag.full_update || dis.flag.line2_full_update)
      {
            clear_line(LINE2);
            fptr_lcd_l2(LINE2, dis_LINE_UPDATE_FULL);
      }

      else if (ptrMenu_L2->dis_update() && !message.all_flags)
      {
            fptr_lcd_l2(LINE2, dis_LINE_UPDATE_PARTIAL);
      }

      if (dis.flag.full_update)
      {
            if (is_bluerobin() == BLUEROBIN_CONNECTED)
            {
                  dis_sym(LCD_ICON_BEEPER1, SEG_ON_BLINK_OFF);
                  dis_sym(LCD_ICON_BEEPER2, SEG_ON_BLINK_OFF);
                  dis_sym(LCD_ICON_BEEPER3, SEG_ON_BLINK_OFF);
            }
      }

      dis.all_flags = 0;
}

void to_lpm(void)
{
      _BIS_SR(LPM3_bits + GIE);
      __no_operation();
}

void idle_loop(void)
{
      to_lpm();

#ifdef USE_WATCHDOG
      WDTCTL = WDTPW + WDTIS__512K + WDTSSEL__ACLK + WDTCNTCL;
#endif
}

void read_calibration_values(void)
{
      u8 cal_data[CALIBRATION_DATA_LENGTH];
      u8 i;
      u8 * flash_mem;


      flash_mem = (u8 *)0x1800;
```

```
	for (i=0; i<CALIBRATION_DATA_LENGTH; i++)
	{
		cal_data[i] = *flash_mem++;
	}

	if (cal_data[0] == 0xFF)
	{
		rf_frequoffset    = 4;
		sTemp.offset      = -250;
		sBatt.offset      = -10;
		simpliciti_ed_address[0] = 0x79;
		simpliciti_ed_address[1] = 0x56;
		simpliciti_ed_address[2] = 0x34;
		simpliciti_ed_address[3] = 0x12;
		sAlt.altitude_offset     = 0;
	}
	else
	{
		rf_frequoffset    = cal_data[1];
		if ((rf_frequoffset > 20) && (rf_frequoffset < (256-20)))
		{
			rf_frequoffset = 0;
		}
sTemp.offset      = (s16)((cal_data[2] << 8) + cal_data[3]);

sBatt.offset      = (s16)((cal_data[4] << 8) + cal_data[5]);

		simpliciti_ed_address[0] = cal_data[6];
		simpliciti_ed_address[1] = cal_data[7];
		simpliciti_ed_address[2] = cal_data[8];
		simpliciti_ed_address[3] = cal_data[9];

		if (cal_data[12] != 0xFF)
		{
sAlt.altitude_offset = (s16)((cal_data[10] << 8) + cal_data[11]);;
		}
		else
		{
			sAlt.altitude_offset = 0;
		}
	}
}
```

menu.c

```
// *******************************************************
// Menu
// *******************************************************

// Include section

// system
#include "project.h"

// driver
#include "dis.h"

// logic
#include "menu.h"
#include "user.h"
#include "clock.h"
#include "date.h"
#include "alarm.h"
#include "SW.h"
#include "temperature.h"
#include "altitude.h"
#include "battery.h"
#include "bluerobin.h"
#include "rfsimpliciti.h"
#include "acceleration.h"
#include "rfbsl.h"


// *******************************************************
// Defines section
#define FUNCTION(function)  function


// *******************************************************
// Global Variable section
const struct menu * ptrMenu_L1 = NULL;
const struct menu * ptrMenu_L2 = NULL;


// *******************************************************
// Global Variable section

void dis_nothing(u8 line, u8 update) {}

u8 update_time(void)
{
	return (dis.flag.update_time);
}
u8 update_SW(void)
{
	return (dis.flag.update_SW);
}
u8 update_date(void)
```

```
{
	return (dis.flag.update_date);
}
u8 update_alarm(void)
{
	return (dis.flag.update_alarm);
}
u8 update_temperature(void)
{
	return (dis.flag.update_temperature);
}
u8 update_battery_voltage(void)
{
	return (dis.flag.update_battery_voltage);
}
u8 update_acceleration(void)
{
	return (dis.flag.update_acceleration);
}



// Line1 - Time
const struct menu menu_L1_Time =
{
	FUNCTION(sx_time),
	FUNCTION(mx_time),
	FUNCTION(dis_time),
	FUNCTION(update_time),
	&menu_L1_Alarm,
};
// Line1 - Alarm
const struct menu menu_L1_Alarm =
{
	FUNCTION(sx_alarm),
	FUNCTION(mx_alarm),
	FUNCTION(dis_alarm),
	FUNCTION(update_alarm),
	&menu_L1_Temperature,
};
// Line1 - Temperature
const struct menu menu_L1_Temperature =
{
	FUNCTION(dummy),
	FUNCTION(mx_temperature),
	FUNCTION(dis_temperature),
	FUNCTION(update_temperature),
	&menu_L1_Altitude,
};
// Line1 - Altitude
const struct menu menu_L1_Altitude =
{
	FUNCTION(sx_altitude),
	FUNCTION(mx_altitude),
	FUNCTION(dis_altitude),
	FUNCTION(update_time),
	&menu_L1_Heartrate,
```

