Pacemaker Firmware Documentation

MECHTRON 3K04 Assignment 1
Group 4
Andrew De Rango, Ali Hussin, Ethan Otteson,
Marco Tan, Rafael Toameh
Friday, October 25, 2024

Table of Contents

1 Scope

This document provides a comprehensive overview of the Pacemaker Firmware, an embedded system software designed to control the functionality and operation of pacemaker effectively. It outlines the firmware's requirements, functionality, implementation, design decisions, justifications, and testing made throughout the development process, enabling users and developers to have a clear understanding of the firmware's features and software architecture.

2 Terms and Definitions

Pacemaker Firmware - All software that is embedded on to the pacemaker microcontroller.

Off - Pacemaker mode in which no sensing or shocking occurs.

AOO - Pacemaker mode in which the atria are shocked rhythmically, without considering the natural atrial pulse.

VOO - Pacemaker mode in which the ventricles are shocked rhythmically, without considering the natural ventricular pulse.

AAI - Pacemaker mode in which the atria are shocked rhythmically unless inhibited by natural atrial pulses.

VVI - Pacemaker mode in which the ventricles are shocked rhythmically unless inhibited by natural ventricular pulses.

ARP - Atrial refractory period

VRP - Ventricular refractory period

BPM - Beats per minute

DCM - Device Controller-Monitor, see separate documentation for further details.

State - Block in Simulink which can set variables to value and acts as a state in a finite state machine.

Chart - Block in Simulink which contains states, acts as a finite state machine.

Subchart - Chart inside of another chart which contains states

HeartView - Software in which natural atrium and ventricle pulses can be created and natural and artificial pacemaker pulses can be visualized and monitored. Separate program provided to the project team.

Charging states - Variables that tell the firmware which pins should be active.

3 Requirements

The following outlines all the recognized Assignment 1 Pacemaker Firmware requirements. These requirements come from either the PACEMAKER requirements document provided as part of the project instruction package or from the project team directly. Not all requirements presented in the PACEMAKER document are included as many were considered out of scope for the implementation required for Assignment 1. This section includes all of the requirements given in natural language with many also provided in tabular expression for clarity, and a note regarding the expected future requirement changes.

3.1 Natural Language Requirements

The Pacemaker Firmware should abide by the following natural language requirements. These requirements are provided for ease of understanding and complement the formal specifications provided below. If the natural language requirements and formal specifications do not agree, the formal specifications are given precedence.

1. Pacemaker Firmware should implement the five programmable modes Off, AOO, VOO, AAI, & VVI.
2. The ability to toggle between modes should be available.
3. All modes except for Off should regulate the patient's heart rate with electrical pulses.
4. The following independent programmable parameters should be available:
    - 4.1. Pulse Amplitude should operate as follows:
        - 4.1.1. Two independent parameters, one for the atria and one for the ventricles.
        - 4.1.2. Pulse amplitude should be the voltage delivered to the heart when paced within a 12% tolerance.
        - 4.1.3. Atrial and ventricular pulse amplitudes should be between 0V and 5V.
    - 4.2. Pulse width should operate as follows:
        - 4.2.1. Two independent parameters, one for the atria and one for the ventricles.
        - 4.2.2. Pulse width should be the amount of time a pulse is delivered to the heart during a given pace within a 0.2 ms tolerance.
        - 4.2.3. Atrial and ventricular pulse width should be between 0.05 ms to 1.9 ms
    - 4.3. Refractory periods should operate as follows:
        - 4.3.1. Two independent parameters, one for the atria and one for the ventricles.
        - 4.3.2. No pulses should be delivered to the atria during the ARP and to the ventricles during the VRP following a sensed or delivered pulse of that chamber.
        - 4.3.3. Atrial and ventricular refractory periods should be between 150 ms and 500 ms.
    - 4.4. Lower rate limit should operate as follows:
        - 4.4.1. A single programmable parameter for both the atria and ventricles.
        - 4.4.2. If the heart rate is less than the lower rate limit then the heart is considered to be in bradycardia.

4.4.3. The lower rate limit should be between 30 to 175 BPM with a tolerance of 8 ms.

5. Off mode should operate as follows:
    5.1. Not sense or regulate any chambers of the heart.
6. AOO mode should operate as follows:
    6.1. Pace the atria periodically regardless of the patient's heart activity.
    6.2. Periodic pacing should be based on the lower rate limit provided.
    6.3. Does not affect the ventricles.
7. VOO mode should operate as follows:
    7.1. Pace the ventricles periodically regardless of the patient's heart activity.
    7.2. Periodic pacing should be based on the lower rate limit provided.
    7.3. Does not affect the atria.
8. AAI mode should operate as follows:
    8.1. Monitor the patient's heart rate for bradycardia and provide pacing if bradycardia is detected.
    8.2. If the sensed atrial heart rate is less than the lower rate limit then the mode should detect bradycardia and deliver a pacing pulse to the atria.
    8.3. Does not affect the ventricles.
9. VVI mode should operate as follows:
    9.1. Monitor the patient's heart rate for bradycardia and provide pacing if bradycardia is detected.
    9.2. If the sensed ventricular heart rate is less than the lower rate limit then the mode should detect bradycardia and deliver a pacing pulse to the ventricles.
    9.3. Does not affect the atria.

3.2 Formal Specifications

These formal specifications provided in tabular expression are intended to provide a clear, unambiguous, and precise description for many of the natural language requirements. The natural language requirement corresponding to a specific specification is noted in the title of the table. Not all requirements have a corresponding specification. If the natural language requirements and formal specifications do not agree, the formal specifications are given precedence.

Table 3.2.a: Formal specification of 3.1.1 describing the required modes.

	Pacemaker Firmware Modes				
Mode Behavior	**Off**	**AOO**	**VOO**	**AAI**	**VVI**
Pace Chamber	N/A	Atrium	Ventricle	Atrium	Ventricle
Sense Chamber	N/A	N/A	N/A	Atrium	Ventricle
Response to Sense	N/A	N/A	N/A	Inhibitor	Inhibitor

Table 3.2.b: Formal specification of 3.1.4 describing the behavior of the programmable parameter.

	Result
Condition	**Given Programmable Parameter**
Less than lower range limit	Round parameter to lower range limit
Within range	Parameter remains unchanged
Greater than upper range limit	Round parameter to upper range limit

Table 3.2.c: Formal specification of 3.1.5 describing the behavior of the Off mode.

	Result	
Condition	**Pace Atria**	**Pace Ventricles**
Heart Rate < Lower Rate Limit	No	No
Heart Rate = Lower Rate Limit	No	No
Heart Rate > Lower Rate Limit	No	No

Table 3.2.d: Formal specification of 3.1.6 describing AOO mode behavior.

