

Deliverable 1: Documentation Outline

Group #2

Fall 2025

Contents

1	Group Members	3
2	Part 1	3
2.1	Introduction	3
2.1.1	Purpose	3
2.1.2	Goals	3
2.1.3	Scope	3
2.2	Requirements	4
2.2.1	AOO Mode	4
2.2.2	VOO Mode	4
2.2.3	AAI Mode	4
2.2.4	VVI Mode	4
2.3	Design	5
2.3.1	Pacemaker	5
2.3.2	DCM	5
3	Part 2	5
3.1	Requirements Potential Changes	5
3.2	Design Decision Potential Changes	5
3.3	Module Description	5
3.4	Testing	6
3.5	GenAI Usage	6
4	General Notes	6

1 Group Members

Jeffrey Yueh	yuehj	400495097
Johnson Ji	jih21	400499564
Kelby To	tok13	400507403
Hongliang Qi	qih25	400493278

2 Part 1

2.1 Introduction

2.1.1 Purpose

The purpose of a pacemaker is to regulate and restore a normal heart rhythm in patients with cardiac disorders such as arrhythmias, bradycardia, and heart failure. The pacemaker accomplishes this by delivering small electrical pulses to the atria and ventricles to make the heart beat at the correct speed and pattern.

The system consists of two main components: the **Pacemaker** and the **Device Controller-Monitor (DCM)**. The Pacemaker handles sensing and pacing functions, while the DCM allows users to configure, monitor, and manage the settings of the Pacemaker.

2.1.2 Goals

The main goal of Deliverable 1 is to design and implement the foundational components of the Pacemaker and DCM. Specifically:

- **Pacemaker:** Create stateflow models for AOO, VOO, AAI, and VVI modes with parameters specified in the Deliverable 1 document.
- **DCM:** Develop an interface that enables user registration and login, displays pacing modes, and allows input and storage of modifiable parameters.
- **Documentation:** Provide a detailed document outlining the design process, decisions, implementation, and testing procedures for the Pacemaker and DCM.

2.1.3 Scope

Deliverable 1 focuses on developing the initial components of the Pacemaker and DCM. This includes:

- Creating stateflow models for the AOO, VOO, AAI, and VVI pacing modes.
- Building the DCM interface with registration, login, and parameter customization functionality.

- Documenting the design and implementation process.

This deliverable does not include advanced features such as wireless communication, complex arrhythmia detection algorithms, or additional pacing modes beyond those specified. Hardware testing is limited to simulation and software verification environments.

2.2 Requirements

2.2.1 AOO Mode

Pacing: Atrial only

Sensing: None

Response to Sensing: None (asynchronous)

Behaviour: Delivers atrial pacing pulses at a fixed rate, regardless of intrinsic activity.

2.2.2 VOO Mode

Pacing: Ventricular only

Sensing: None

Response to Sensing: None (asynchronous)

Behaviour: Delivers ventricular pacing pulses at a fixed rate, regardless of intrinsic activity.

2.2.3 AAI Mode

Pacing: Atrial

Sensing: Atrial

Response to Sensing: Inhibited

Behaviour: Paces the atrium only if no intrinsic atrial activity is sensed within the programmed interval.

2.2.4 VVI Mode

Pacing: Ventricular

Sensing: Ventricular

Response to Sensing: Inhibited

Behaviour: Paces the ventricle only if no intrinsic ventricular activity is sensed within the programmed interval.

2.3 Design

2.3.1 Pacemaker

2.3.2 DCM

In this section, you should expand on design decisions based on the requirements. Be specific about your system design and how components interact.

- System architecture (major subsystems, hardware abstraction, pin mapping)
- Programmable parameters (rate limits, amplitudes, pulse widths, refractory periods, etc.)
- Hardware inputs and outputs (signals sensed, signals controlled)
- State machine design for each pacing mode (include diagrams or tables if applicable)
- Simulink diagram
- Screenshots of your DCM, explaining its software structure

Explicitly explain how each design decision maps directly to the stated requirements.

3 Part 2

3.1 Requirements Potential Changes

Identify requirements that may evolve in future deliverables (e.g., adding new pacing modes, communication capabilities, or additional parameters).

3.2 Design Decision Potential Changes

List design choices that may need revisiting (e.g., choice of libraries, interface design, architecture decisions).

3.3 Module Description

- Purpose of the component
- Key functions/methods (public vs. internal)
- Global or state variables (if any)
- Interactions with other components

3.4 Testing

Document test cases for each module. Each test case should include:

1. Purpose of the test
2. Input conditions
3. Expected output
4. Actual output
5. Result (Pass/Fail)

For the DCM, test registration and login, parameter input validation, mode selection, and data storage/retrieval. Depending on your system, you may need to test other components as well.

3.5 GenAI Usage

Provide a summary of any usage of GenAI tools in developing the model, DCM, or documentation. If no GenAI tools were used, state that explicitly.

4 General Notes

- This outline is based on the Deliverable 1 handout; ensure all required sections are included.
- Include screenshots of Simulink diagrams and the DCM interface where applicable.
- Ensure requirements are traceable to design and test cases.
- Keep content concise and clear.
- You may add other sections or modify this structure as needed, but these are the main expected components.