```
};
// Line1 - Heart Rate
const struct menu menu_L1_Heartrate =
{
	FUNCTION(sx_bluerobin),
	FUNCTION(mx_bluerobin),
	FUNCTION(dis_heartrate),
	FUNCTION(update_time),
	&menu_L1_Speed,
};
// Line1 - Speed
const struct menu menu_L1_Speed =
{
	FUNCTION(dummy),
	FUNCTION(dummy),
	FUNCTION(dis_speed),
	FUNCTION(update_time),
	&menu_L1_Acceleration,
};
// Line1 - Acceleration
const struct menu menu_L1_Acceleration =
{
	FUNCTION(sx_acceleration),
	FUNCTION(dummy),
	FUNCTION(dis_acceleration),
	FUNCTION(update_acceleration),
	&menu_L1_Time,
};

// Line2 - Date
const struct menu menu_L2_Date =
{
	FUNCTION(sx_date),
	FUNCTION(mx_date),
	FUNCTION(dis_date),
	FUNCTION(update_date),
	&menu_L2_SW,
};
// Line2 - SW
const struct menu menu_L2_SW =
{
	FUNCTION(sx_SW),
	FUNCTION(mx_SW),
	FUNCTION(dis_SW),
	FUNCTION(update_SW),
	&menu_L2_Battery,
};
// Line2 - Battery
const struct menu menu_L2_Battery =
{
	FUNCTION(dummy),
	FUNCTION(dummy),
	FUNCTION(dis_battery_V),
	FUNCTION(update_battery_voltage),
	&menu_L2_Rf,
};
// Line2 - ACC (acceleration data + button events via SimpliciTI)
```

```
const struct menu menu_L2_Rf =
{
	FUNCTION(sx_rf),
	FUNCTION(dummy),
	FUNCTION(dis_rf),
	FUNCTION(update_time),
	&menu_L2_Ppt,
};
// Line2 - PPT (button events via SimpliciTI)
const struct menu menu_L2_Ppt =
{
	FUNCTION(sx_ppt),
	FUNCTION(dummy),
	FUNCTION(dis_ppt),
	FUNCTION(update_time),
	&menu_L2_Sync,
};
// Line2 - SXNC (synchronization/data download via SimpliciTI)
const struct menu menu_L2_Sync =
{
	FUNCTION(sx_sync),
	FUNCTION(dummy),
	FUNCTION(dis_sync),
	FUNCTION(update_time),
	&menu_L2_CalDist,
};
// Line2 - Calories/Distance
const struct menu menu_L2_CalDist =
{
	FUNCTION(sx_caldist),
	FUNCTION(mx_caldist),
	FUNCTION(dis_caldist),
	FUNCTION(update_time),
	&menu_L2_RFBSL,
};
// Line2 - RFBSL
const struct menu menu_L2_RFBSL =
{
	FUNCTION(sx_rfbsl),
	FUNCTION(dummy),
	FUNCTION(dis_rfbsl),
	FUNCTION(update_time),
	&menu_L2_Date,
};
```

buzzer.c

```c
// *************************************************
// Buzzer
// *************************************************


// Include section

// system
#include "project.h"

// driver
#include "buzz.h"
#include "timer.h"
#include "dis.h"

// logic
#include "alarm.h"


// Prototypes section
void toggle_buzz(void);
void countdown_buzz(void);



// Defines section

// Global Variable section
struct buzz sbuzz;


void reset_buzz(void)
{
	sbuzz.time  = 0;
	sbuzz.state 	= buzz_OFF;
}

void start_buzz(u8 cycles, u16 on_time, u16 off_time)
{
	if (sbuzz.time == 0)
	{
		sbuzz.time    = cycles;
		sbuzz.on_time  = on_time;
		sbuzz.off_time = off_time;

		TA1CTL = TACLR | MC_1 | TASSEL__ACLK;

		TA1CCR0 = buzz_TIMER_STEPS;

		TA1CCTL0 = OUTMOD_4;

		P2SEL |= BIT7;
```

```
		fptr_Timer0_A3_function = toggle_buzz;
		Timer0_A3_Start(sbuzz.on_time);

		sTimer.timer0_A3_ticks = sbuzz.off_time;

		sbuzz.state = buzz_ON_OUTPUT_ENABLED;
	}
}


void toggle_buzz(void)
{
	if (sbuzz.state == buzz_ON_OUTPUT_ENABLED)
	{
		TA1CTL &= ~(BIT4 | BIT5);

		P2OUT &= ~BIT7;
		P2SEL &= ~BIT7;

		sbuzz.state = buzz_ON_OUTPUT_DISABLED;

		sTimer.timer0_A3_ticks = sbuzz.on_time;
	}
	else
	{

		countdown_buzz();

		if (sbuzz.state != buzz_OFF)
		{
			TA1R = 0;
			TA1CTL |= MC_1;
			P2SEL |= BIT7;

			sbuzz.state = buzz_ON_OUTPUT_ENABLED;

			sTimer.timer0_A3_ticks = sbuzz.off_time;
		}
	}
}


void stop_buzz(void)
{
	TA1CTL &= ~(BIT4 | BIT5);