	Result	
Condition	**Pace Atria**	**Pace Ventricles**
Heart Rate < Lower Rate Limit	Yes for atrial pulse width time and with atrial pulse amplitude	No
Heart Rate = Lower Rate Limit	Yes for atrial pulse width time and with atrial pulse amplitude	No
Heart Rate > Lower Rate Limit	Yes for atrial pulse width time and with atrial pulse amplitude	No

Table 3.2.e: Formal specification of 3.1.7 describing VOO mode behavior.

	Result	
Condition	**Pace Atria**	**Pace Ventricles**
Heart Rate < Lower Rate Limit	No	Yes for ventricular pulse width time and with ventricular pulse amplitude
Heart Rate = Lower Rate Limit	No	Yes for ventricular pulse width time and with ventricular pulse amplitude
Heart Rate > Lower Rate Limit	No	Yes for ventricular pulse width time and with ventricular pulse amplitude

Table 3.2.f: Formal specification of 3.1.8 describing AAI mode behavior.

	Result	
Condition	**Pace Atria**	**Pace Ventricles**
Heart Rate < Lower Rate Limit	Yes for atrial pulse width time and with atrial pulse amplitude	No
Heart Rate = Lower Rate Limit	No	No
Heart Rate > Lower Rate Limit	No	No

Table 3.2.g: Formal specification of 3.1.8 describing VVI mode behavior.

	Result	
Condition	**Pace Atria**	**Pace Ventricles**
Heart Rate < Lower Rate Limit	No	Yes for ventricular pulse width time and with ventricular pulse amplitude
Heart Rate = Lower Rate Limit	No	No
Heart Rate > Lower Rate Limit	No	No

3.3 Expected Changes

As the project develops there will inevitably be requirement changes, many of which can be foreseen. It is expected that more pacing modes will be added in the future which will cause several requirement changes. The requirements describing the behavior of the new modes will need to be added to differentiate behavior from the existing modes. Additionally, new programmable parameters will be required to define the characteristics of these new modes. It is expected that the upper rate limit parameter will be required. The current revision of the Pacemaker Firmware requirements does not include the upper rate limit as all of the current pacing modes are intended to treat bradycardia which only requires treatment when the heart rate is below a certain threshold. However, the upper rate limit may be required if additional modes intend to treat tachycardia or have varying pulse rates. As the Pacemaker Firmware will need to communicate with the DCM there are requirement changes expected related to this communication. These are the primary requirement changes expected moving forward with the project, however, there will likely be unexpected changes as well.

4 Design

The following outlines all design decisions that were made while creating the Pacemaker Firmware, including the functionality of every module, and the reasoning for the design decisions.

4.1 Overall Design

The Pacemaker Firmware created using MATLAB Simulink consists of three main sections, the input subsystem, the pacemaker controller chart, and the output subsystem. The input subsystem is where all the necessary inputs are taken in, checked to be within the correct range, and converted into the expected form. In the pacemaker control chart, the behaviour is decided based on the given pacemaker mode. The output subsystem takes values from both the input subsystem and the pacemaker control chart, specific pin behavior is then decided based on these input values.

The design of the Pacemaker Firmware ensures hardware hiding through the use of the 4 variable model by having direct access to board pins in only 2 locations, the input and output subsystem. This allows the charts to perform computation without direct access to the pins. This decomposition means that if an element related to the inputs is changed, such as connecting the Pacemaker Firmware to the DCM, only the input subsystem will need to be modified with the other sections remaining unchanged. Alternatively, if a different circuit board was used to represent the pacemaker with different sensing and pacing circuitry then the main logic of the pacemaker control chart could be left unchanged with the only modifications required being in the input and output subsystems.

The design does have some coupling in the output system however this is to minimize the duplication of code. The modules that do have problems of coupling are designed to work with the other modules as if the module they are coupled with were part of them. Because of the finite state machine design used in Simulink, only one of the coupled modules will ever be active at a time reducing the possible trouble that could be caused by the coupling. The finite state machine design also makes it as if the module using the code is working like a stand alone module, reducing the effects of coupling even more.

Simulink has increased safety compared to other programming languages due to the usage of finite state machines. A novel unexpected state could pose serious safety risks to the patient as the pacemaker could act unpredictably or incorrectly. Through the use of finite state machines, only specific states are representable, this allows only these states which are considered to be safe to have an effect on the user's heart.

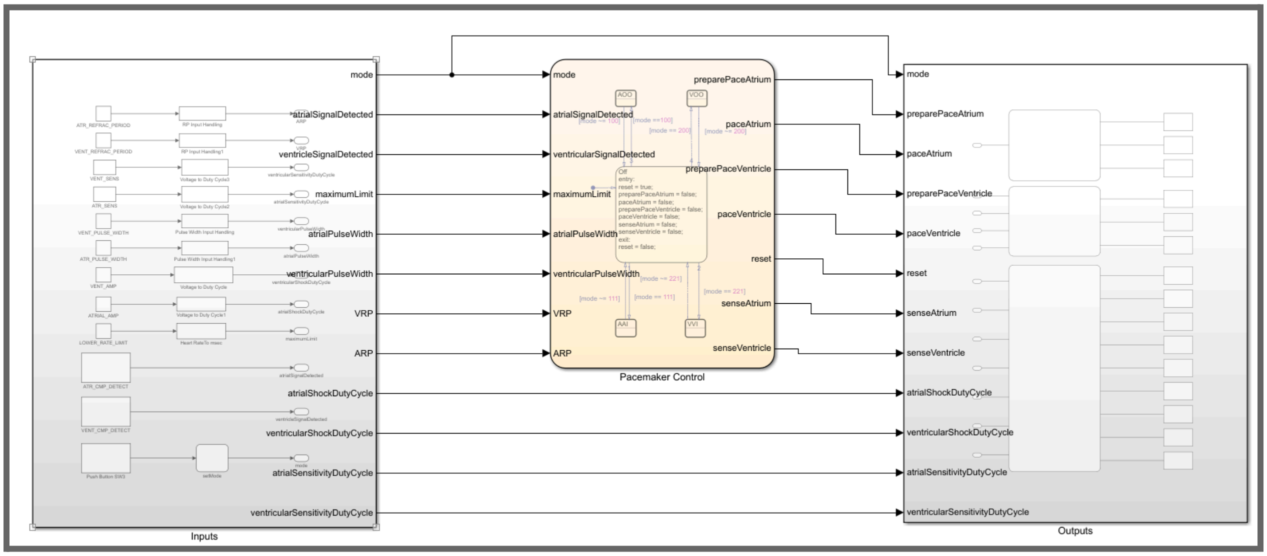

Figure 4.1.a: The entire Simulink model with the input subsystem on the left, pacemaker control chart in the middle, and output subsystem on the right.

