Software Development - 3K04

Assignment 2 - Part 2 - DCM

Instructor: Dr. Alan Wassyng

November 27, 2017

Group 7 - "Group 4"

Adam Cool	400032857	coola
Andrew Rehkopf	001412499	rehkopaz
Evan Gagich	400027794	gagiche
Ibrahim Tannir	001204850	tanniri
Michael Henry	400011728	henrym1
Michael Soosaipillai	400034820	soosaipm
Moshiur Howlader	001316948	howlam

Contents

1 Introduction

The DCM is improved from assignment 1 in a variety of ways. First of all, serial communication with the pacemaker has been completed as well as electrogram functionality. Unfortunately, due to repeated issues with serial communication and the board, some elements intended to be functional for this assignment had to be scraped to simplify the design.

2 Parameter Send

2.1 Design Decisions

The parameter sending from DCM to pacemaker occurs over the serial port of the pacemaker. It is enabled by a function which is called upon a "Save and Send" button press. Upon this function call, an array of uint8 parameters are sent to the pacemaker. Hidden alongside this functionality, any modifications to the programmable parameters are saved to a .MAT file. If the file does not exist, then it will be created and the values corresponding to each parameter are stored in numerical double form. Whenever the parameters are to be sent to the pacemaker, the values are read from the users input and appropriately checked to ensure that none of the values are invalid (for example, pulse width of 0). Some values to be sent are too large to fit in the standard uint8 value to be sent thus their are either cast or shifted into two uint8s from a single uint16.

What is hidden from the user is how these values are stored and how they are transmitted. In addition to all the values which are seen on the programmable parameters GUI screen, there is a secret code appended to the serial parameter transmission to control the electrogram. This code is configured by the drop down that either chooses ventricle or atrium monitoring. Note that egram is never enabled upon updating parameters because likely the user only wants to see egram upon pressing the "Show Egram" button.

The black box behaviour of this function triggered upon button press is to take the parameters inputted to the GUI and configure the pacemaker to use them.

Some rather tough design decisions had to be made regarding the nature of the serial transmissions as serial posed such a challenge. In order to simplify the design as much as possible, values which were over 256 (uint8 maximum) were divided by 10 or 100 prior to sending and the data loss through this integer division was thrown out.

2.2 Future Changes

As previously mentioned, there are many things that were not accomplished with this iteration of the DCM simply due to the challenges presented by serial communication. In future designs, the accuracy of the parameters sent will be improved by employing a bit-shifting into two packages method. Although we doubt that the FRDM-K64F is accurate to the mV, this level of accuracy is easy to achieve through serial communication provided that the transmission occurs properly.

Some additional safety features were dropped from the design to simplify the serial communication. Initially we had planned to include a pacemaker ID that was to be read from the pacemaker every so often such that the user of the DCM can know that the device connected is the device they expect to be communicating with. In addition, we would have liked to include a parameter change confirmation sent from the pacemaker to the DCM upon the successful change of parameters. These design decisions are likely to change in future prototypes provided serial communication can work properly.

As for requirement changes, there are many considerations that are possible as changes to the pacemaker will require different support from the DCM. For example, there may be additional parameters that are to be sent or perhaps the units or range of potential values for parameters will change.

3 Electrogram

3.1 Design Decisions

The electrogram is a feature that allows those in control of the DCM to view what is electrically happening within particular heart chambers in real time. The DCM implements this in a simple manner by receiving serially the ADC

voltages probed by the pacemaker in unit8 form. These data points are streamed in real time at a rate of approximately 1 sample per millisecond. This egram functionality was encapsulated within the egram display window.

The egram is controlled by a press of a button which sends all of the pacemaker parameters and a code for the appropriate chamber which is to be paced. Because of the way serial transmission is done, the entire list of parameters must be sent at one time however this is hidden from the user. The black box behaviour of this function is button press enables a graph window which shows the voltage activity within the heart chambers.

3.2 Future Changes

At the moment, the egram control can only access the behaviour of one chamber at a time. In later designs, we hope to allow for both chambers to be monitored at the same time which would require perhaps some kind of identifier appended to the serial package received to differentiate between the chamber signals. Due to the way serial transmissions are sent, in order to enable egram, all the parameters of the pacemaker must be sent including a separate member for egram control. This is fairly wasteful in terms of serial usage and as pacemakers must be efficient devices, perhaps a method where egram is enabled by a single value can be employed.

As for requirements changes, there are certainly many things that can become necessary in future designs. One thing is the ability to scale the incoming voltage to be more accurate. At the moment, the raw ADC value is sent from the pacemaker, we could potentially scale and normalize this value on the pacemaker end of the transmission for a better result in the DCM. In addition, if were required to have a more accurate picture of the behaviour within the heart, greater accuracy in ADC samples could be obtained on perhaps a uint16 which would allow for fine deviations of voltage reading. As this greater accuracy comes at a cost of reduced sample rate, there could perhaps be a control for rate of samples added to give more choice to the user.