	P2OUT &= ~BIT7;
	P2SEL &= ~BIT7;

	TA1CCTL0 &= ~CCIE;

	Timer0_A3_Stop();

	reset_buzz();
}
```

```c
u8 is_buzz(void)
{
	return (sbuzz.state != buzz_OFF);
}


void countdown_buzz(void)
{
	if (--sbuzz.time == 0)
	{
		stop_buzz();
	}
}


void start_buzz1(void)
{

			TA1R = 0;
			TA1CTL |= MC_1;
			P2SEL |= BIT7;

}
```

heartrate.c

```c
// *********************************************
// Heart Rate Monitor
// *********************************************


// Include section

// system
#include "project.h"

// driver
#include "dis.h"
#include "buzz.h"
#include "ports.h"
#include "timer.h"
#include "bluerobin.h"

// logic
#include "alarm.h"
#include "clock.h"
#include "user.h"
#include "SW.h"
```

```
void HRMonitor(void)
{

      sBlueRobin.user_age     = 30;
      set_value(&age, 3, 2, USER_AGE_MIN, USER_AGE_MAX,
SETVALUE_dis_VALUE + SETVALUE_NEXT_VALUE, LCD_SEG_L2_4_2, dis_value1);

      HRMAX = 205.8 - 0.685 x sBlueRobin.user_age;

      MAX = 0.80 x HRMAX;
      MIN = 0.60 x HRMAX;


if (sBlueRobin.heartrate >= MAX)
{
      start_buzz(4, CONV_MS_TO_TICKS(20), CONV_MS_TO_TICKS(100));
      start_SW();
      dis_sym(LCD_ICON_HEART, SEG_ON_BLINK_ON);
      dis_sym(LCD_ICON_ALARM, SEG_ON_BLINK_ON);
      dis_sym(LCD_SYMB_ARROW_UP, SEG_ON_BLINK_OFF);
      dis_sym(LCD_SYMB_ARROW_DOWN, SEG_ON_BLINK_ON);

}

if (sBlueRobin.heartrate <= MIN)
{
       start_buzz(8, CONV_MS_TO_TICKS(20), CONV_MS_TO_TICKS(100));
       stop_SW();
       reset_SW();
       dis_sym(LCD_ICON_HEART, SEG_ON_BLINK_ON);
       dis_sym(LCD_ICON_ALARM, SEG_ON_BLINK_ON);
      dis_sym(LCD_SYMB_ARROW_UP, SEG_ON_BLINK_ON);
      dis_sym(LCD_SYMB_ARROW_DOWN, SEG_ON);
}



if (sBlueRobin.heartrate > MIN && sBlueRobin.heartrate < MAX)
{
stop_alarm();
stop_SW();
reset_SW();
}

}
```

bluerobin.c

```c
// *************************************************
// BlueRobin functions.
// *************************************************


// *************************************************
// Include section

// system
#include "project.h"

// driver
#include "dis.h"
#include "radio.h"
#include "ports.h"
#include "timer.h"
#include "rf1a.h"
#include "buzz.h"

// logic
#include "BlueRobin_RX_API.h"
#include "bluerobin.h"
#include "rfsimpliciti.h"
#include "user.h"
#include "alarm.h"
#include "cc430x613x.h"




void dis_calories(u8 line, u8 update);
void dis_distance(u8 line, u8 update);


#define REMEMBER_TX_ID              (FALSE)
#define INITSTATE    0
#define BACKLIGHT    BIT3

struct br sBlueRobin;

extern void BRRX__StopTimer_v(void);


extern u8 rf_frequoffset;


void reset_bluerobin(void)
{
      sBlueRobin.state = BLUEROBIN_OFF;

      sBlueRobin.cs_id        = 0;

      sBlueRobin.update             = BLUEROBIN_NO_UPDATE;
```

```
      sBlueRobin.heartrate    = 0;
      sBlueRobin.speed        = 0;
      sBlueRobin.distance     = 0;
      sBlueRobin.calories     = 0;

      sBlueRobin.user_weight  = 75;

      sBlueRobin.caldist_view = 0;
}


void mx_bluerobin(u8 line)
{
#if REMEMBER_TX_ID == TRUE
  u8 i;

      sBlueRobin.cs_id = 0;

  dis_chars(LCD_SEG_L1_2_0, (u8*)"CLR", SEG_ON);
  for (i=0; i<4; i++) Timer0_A4_Delay(CONV_MS_TO_TICKS(500));
#endif

      button.all_flags = 0;
}


void sx_bluerobin(u8 line)
{
      u8 stop = 0;

      if (sys.flag.low_battery) return;

      if (is_rf()) return;

      if(button.flag.up)
      {
            if (sBlueRobin.state == BLUEROBIN_OFF)
            {
                  open_radio();

                  BRRX_Init_v();

                  BRRX_SetPowerdownDelay_v(10);
                  BRRX_SetSearchTimeout_v(8);