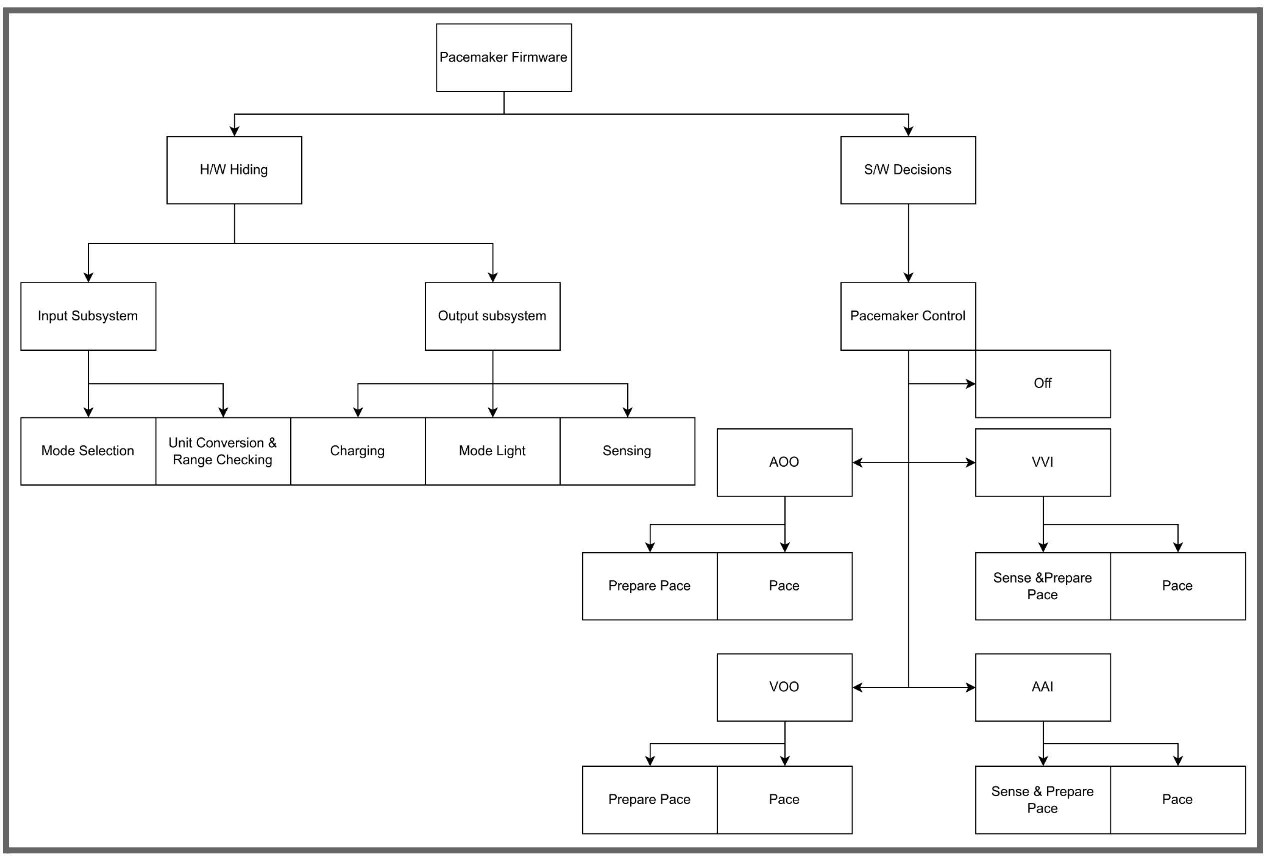

Figure 4.1.b: Module Decomposition of Pacemaker Firmware

At the very top level of the diagram is the Pacemaker Firmware, the app that controls all other decisions made. The second level is hardware hiding and the software decisions.

Hardware hiding is implemented through the input and output subsystem, software decisions are handled by the Pacemaker Control chart.

In the Input Subsystem, there is the mode selection, and Unit conversion & range checking, this is what occurs in the input subsystem. Inputs are converted to a more useful form if required and then they are range checked to ensure they are in the required ranges. In the Output Subsystem there is charging, mode light, and sensing. These are the 3 things that occur in this subsystem. They all deal with setting the pins to the correct values, pacing circuitry, lighting up an indicator to signal the active mode, and setting the sensing pins to detect natural heart rhythms.

In Pacemaker Control there are 4 main modes and Off. Under the 4 main modes is what takes place in them, for AOO and VOO Pace and Prepare Pace, under AAI and VVI are Sense & Prepare Pace and Pace.

Multiple naming conventions were used in the Pacemaker Firmware to differentiate how the variables are used. The different naming convention makes it so that at a glance a programmer can get information about the variable's usage based on the way the name is written. See Table 4.1.a below for the naming conventions used.

Table 4.1.a: Naming convention for variables in Simulink

Variable Type	**Naming Convention**
External inputs and output, and variables that connect to pins	ALL_CAPS_SNAKE_CASE
Subsystems and chart names	Normal Sentences With First Letter Of Each Word Uppercase
Chart input and outputs	camelCase
State titles	CamelCaseWithCapitalFirstLetter

4.2 Input Subsystem

This subsystem is responsible for handling all inputs to the Pacemaker Firmware. Any inputs that are anticipated to be connected to the DCM later and considered programmable parameters are hard-coded for this assignment before integration in the future. Any input interactions with pins are also handled in this subsystem. Many values are transformed into more useful forms or are range-checked to ensure they are within the proper bounds before being sent to other parts of the system for use. This subsystem is also where the pacemaker mode is selected and cycled through by use of a button on the board. Details for each input variable are discussed below.

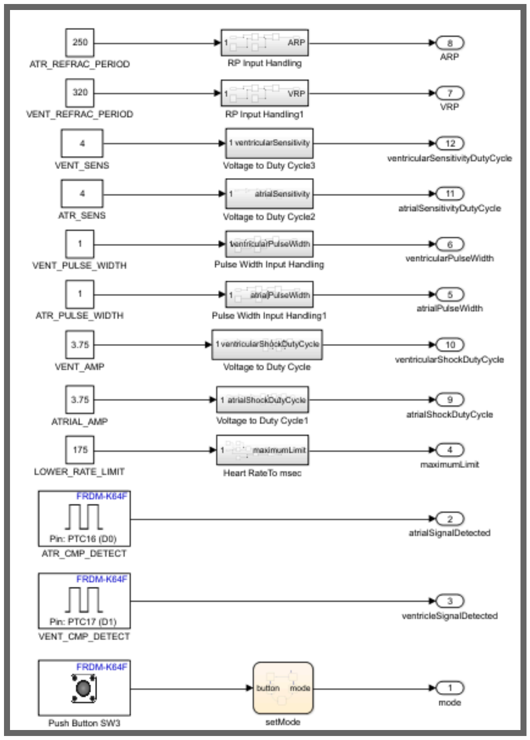

Figure 4.2.a: The Simulink input subsystem with programmable parameters at the top, the pins connected to the sensing circuitry in the middle, and the mode selection and button interactions at the bottom.

