SFWRENG/MECHTRON 3K04

Deliverable 2

Deliverable 2 is 20% of your final grade

Due: November 24, 2025 - 8 a.m.

Introduction

There are 3 major components to this deliverable. The first part is continuing to build out pacemaker functionality. The second part will require you to complete an assurance case on the safety of your design. And finally all of these changes should be reflected in your updated documentation.

Prerequisites

These are the same as Deliverable 1.

1. Familiarity with the (natural language) PACEMAKER Requirements Document. Specific sections that you need to understand at this stage are:

1	Introduction
2.1 - 2.2	Overview/Components
3.1 - 3.2	DCM *(English only, 3.2.2, 3.2.3, 3.2.4 only 1 and 2, for 3.2.5 display egrams without markers)*
3.4	Pacing Pulses
3.5 - 3.6	Modes and States
4.7	*Real-Time Electrograms (egrams)*
5	Modes and Their Programmable Parameters
Appendix A	Programmable Parameters

   The document *srsVVI* rev 2 provides formal requirements for VVI mode. You should consult this document if you are unsure of the exact pacing requirements you will need in the future. Also, use this document for a *serial protocol for communications, and details regarding egrams (electrograms)*.

2. Familiarity with Pacemaker Microcontrolled Shield document to understand how **to operate the pacemaker shield. (New for Deliverable 2)**

3. Familiarity with software development on the hardware platform.

Part 1 - Pacemaker Design

This deliverable uses the model developed in the previous one and adds on to its functionality. You are required to implement three different functionalities:

1. Have a working implementation of the following modes using all meaningful programmable parameters: AOO, VOO, AAI, VVI, AOOR, VOOR, AAIR, VVIR.

2. Rate adaptive modes must track activity using the on-board accelerometer. The pulse rate (LRL) should increase once sufficient activity is detected. It is the responsibility of each group to design and justify the rate at which the LRL changes.

3. If you haven't already done so for Deliverable 1 1, all the modes must be integrated into one model.

Demo scenario for Rate Adaptivity: The pacemaker will be shaken by hand to simulate various speeds of activity; walking, jogging, and running. The device's rate should change accordingly.

Part 2 - Assurance Case

As you have now designed a safety critical embedded system, the pacemaker, it is of critical importance that you be able to generate assurance in why your system is safe. This is referred to as an assurance case. For Deliverable 2 you are supposed to generate an assurance case for the safety requirements and specifications that you have implemented in your design.

1 Part 3 - DCM

1. Expand the DCM to include all required modes and parameters.

2. Implement serial communication to transmit and receive information between the DCM and the Pacemaker.

3. The system will be able to set, store, transmit programmable parameter data , and verify it is stored correctly on the Pacemaker device. A minimal set of the parameter data is specified in the *srsVVI rev 2* document in section 3.1); the complete set is in PACEMAKER, Table 7.

   The following requirement changes are introduced to reflect the specifications of our system and to facilitate demonstrative purposes. Accommodate these changes in your implementation.

Parameter	Programmable Values	Increment	Nominal	Tolerance
A or V Pulse Amplitude Regulated	Off, 0.1-5.0V	0.1V	5V	$\pm$ 12%
A or V Pulse Width	1-30 ms	1 ms	1 ms	1 ms
A or V Sensitivity	0-5V	0.1V	-	$\pm 2\%$

4. The system will be able to display egram data when the user chooses to do so (for either ventricle, atrium, or both). The DCM must receive the egram data from the Pacemaker over the serial communication link in order to display it.

5. Document how programmable parameters originate at the DCM and are implemented in the device. Show how you can ensure the parameters stored in the Pacemaker are what the doctor input on the DCM. Also justify your choice of the data types used to represent parameters data.

What is an egram?

An egram or Electrogram is a visual output of the electrical activity inside each chamber of the heart. For our hardware, pins VENT_SIGNAL and ATR_SIGNAL are electrically connected to the analog egram signals for each chamber.

Part 4 - Documentation

The Heartview testing environment allows you to build test cases by configuring natural atrial activity, ventricle activity, heart rate and AV delay to perform functional testing on the pacemaker. You are required to prove correctness in functionality and safety by designing sufficient test-sets for your implementation. When documenting the testing portion provide the following:

1. Purpose/Test Justification, very brief explanation

2. System Input

3. Expected Output

4. Actual Output

5. Result (Pass/Fail)

   For part 4 you need to update your documentation to reflect the steps that you have taken to implement the new requirement changes that you have received or generated. This should be a continuation of your previously submitted documentation. This **SHOULD NOT** be a new document. The same principles you applied to the documentation for Deliverable 1 still apply to Deliverable 2.

Bonus

When the pacemaker is performing the DDD operating mode both chambers should be receiving a pace. But remember, both chambers cannot pace at the same time! There must be an **AV delay** between their paces to allow the blood to flow from the atrium to the ventricle.

To complete the bonus you will have to implement two functionalities:

1. DDDR mode with the proper AV delay (don't forget to make it rate adaptive).

2. Inhibit only ventricular pacing when holding down the pushbutton (i.e. atrial functionality is unaffected). Once the pushbutton is released then ventricular pacing will appear.

Deliverables

You are required to submit documentation as a PDF format. The file(s) must be included in a .zip folder that is submitted to Avenue by any team member. Your code for the DCM and Simulink programs should be located in a code folder which must also be included in the .zip submission folder. The .zip submission folder should be of the format deliverable2_group#.zip (you will be able to see your group number on Avenue).

Documentation Folder	docs_group#
Simulink Folder	code_group#
DCM Folder	DCM_group#

Design Principles

This is a course on software development, and design principles are a huge part of being able to achieve dependable, safe and secure applications. As such, you should pay particular attention to the following design principles as you develop this project:

1. Separation of concerns – in particular, effective modularity

2. Your designs must be robust with respect to anticipated changes – apply information hiding.

3. High cohesion and low coupling of modules

What do you need to do?

1. All design and code pushed to your repository if you have one and submitted on Avenue in a zip file.

2. Demo in lab

Grading

1. Demo: **60 Marks**

Pacemaker/Simulink	25
DCM/GUI	25
Discussion	10

2. Documentation: **40 Marks**

3. Bonus: **10 Marks**