#if REMEMBER_TX_ID == TRUE

if (sBlueRobin.cs_id == 0) BRRX_SetSignalLevelReduction_v(5);

#else
                  sBlueRobin.cs_id = 0;
                  BRRX_SetSignalLevelReduction_v(5);
#endif

                  WriteSingleReg(FSCTRL0, rf_frequoffset);

                  sBlueRobin.state = BLUEROBIN_SEARCHING;
```

```
			dis_sym(LCD_ICON_BEEPER1, SEG_ON_BLINK_ON);
			dis_sym(LCD_ICON_BEEPER2, SEG_ON_BLINK_ON);
			dis_sym(LCD_ICON_BEEPER3, SEG_ON_BLINK_ON);

			if (BRRX_GetState_t(HR_CHANNEL) == TX_OFF)
			{
				BRRX_SetID_v(HR_CHANNEL, sBlueRobin.cs_id);
				BRRX_Start_v(HR_CHANNEL);

				while (BRRX_GetState_t(HR_CHANNEL)==TX_SEARCH)
				{
					Timer0_A4_Delay(CONV_MS_TO_TICKS(200));
			}
			}

			if (BRRX_GetState_t(HR_CHANNEL)==TX_ACTIVE)
			{
				sBlueRobin.state = BLUEROBIN_CONNECTED;

				if (sBlueRobin.cs_id == 0)
				{
				sBlueRobin.cs_id = BRRX_GetID_u32(HR_CHANNEL);
				}

				dis_sym(LCD_ICON_BEEPER1, SEG_ON_BLINK_OFF);
				dis_sym(LCD_ICON_BEEPER2, SEG_ON_BLINK_OFF);
				dis_sym(LCD_ICON_BEEPER3, SEG_ON_BLINK_OFF);
				dis_sym(LCD_ICON_HEART, SEG_ON_BLINK_ON);
			}
			else
			{
				stop = 1;
			}
		}
		else if (sBlueRobin.state == BLUEROBIN_CONNECTED)
		{

			stop = 1;
		}
	}

if (stop)
	{
		stop_bluerobin();
	}
}


void mx_caldist(u8 line)
{
	u8 select;
	s32 kcalories;
	s32 weight;

	clear_dis_all();
```

```
	kcalories	= sBlueRobin.calories/1000;
if (sys.flag.use_metric_units)		weight = sBlueRobin.user_weight;
	else		weight = ((s32)sBlueRobin.user_weight * 2205) / 1000;

	select = 0;

	while(1)
	{
		if (sys.flag.idle_timeout) break;

		if (button.flag.star)
		{
			sBlueRobin.calories	= kcalories*1000;
			if (sys.flag.use_metric_units)
	sBlueRobin.user_weight = weight;
			else
	sBlueRobin.user_weight = (weight * 1000) / 2205;

			dis.flag.line1_full_update = 1;

			break;
		}

		switch (select)
		{
			case 0:
dis_sym(LCD_UNIT_L2_KCAL, SEG_ON);
set_value(&kcalories, 6, 5, 0, 199999, SETVALUE_dis_VALUE +
SETVALUE_FAST_MODE + SETVALUE_NEXT_VALUE, LCD_SEG_L2_5_0, dis_value1);
dis_sym(LCD_UNIT_L2_KCAL, SEG_OFF);
clear_line(LINE2);

select = 1;

break;
			case 1:
					if (sys.flag.use_metric_units)
					{
dis_chars(LCD_SEG_L2_1_0, (u8 *)"KG", SEG_ON);
set_value(&weight, 3, 2, USER_WEIGHT_MIN_KG, USER_WEIGHT_MAX_KG,
SETVALUE_dis_VALUE + SETVALUE_NEXT_VALUE, LCD_SEG_L2_4_2, dis_value1);
					}
					else
					{
dis_chars(LCD_SEG_L2_1_0, (u8 *)"LB", SEG_ON);
set_value(&weight, 3, 2, USER_WEIGHT_MIN_LB, USER_WEIGHT_MAX_LB,
SETVALUE_dis_VALUE + SETVALUE_NEXT_VALUE, LCD_SEG_L2_4_2, dis_value1);
					}
select = 0;
break;
		}
	}

	button.all_flags = 0;
}
```

```
void sx_caldist(u8 line)
{
	dis_caldist(line, dis_LINE_CLEAR);

	if (sBlueRobin.caldist_view == 0)	sBlueRobin.caldist_view = 1;
	else
	sBlueRobin.caldist_view = 0;

	dis_caldist(line, dis_LINE_UPDATE_FULL);
}


void dis_heartrate(u8 line, u8 update)
{
	u8 * str;

	if (update != dis_LINE_CLEAR)
	{
		if (is_bluerobin())
		{
			str = itoa(sBlueRobin.heartrate, 3, 2);
			dis_chars(LCD_SEG_L1_2_0, str, SEG_ON);
		}
		else
		{
			dis_chars(LCD_SEG_L1_2_0, (u8 *)"---", SEG_ON);
		}
	}

	if (!is_bluerobin())
	{
		if (update == dis_LINE_UPDATE_FULL)
		{
			 dis_sym(LCD_ICON_HEART, SEG_ON);
		}
		else if (update == dis_LINE_CLEAR)
		{
			dis_sym(LCD_ICON_HEART, SEG_OFF);
		}
	}
}


void dis_speed(u8 line, u8 update)
{
	u8 milesPerHour;
	u8 * str;

	if (update != dis_LINE_CLEAR)
	{
		if (sys.flag.use_metric_units)
		{
			str = itoa(sBlueRobin.speed, 3, 1);
		}
		else
		{
			milesPerHour = (u16)(sBlueRobin.speed * 0.6214);
```