4.2.1 Programmable Parameters

As mentioned, the programmable parameters are hard coded as without DCM interfacing there is no way to set and change the required parameters. By collecting the parameters into one location they are easy to change and test. Each of the parameters is checked to ensure no issues arise from the parameters being outside of reasonable bounds and to ensure compliance with the requirements. In addition to being range checked, many of the parameters are either converted into a different form or have a correction factor applied. Details are covered in the following subsections.

4.2.1.1 Refractory Periods

The atrial refractory period is simply range checked to ensure the given value complies with the requirements. If it is out of range, the value will be rounded to the nearest value within the acceptable range.

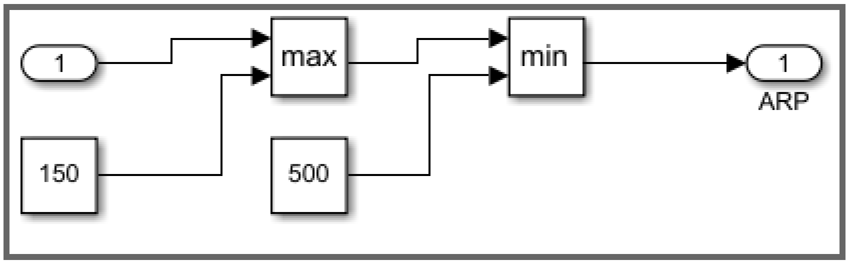

Figure 4.2.1.1.a: Subsystem which range-checks ATR_REFRAC_PERIOD.

Both the refractory period range checking subsystems operate in the same way with the same range. XXX_REFRAC_PERIOD is defined as a constant, in the associated subsystem, the value of XXX_REFRAC_PERIOD is range checked between 150ms and 500ms. The variable from the subsystem, ARP or VRP, is used in the pacemaker control chart.

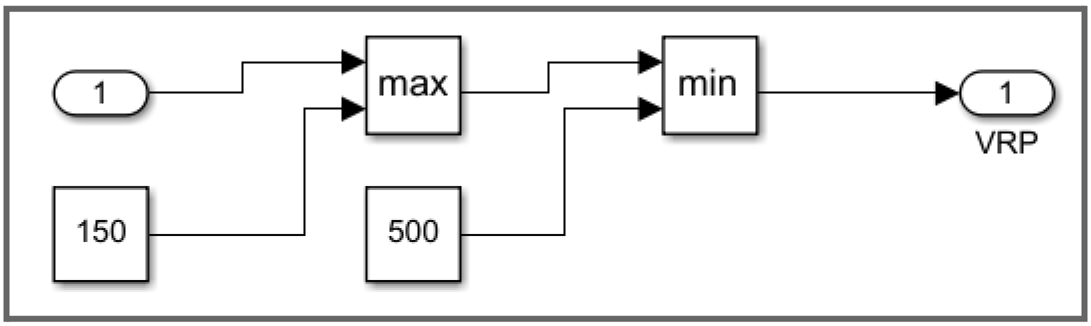

Figure 4.2.1.1.b: Subsystem which range checks VEN_REFRAC_PERIOD.

4.2.1.2 Sensitivities

While the sensitivity is not a required programmable parameter for this assignment, it was included with the rest of the parameters for simplicity and effective coding practice. The value of these parameters is not intended to be modified. Through informal testing, it was determined that setting the sensitivity too low would cause the sensing pins constantly reading noise and outputting a value of true. Alternatively, when the sensitivity was too high it would have trouble detecting the natural heart pulses, particularly at higher heart rates. Either way, this would lead to unexpected behaviour within the pacemaker modes which utilize the sensing circuitry. Using a value of 4V for the sensitivity seemed to mitigate any issues.

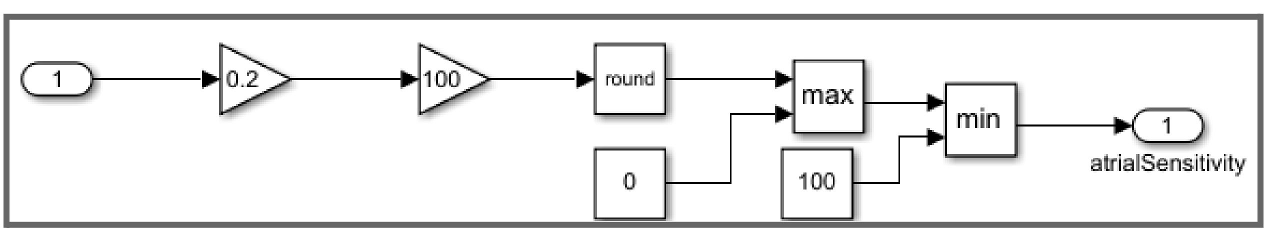

Figure 4.2.1.2.a: Subsystem which range checks and converts ATR_SENS.

The sensitivity range checking subsystem operates the same way for both variables. The value is first converted from a voltage into a duty cycle and then the bounds are checked to ensure it is a valid duty cycle. XXX_SENS is defined as a constant, in the associated subsystem, the value of XXX_SENS is range checked and converted to a number between 0 and 100. The variable from the subsystem, atrialSensitivity or ventricularSensitivity, is used in the output subsystem to control the sensitivity of the sensing circuitry.

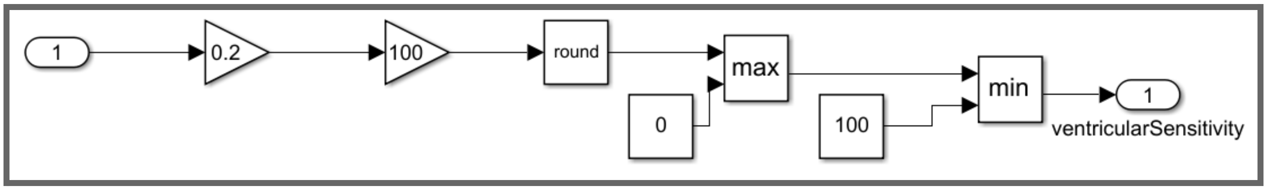

Figure 4.2.1.2.b: Subsystem which range checks and converts VENT_SENS.

4.2.1.3 Pulse Widths

The pulse width is the amount of time in milliseconds in which the capacitor delivering the pace to the heart will be given to discharge. This subsystem simply checks the bounds to ensure compliance with the requirements.