```
                str = itoa(milesPerHour, 3, 1);
            }
            dis_chars(LCD_SEG_L1_2_0, str, SEG_ON);
        }

        if (update == dis_LINE_UPDATE_FULL)
        {
            dis_sym(LCD_SEG_L1_DP0, SEG_ON);
            if (sys.flag.use_metric_units)
            {
                dis_sym(LCD_UNIT_L1_K, SEG_ON);
                dis_sym(LCD_UNIT_L1_M, SEG_ON);
            }
            else
            {
                dis_sym(LCD_UNIT_L1_M, SEG_ON);
                dis_sym(LCD_UNIT_L1_I, SEG_ON);
            }
            dis_sym(LCD_UNIT_L1_PER_H, SEG_ON);
        }
        else if (update == dis_LINE_CLEAR)
        {
            dis_sym(LCD_SEG_L1_DP0, SEG_OFF);
            dis_sym(LCD_UNIT_L1_K, SEG_OFF);
            dis_sym(LCD_UNIT_L1_M, SEG_OFF);
            dis_sym(LCD_UNIT_L1_M, SEG_OFF);
            dis_sym(LCD_UNIT_L1_I, SEG_OFF);
            dis_sym(LCD_UNIT_L1_PER_H, SEG_OFF);
        }
}

void dis_distance(u8 line, u8 update)
{
        u8 * str;
        u32 miles;

        if (update != dis_LINE_CLEAR)
        {
            if (sys.flag.use_metric_units)
            {
                if (sBlueRobin.distance < 2000000)
                {
                    str = itoa(sBlueRobin.distance/10, 6, 3);
                }
                else
                {
                    str = itoa(199999, 6, 3);
                }
            }
            else
            {
                miles = (u32)(sBlueRobin.distance * 0.06214);

                if (miles < 2000000)
                {
                    str = itoa(miles, 6, 3);
                }
```

```
                    else
                    {
                          str = itoa(199999, 6, 3);
                    }
              }
              dis_chars(LCD_SEG_L2_5_0, str, SEG_ON);
       }

       if (update == dis_LINE_UPDATE_FULL)
       {
              if (sys.flag.use_metric_units)
              {
                    dis_sym(LCD_UNIT_L2_KM, SEG_ON);
              }
              else
              {
                    dis_sym(LCD_UNIT_L2_MI, SEG_ON);
              }
              dis_sym(LCD_SEG_L2_DP, SEG_ON);
       }
       else if (update == dis_LINE_CLEAR)
       {
              dis_sym(LCD_UNIT_L2_KM, SEG_OFF);
              dis_sym(LCD_UNIT_L2_MI, SEG_OFF);
              dis_sym(LCD_SEG_L2_DP, SEG_OFF);
       }
}


void dis_caldist(u8 line, u8 update)
{
       if (sBlueRobin.caldist_view == 0)   dis_calories(line, update);
       else
       dis_distance(line, update);
}


void dis_calories(u8 line, u8 update)
{
       u8 * str;

       if (update != dis_LINE_CLEAR)
       {
              str = itoa(sBlueRobin.calories / 1000, 6, 5);
              dis_chars(LCD_SEG_L2_5_0, str, SEG_ON);
       }

       if (update == dis_LINE_UPDATE_FULL)
       {
              dis_sym(LCD_UNIT_L2_KCAL, SEG_ON);
       }
       else if (update == dis_LINE_CLEAR)
       {
              dis_sym(LCD_UNIT_L2_KCAL, SEG_OFF);
       }
}
```

```
u8 is_bluerobin(void)
{
      return (sBlueRobin.state == BLUEROBIN_CONNECTED);
}


u8 is_bluerobin_searching(void)
{
      return (sBlueRobin.state == BLUEROBIN_SEARCHING);
}


void get_bluerobin_data(void)
{
      u16 calories;
      brtx_state_t bChannelState;



      bChannelState = BRRX_GetState_t(HR_CHANNEL);

      switch (bChannelState)
      {
            case TX_ACTIVE:
      sBlueRobin.heartrate = BRRX_GetHeartRate_u8();

      sBlueRobin.speed    = BRRX_GetSpeed_u8();

      sBlueRobin.distance  = BRRX_GetDistance_u16();

      if (sBlueRobin.distance > 2000000) sBlueRobin.distance = 0;


      if (sBlueRobin.heartrate >= 65 && sBlueRobin.user_weight != 0)
{
calories = ((sBlueRobin.heartrate - 60) * sBlueRobin.user_weight) / 32;


if (sBlueRobin.calories > 200000000) sBlueRobin.calories = 0;
                                                  }

// Heart Rate Monitor will continuously monitor the HR and responses
accordingly

HRMonitor();

sBlueRobin.update = BLUEROBIN_NEW_DATA;

break;

            case TX_OFF:
                                          stop_bluerobin();
break;

            default:
break;
      }
```

```
}

void stop_bluerobin(void)
{
	sBlueRobin.state = BLUEROBIN_OFF;

	BRRX_Stop_v(HR_CHANNEL);

	close_radio();

	sBlueRobin.heartrate		= 0;
	sBlueRobin.speed		= 0;
	sBlueRobin.distance		= 0;
	dis.flag.full_update	= 1;

	dis_sym(LCD_ICON_HEART, SEG_OFF_BLINK_OFF);
	dis_sym(LCD_ICON_BEEPER1, SEG_OFF_BLINK_OFF);
	dis_sym(LCD_ICON_BEEPER2, SEG_OFF_BLINK_OFF);
	dis_sym(LCD_ICON_BEEPER3, SEG_OFF_BLINK_OFF);
}
```

stopwatch.c

```
// *********************************************
// StopWatch
// *********************************************


// Include section

// system
#include "project.h"
#include <string.h>

// driver
#include "SW.h"
#include "ports.h"
#include "dis.h"
#include "timer.h"

// logic
#include "menu.h"


// Prototypes section
void start_SW(void);
void stop_SW(void);
void reset_SW(void);
```

```c
void SW_tick(void);
void update_SW_timer(void);
void mx_SW(u8 line);
void sx_SW(u8 line);
void dis_SW(u8 line, u8 update);



// Defines section

// Global Variable section
struct SW sSW;


// Extern section

void update_SW_timer(void)
{
	u16 value;

	if (sSW.viewStyle == DIS_DEFAULT_VIEW)
	{
		value = TA0CCR2 + SW_100HZ_TICK;

		if (sSW.swtIs1Hz)
		{
			value -= 5;
			sSW.swtIs1Hz = 0;
			sSW.swtIs10Hz = 0;
		}
		else if (sSW.swtIs10Hz)
		{
			value -= 3;
			sSW.swtIs10Hz = 0;
		}
	}
	else
	{

		value = TA0CCR2 + SW_1HZ_TICK;
	}


	TA0CCR2 = value;
}

void SW_tick(void)
{
	static u8 delay = 0;

	if (sSW.viewStyle == DIS_DEFAULT_VIEW)
	{
		sSW.time[7]++;

		if (delay++ > 17)
		{
			sSW.drawFlag = 8;
```

```
			delay = 0;
		}

		if (sSW.time[7] == 0x3A)
		{
			sSW.time[7]='0';
			sSW.time[6]++;

			sSW.swtIs10Hz = 1;

			sSW.drawFlag = 7;
		}
	}
	else
	{

		sSW.time[6] = 0x3A;
	}

	if (sSW.time[6] == 0x3A)
	{
	sSW.drawFlag = 1;

		sSW.swtIs1Hz = 1;

		sSW.time[6]='0';
		sSW.time[5]++;
		if (sSW.time[5] == 0x3A)
		{
			sSW.drawFlag++;
			sSW.time[5] = '0';
			sSW.time[4]++;
			if (sSW.time[4] == '6')
			{
				sSW.drawFlag ++;
				sSW.time[4] = '0';
				sSW.time[3]++;
				if (sSW.time[3] == 0x3A)
				{
					sSW.drawFlag++;
					sSW.time[3] = '0';
					sSW.time[2]++;
					if (sSW.time[2] == '2')
					{
				sSW.viewStyle = DIS_ALTERNATIVE_VIEW;
				dis_SW(LINE2, DIS_LINE_UPDATE_FULL);

					}
					else if (sSW.time[2] == '6')
					{
						sSW.drawFlag++;
						sSW.time[2] = '0';
						sSW.time[1]++;

						if (sSW.time[1] == 0x3A)
						{
							sSW.drawFlag++;
```

```
                                    sSW.time[1] = '0';
                                    sSW.time[0]++;

                                    if (sSW.time[0] == '2')
                                    {
                                          reset_SW();
                                          sSW.state = SW_RUN;
                  dis_SW(LINE2, DIS_LINE_UPDATE_FULL);
                                    }
                              }
                        }
                  }
            }
      }
}

      dis.flag.update_SW = 1;
}


void reset_SW(void)
{
      memcpy(sSW.time, "00000000", sizeof(sSW.time));

      sSW.swtIs10Hz     = 0;
      sSW.swtIs1Hz      = 0;

      sSW.state         = SW_STOP;

      sSW.viewStyle     = DIS_DEFAULT_VIEW;
}

u8 is_SW(void)
{
      return ((sSW.state == SW_RUN) && (ptrMenu_L2 == &menu_L2_SW));
}

void start_SW(void)
{
      sSW.state = SW_RUN;

      TA0CCR2 = TA0R;

      update_SW_timer();