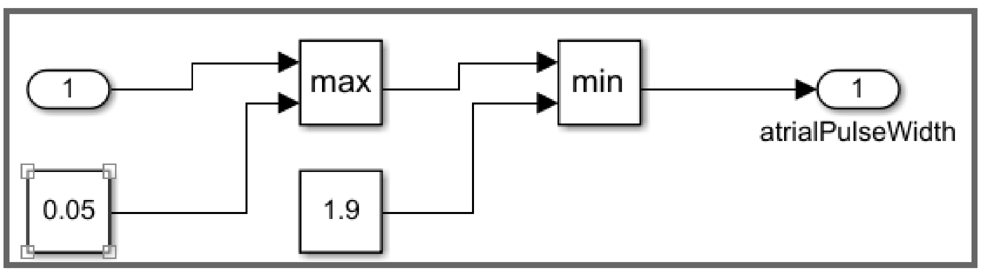

Figure 4.2.1.3.a: Subsystem which range checks ATR_PULSE_WIDTH.

Both pulse width variables are limited to the same range through identical subsystems. XXX_PULSE_WIDTH is defined as a constant. In the associated subsystem, the value of XXX_PULSE_WIDTH is range checked between 0.05ms and 1.9ms. The variable from the subsystem, atrialPulseWidth or ventricularPulseWidth, is used in the pacemaker control charge to determine how much time the pulse is delivered for in each mode.

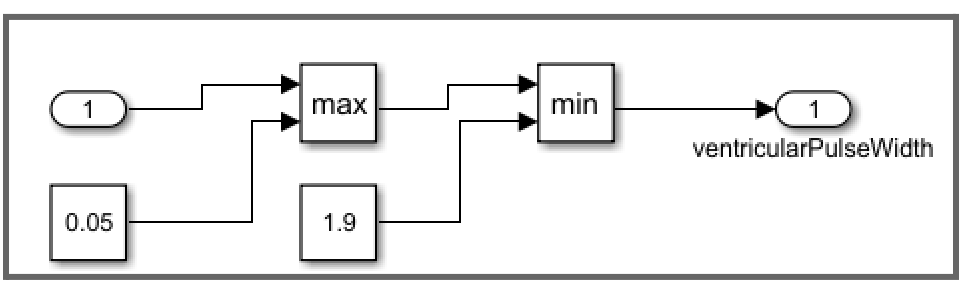

Figure 4.2.1.3.b: Subsystem which range checks VENT_PULSE_WIDTH.

4.2.1.4 Pulse Amplitudes

The range checking for the pulse amplitudes works similarly to the sensitivities. First, ensuring it is converted to a duty cycle, and then making sure the value is a valid duty cycle.

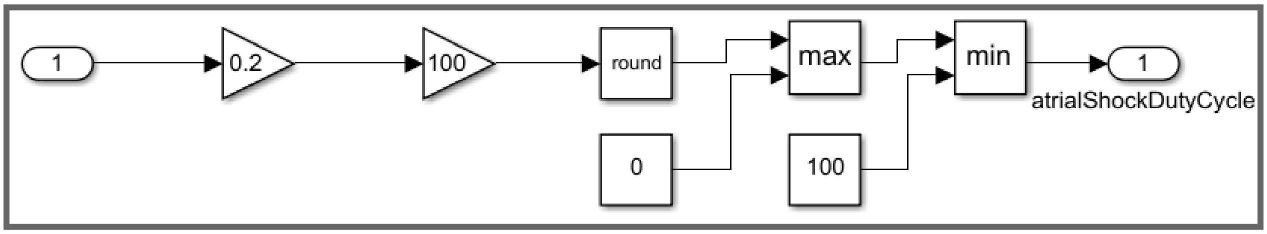

Figure 4.2.1.4.a: Subsystem which range checks and converts ATRIAL_AMP.

XXX_AMP is defined as a constant. In the associated subsystem, the value of XXX_AMP is converted to duty cycle by multiplying by 20 and is range checked between 0 and 100. The variable from the subsystem, atrialShockDutyCycle or ventricularShockDutyCycle, is used in the output subsystem to control the PWM reference used for pacing the heart.

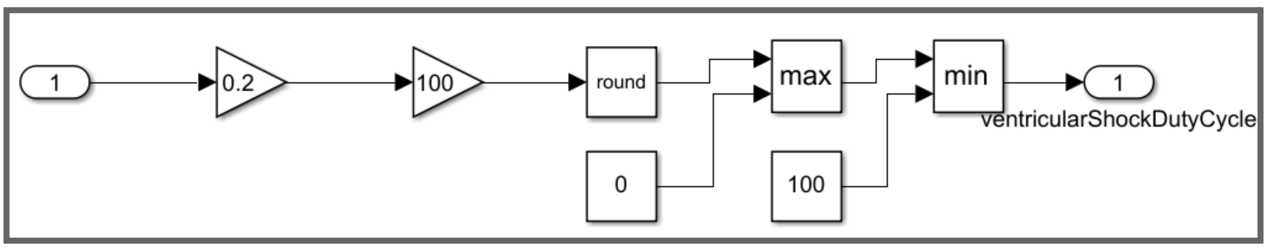

Figure 4.2.1.2.b: Subsystem which range checks and converts VENT_AMP.

4.2.1.5 Lower Rate Limit

The lower rate limit was considered to be the definition of bradycardia. It is the rate at which the AOO and VOO modes will pace and the rate at which the AAI and VVI modes are detecting to determine if they should pace or not. This parameter is given in beats per minute (BPM) and converted into pulse period in milliseconds. Initially, this was all that this subsystem did, however, it was recognized that due to hardware limitations causing the artificial pulsing to be slightly below the given rate and because HeartView is slightly faster than the given rate, the artificial and natural heart rates would operate at slightly different rates when set to the same value causing the plots to shift in phase relative to one another over time. To correct for this the period of the natural and artificial pulses was measured at eight different heart rates. These measurements at each heart rate were an average of the measurement of several periods. The difference between the natural and artificial heart rates was calculated and plotted as a function of heart rate. The trendline was determined for this data and was best matched by a power function, specifically $correction\, factor = 94.28 \cdot (heart\, rate)^{-0.686}$. This correction factor corrects the difference between the natural and artificial heart rate over the whole range to a maximum difference of about 1 ms, which is well within the 8 ms tolerance given by the requirements. This design was pursued as it allows for near-perfect alignment between what the Pacemaker Firmware and HeartView consider to be any given heart rate. This correction factor makes the lower rate limit the most complicated parameter processing for this system.