      TA0CCTL2 &= ~CCIFG;
      TA0CCTL2 |= CCIE;

      dis_sym(LCD_ICON_SW, SEG_ON);
}

void stop_SW(void)
{
      TA0CCTL2 &= ~CCIE;
```

```
	sSW.state = SW_STOP;

	dis_sym(LCD_ICON_SW, SEG_OFF);

	dis_SW(LINE2, DIS_LINE_UPDATE_FULL);
}


void mx_SW(u8 line)
{
	stop_SW();

	reset_SW();

	dis_SW(line, DIS_LINE_UPDATE_FULL);
}


void sx_SW(u8 line)
{
	if(button.flag.down)
	{
		if (sSW.state == SW_STOP)
		{
			start_SW();
		}
		else
		{
			stop_SW();
		}

	}
}


void dis_SW(u8 line, u8 update)
{
	if (update == DIS_LINE_UPDATE_PARTIAL)
	{
		if (dis.flag.update_SW)
		{
			if (sSW.viewStyle == DIS_DEFAULT_VIEW)
			{
				if(sSW.drawFlag != 0)
				{
					switch(sSW.drawFlag)
					{
		case 4: dis_char(LCD_SEG_L2_5, sSW.time[2], SEG_ON);
		case 3: dis_char(LCD_SEG_L2_4, sSW.time[3], SEG_ON);
		case 2: dis_char(LCD_SEG_L2_3, sSW.time[4], SEG_ON);
		case 1: dis_char(LCD_SEG_L2_2, sSW.time[5], SEG_ON);
		case 7: dis_char(LCD_SEG_L2_1, sSW.time[6], SEG_ON);
		case 8:	dis_char(LCD_SEG_L2_0, sSW.time[7], SEG_ON);
					}
				}
			}
			else
```

```
            {
                switch(sSW.drawFlag)
                {
        case 6: dis_char(LCD_SEG_L2_5, sSW.time[0], SEG_ON);
        case 5: dis_char(LCD_SEG_L2_4, sSW.time[1], SEG_ON);
        case 4: dis_char(LCD_SEG_L2_3, sSW.time[2], SEG_ON);
        case 3: dis_char(LCD_SEG_L2_2, sSW.time[3], SEG_ON);
        case 2: dis_char(LCD_SEG_L2_1, sSW.time[4], SEG_ON);
        case 1: dis_char(LCD_SEG_L2_0, sSW.time[5], SEG_ON);
                }
            }
        }
    }
    else if (update == DIS_LINE_UPDATE_FULL)
    {
        if (sSW.viewStyle == DIS_DEFAULT_VIEW)
        {
            dis_chars(LCD_SEG_L2_5_0, sSW.time+2, SEG_ON);
        }
        else
        {
            dis_chars(LCD_SEG_L2_5_0, sSW.time, SEG_ON);
        }
        dis_sym(LCD_SEG_L2_COL1, SEG_ON);
        dis_sym(LCD_SEG_L2_COL0, SEG_ON);
    }
    else if (update == DIS_LINE_CLEAR)
    {

    }
}
```

Bibliography

[1] National Strength and Conditioning Association (2000). *"Essentials of Strength training and Conditioning",* 2nd ed., Champaign, IL: Human Kinetics.

[2] Bishop D., *"Warm up ii: performance changes following active warm up and how to structure the warm up,"* Sports Med, vol. 33, no. 7, pp. 484 - 498, 2003.

[3] deVries, 1980; McArdle et al',1981

[4] American College of Sports Medicine, *"Acsm's guidelines for exercise testing and prescription,"* Baltimore: Lippincott Williams and Wilkins, vol. 7, 2006.

[5] Inbar O, Oren A, Scheinowitz M, Rotstein A, Dlin R, Casaburi R, *"Normal cardiopulmonary responses during incremental exercise in 20-70-yr-old men,"* Med Sci Sport Exercise, vol. 26, no. 5, pp. 538-546, 1994.

[6] Balderrama C, Ibarra G, De La Riva J, Lopez S, *"Evaluation of three methodologies to estimate the vo2max in people of different ages,"* Applied Ergonomics, vol. 42, pp. 162-168, 2010.

[7] http://dictionary.reference.com/browse/cooling+down

[8] *"National Physical Activity Guidelines, Glossary".* The State of Queensland. Retrieved December 28, 2010.

[9] Donatelle, Rebecca J., *"Health, the basics,";* San Francisco: Pearson Education, Inc., vol. 6.

[10] *"The American Heritage Science Dictionary";* Houghton Mifflin Harcourt Publishing Company. 2010.

[11] *"The American Heritage Science Dictionary";* Houghton Mifflin Harcourt Publishing Company. 2005

[12] *"The American Heritage &reg";* Dictionary of the English Language, Fourth Edition. Houghton Mifflin Company, 2004.

[13] Jackson, A. S., Ross, R. M., *"Understanding exercise for health and fitness,";* Houston: CSI Software, vol. 2, 1992.

[14] Davis J. *"Direct determination of aerobic power. In: Foster"* PJMC, editor. Physiological Assessment of Human Fitness. 1st edition. Champiagn, Ill, USA: Human Kinetics; 1995. pp. 9–18.

[15] Bruce RA, Kusumi F, Hosmer D. *"Maximal oxygen intake and nomographic assessment of functional aerobic impairment in cardiovascular disease"*. The American Heart Journal. 1973;85(4):546–562. [PubMed]

[16] Ebbeling CB, Ward A, Puleo EM, Widrick J, Rippe JM. *"Development of a single-stage submaximal treadmill walking test"*. Medicine and Science in Sports and Exercise. 1991;23(8):966–973.

[17] Astrand PO, Ryhming I. *"A nomogram for calculation of aerobic capacity (physical fitness) from pulse rate during sub-maximal work"*. Journal of Applied Physiology. 1954;7(2):218–221.

[18] Holter NJ. *"New method for heart studies"*. Science 1961; 134:1214-20

[19] Laukkanen RMT. Virtanen PK. *"Heart rate monitors: state of the art".* J Sport Sci 1998; 16: S3-7

[20] Lippincott Williams and Wilkins, *"Acsm's guidelines for exercise: testing and prescription,"* American College of Sports Medicine, vol. 6, 2000.

[21] Myung-kyung Suh, Ani Nahapetian, Jonathan Woodbridge, Mahsan Rofouei, Majid Sarrafzadeh, *"Machine Learning-Based Adaptive Wireless Interval Training Guidance System",* DOI 10.1007/s11036-011-0331-5

[22] Chilibeck, P.D., G.J. Bell, R.P. Farrar, and T.P. Martin. *"Higher mitochondrial fatty acid oxidation following intermittent versus continuous endurance exercise training"*. Can. J. Physiol. Pharmacol.76(9):891–894. 1998.

[23] Warburton DE, McKenzie DC, Haykowsky MJ, Taylor A, Shoemaker P, Ignaszewski AP & Chan SY (2005). *"Effectiveness of high-intensity interval training for the rehabilitation of patients with coronary artery disease"*. Am J Cardiol 95, 1080–1084.

[24] Wisløff U, Støylen A, Loennechen JP, Bruvold M, Rognmo Ø, HaramPM, Tjønna AE, Helgerud J, Slørdahl SA, Lee SJ, VidemV, Bye A, Smith GL,Najjar SM, EllingsenØ& kjaerpe T (2007). *"Superior cardiovascular effect of aerobic interval training versus moderate continuous training in heart failure patients: a randomized study"*. Circulation 115, 3086–3094.

[25] Moholdt TT, Amundsen BH, Rustad LA,Wahba A, Løvø KT, Gullikstad LR, Bye A, Skogvoll E,Wisløff U, Slørdahl SA (2009). *"Aerobic interval training versus continuous moderate exercise after coronary artery bypass surgery: a randomized study of cardiovascular effects and quality of life"*. Am Heart J 158, 1031–1037.

[26] Munk PS, Staal EM, Butt N, Isaksen K & Larsen AI (2009). *"High-intensity interval training may reduce in-stent restenosis following percutaneous coronary intervention with stent implantation A randomized controlled trial evaluating the relationship to endothelial function and inflammation"*. Am Heart J 159, 734–741.

[27] Whyte LJ, Gill JM & Cathcart AJ (2010). *"Effect of 2 weeks of sprint interval training on health-related outcomes in sedentary overweight/obese men"*. Metabolism 59, 1421– 1428.

[28] Rognmo Ø, Hetland E, Helgerud J, Hoff J & Slørdahl SA (2004). *"High intensity aerobic interval exercise is superior to moderate intensity exercise for increasing aerobic capacity in patients with coronary artery disease"*. Eur J Cardiovasc Prev Rehabil 11, 216–222.

[29] Schjerve IE, Tyldum GA, Tjønna AE, Stølen T, Loennechen JP, Hansen HE, Haram PM, Heinrich G, Bye A, Najjar SM, Smith GL, Slørdahl SA, Kemi OJ &Wisløff U (2008). *"Both aerobic endurance and strength training programmes improve cardiovascular health in obese adults"*. Clin Sci (Lond) 115, 283–293.

[30] Hawley, J.A., K.H. Myburgh, T.D. Noakes, and S.C. Dennis. *"Training techniques to improve fatigue resistance and enhance endurance performance."* J. Sports Sci. 15(3):325–333. 1997.

[31] Karp JR, *"Interval training for the fitness professional",* National Strength & Conditioning Association Volume 22, Number 4, pages 64–69

[32] Texas Instruments website:

http://www.ti.com/tool/ccstudio?dcmp=PPC_Google_TI&k_clickid=70458994-60b2-c2c8-c3bb-000061505325

[33] Eclipse website:
"*http://help.eclipse.org/luna/index.jsp?topic=%2Forg.eclipse.platform.doc.isv%2Fguide%2Fint_eclipse.htm*

[34] Kindler, E.; Krivy, I. (2011). *"Object-Oriented Simulation of systems with sophisticated control".* International Journal of General Systems. pp. 313–343.

[35] Lewis, John; Loftus, William (2008). *Java Software Solutions Foundations of Programming Design 6th ed. Pearson Education Inc. ISBN 0-321-53205-8., section 1.6 "Object-Oriented Programming"*