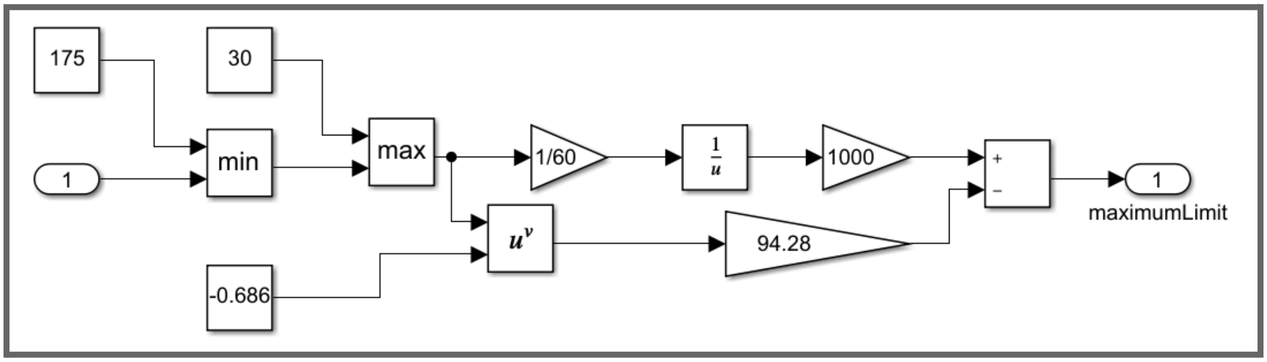

Figure 4.2.1.2.b: Subsystem which range checks and converts LOWER_RATE_LIMIT

The subsystem works as follows to ensure bound checking and correction are implemented. LOWER_RATE_LIMIT is defined as a constant in the associated subsystem, the value of LOWER_RATE_LIMIT is converted from bpm to milliseconds and range checked. The correction factor is also calculated in parallel and subtracted from the final value. The variable from the subsystem, maximumLimit, is used in the pacemaker control chart as the maximum amount of time the heart should go without a natural or artificial pulse.

4.2.2 Input Pins

Atr_CMP_DETECT and VENT_CMP_DETECT are direct readings from the pins D0 and D1, the readings are boolean values. The output from these pins is used in the pacemaker control chart, true means that a natural heart pulse is detected false means that no natural pulse was detected. To ensure effective use of the 4 variable model, the variables are converted to atrialSignalDetected and ventricularSignalDetected before leaving the input system so that if they are ever changed to a different method for detecting a natural pulse in the future only this section of code would need to be modified.

4.2.3 Mode Selection

Within the Simulink model, the modes are defined by integer values. ‘A’ is denoted by a 1, ‘V’ by a 2, and ‘I’ by a 1. This assigns each mode to a three digit number for use in the system. Off is just 0 because this is equivalent to 000.

Table 4.2.3.a: Modes and the associated number

Mode	Off	AOO	VOO	AAI	VVI
Associated number	0	100	200	111	221

Currently, a button is being used to change the mode. This allows easy switching of the modes and easier testing without having to recompile the code. When the button is pressed, the mode will change. On start the mode is set to Off. With each press, it will cycle to the next mode, Off, AOO, VOO, AAI, and VVI in that order. There is a one second wait between switch modes to ensure that when holding the button down, it does not rapidly switch between each mode and rather switches slowly, giving the user more control. The mode output is used in both the pacemaker control chart to determine the mode and the output subsystem to determine the LED colour as indication of the mode to the user. The variable name outside the subsystem is mode.

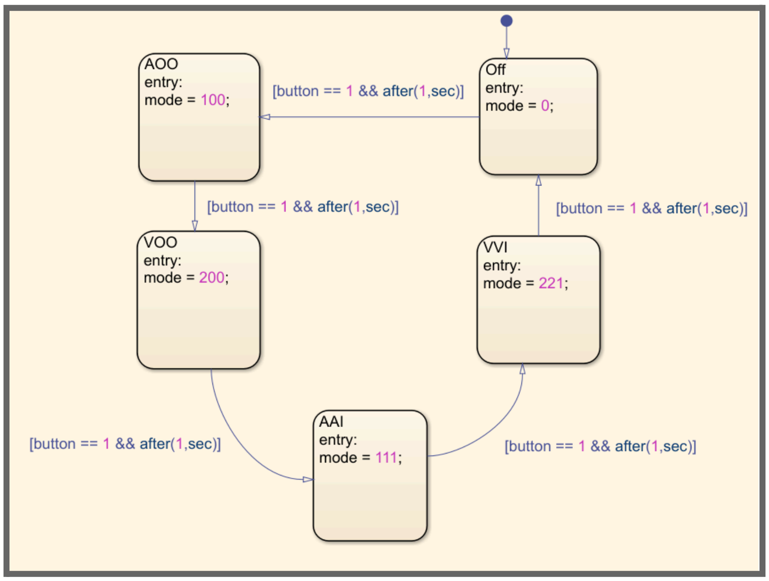

Figure 3.2.3.a: Inside of the mode switching chart which cycles through the modes with the button.

4.3 Pacemaker Control Chart

The pacemaker control chart is the location where the logic of each mode is controlled and the logic of switching between the modes. The system functions by having the Off mode in the center and a branch to and from each of the states. This means that the states can not switch between each other directly and rather have to pass through. This is beneficial in two ways as it prevents unexpected behaviour and eliminates the need to connect each mode to one another which would become a mess once the number of modes grows. The state for a specific mode will be entered once the mode is set to that mode and exited once it is no longer equal to that mode.

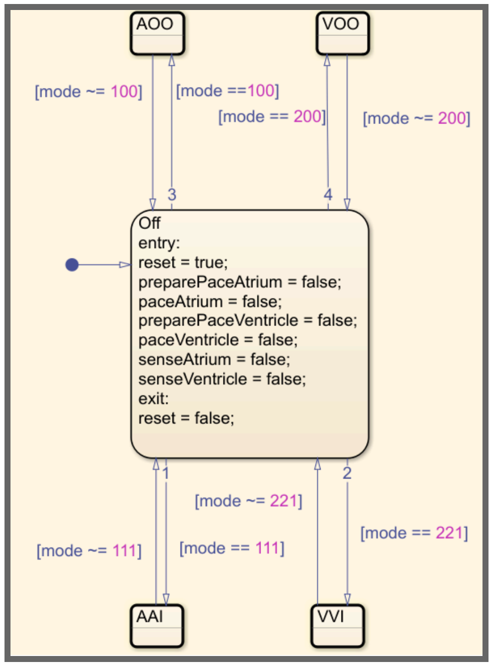

Figure 4.3.a: Inside of Pacemaker Control Chart

Within the Off mode are the many variables that will control states and pin behaviour in the output system. They are as follows:

Reset - Causes pin control to return to the off mode.
paceAtrium - Will turn on the necessary pins to pace the atrium.
paceVentricle - Will turn on the necessary pins to pace the ventricle.
preparePaceAtrium - Will turn on the necessary pins to prepare an atrium pace.
preparePaceVentricle - Will turn on the necessary pins to prepare an ventricle pace.
senseAtrium - Will turn on the necessary pins to sense the natural atrial pulses.
senseVentricle - Will turn on the necessary pins to sense the natural ventricular pulses.

4.3.1 AOO Mode

The AOO subchart can only be active if the AOO mode is active. The subchart contains 2 states, the Prep state and the Shock state. The Prep state turns on the preparePaceAtrium charging state on entry. When exiting the state, preparePaceAtrium is turned off. The Shock state turns on the paceAtrium charging state on entry. When exiting the state, paceAtrium is turned off. To change the state from Prep to Shock, maximumLimit minus atrialPulseWidth amount of time must take place. To change the state from Shock to Prep, atrialPulseWidth amount of time must take place. Remember that the maximumLimit variable is equal to the pulse period, subtracting by the pulse width ensures that the period is met each cycle and that the mode is operating at the desired heart rate. The AOO subchart will be exited from when the mode is no longer AOO.

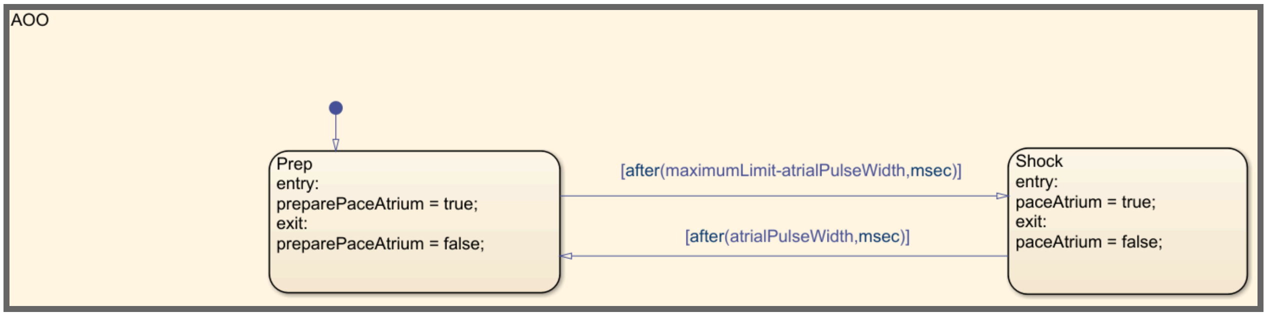

Figure 4.3.1.a: Inside of AOO Subchart

4.3.2 VOO Mode

The VOO mode operates identically to AOO except pacing the ventricle instead. The VOO subchart can only be active if the VOO mode is active. The subchart contains 2 states, the Prep state and the Shock state. The Prep state turns on the preparePaceVentricle charging state on entry, which turns on all pins required for preparing to pace the ventricle when exiting the state, preparePaceVentricle is turned off. The Shock state turns on the paceVentricle charging state on entry, which turns on all pins required to pace the ventricle when exiting the state, paceVentricle is turned off. To change the state from Prep to Shock, maximumLimit minus ventricularPulseWidth amount of time must take place. To change the state from Shock to Prep, ventricularPulseWidth amount of time must take place. Remember that the maximumLimit variable is equal to the pulse period, subtracting by the pulse width ensures that the period is met each cycle and that the mode is operating at the desired heart rate. The VOO subchart will be exited from when the mode is no longer VOO.

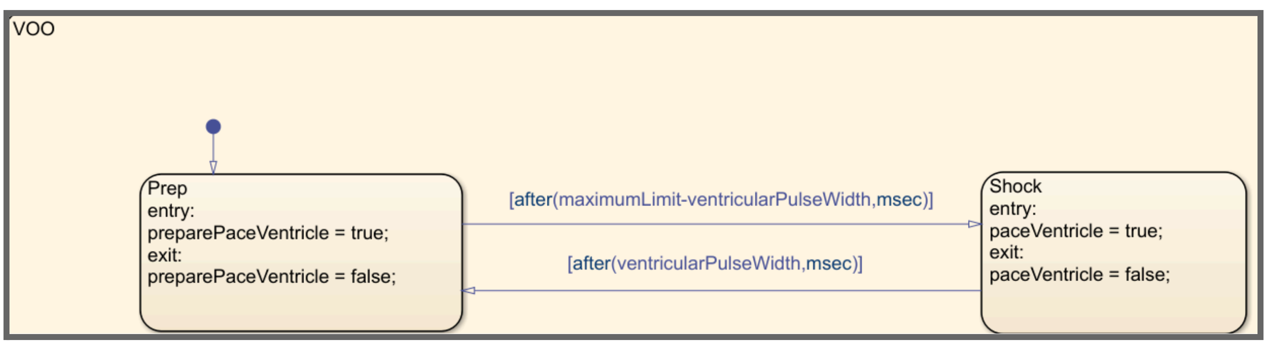

Figure 4.3.2.a: Inside of VOO Subchart

4.3.3 AAI Mode

The AAI subchart is active if AAI mode is active. The subchart contains 3 states, the SenseDelay, SenseNoDelay, and the Pace state. The Sensing states turn on the preparePaceAtrium and the senseAtrium charging states on entry, when exiting, preparePaceAtrium is turned off. The Shock state turns on the paceAtrium charging state on entry, when exiting the state, paceAtrium is turned off. In both sensing states a maximum of the lower rate limit (converted to maximumLimit) amount of time of sensing takes place, if no natural pulses occur in that period, the pacemaker sends a pulse. When in the SenseNoDelay state if a natural pulse is detected the state will change to the SenseDelay state. When in the SenseDelay state a period of 100ms must be passed to loop back into itself. To change the state from Pace to SenseNoDelay, atrialPulseWidth amount of time must take place. This ensures that the pace lasts for the required amount of time. The AAI subchart will be exited from when the mode is no longer AAI.

This design decision was made to avoid either of the following cases. If only SenseDelay was an option then consider the case where the pacemaker pulses and immediately afterward the heart pulses. In this case, the natural pulse may not be detected and the next artificial pulse may be delivered too soon or unnecessarily. This raises the question of whether the 100 ms wait time on the cycle is even required. However, consider a case with a longer pulse rate. Without the counter, this longer natural pulse would continue to be sensed until its falling edge which would mean that the next possible artificial pulse would not be the lower rate limit later but rather the lower rate limit plus the duration of the natural pulse. Both of these cases caused problems for previous iterations of the Pacemaker Firmware, hence the current design which is able to detect natural pulses immediately following an artificial pulse while also only detecting the rising edge of the natural pulse and not the whole thing. The 100 ms wait time was settled on as it is sufficiently large to avoid detecting the same natural pulse multiple times while being smaller than the period between pulses of the fastest possible heart rate. Additionally, unlike AOO the wait before pacing is maximumLimit without subtracting the pulse width. This was done intentionally, as the objective is to sense a heart rate of the lower rate limit and it is impossible to both sense for and pace at the lower rate limit. It was decided that sense time was more important. Additionally, we are still within the tolerance of the lower rate limit as discussed earlier it is a maximum of about 1 ms off and adding the maximum of a 1.9 ms pulse width to this renders the error still within an 8 ms tolerance per the requirements.

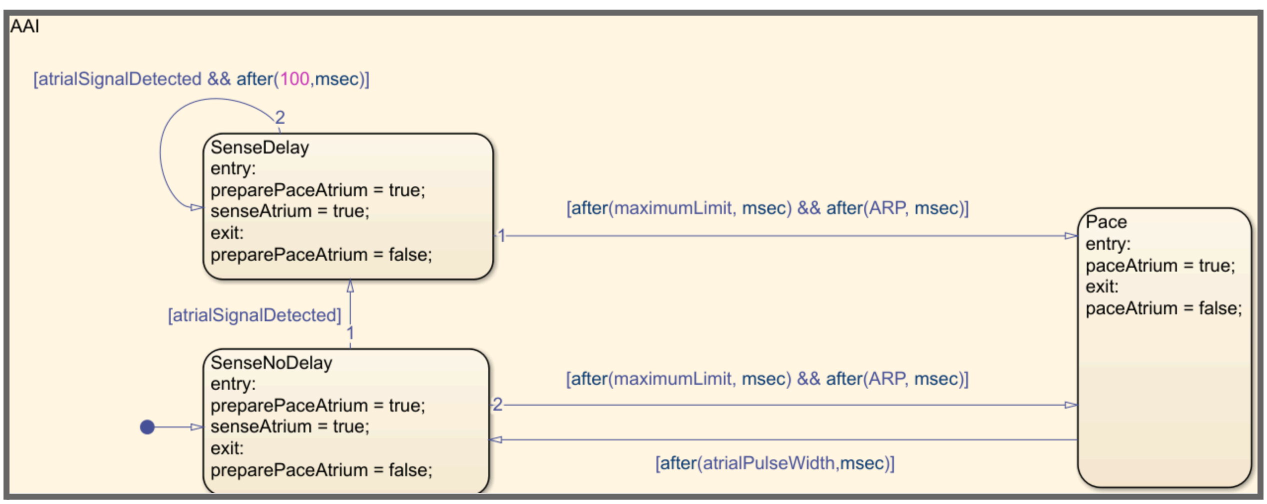

Figure 4.3.3.a: Inside of AAI Subchart

4.3.4 VVI Mode

The VVI subchart works exactly the same as the AAI subchart but using variables associated with the ventricle instead of the atrium, these variables have the same names but use ventricular instead of atrial and ventricle instead of atrium. All discussion of how this mode works and the justification of design decisions can be found in 4.3.3 AAI Mode.

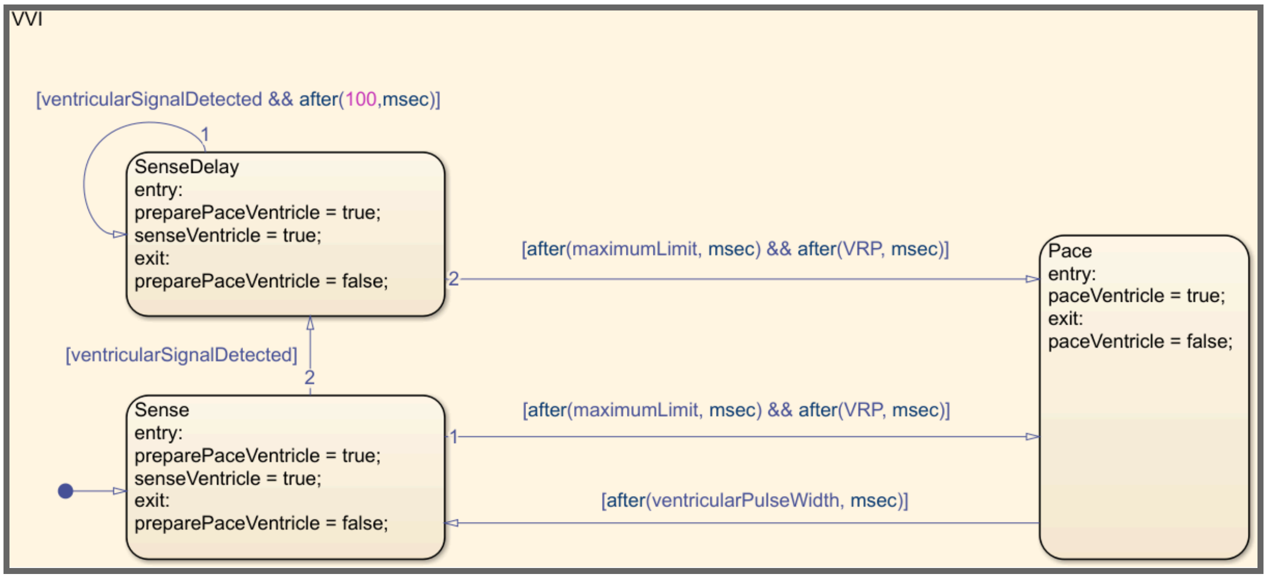

Figure 4.3.4.a: Inside of VVI Subchart

4.4 Output Subsystem

The goal of the output subsystem is to handle all outputs and ensure that only valid outputs occur. This is done using inputs from the previous two systems. This again is in support of the 4 variable model as the interaction with hardware is separate from the decisions being made based on the mode and operation. If the hardware was to change, for instance using an actual pacemaker the sensing and pacing circuitry would likely be different with a different number of switches and capacitors which may need to be interacted with in different orders. Rather than having to go and implement those changes throughout the model multiple times it would need to only be done once in the output system. The layout of the output system also helps to minimize code duplications as every mode uses the same pacing and sensing code.

4.4.1 LED Pin Control

As previously mentioned, the mode can be switched during runtime using the button on the board. However, it may be difficult to remember which mode is being tested. To solve this issue the LED on the pacemaker board is used to indicate the mode. The operation of this is controlled by the LED Pin Control chart within the output system. Inside of the chart there are states that associate a mode to a colour using the variables r, b, and g, see Table 4.4.1.a.

Table 4.4.1.a: Pacemaker modes and associated LED colors.

Mode	Off	AOO	VOO	AAI	VVI
Associated number	White	Red	Green	Purple	Blue

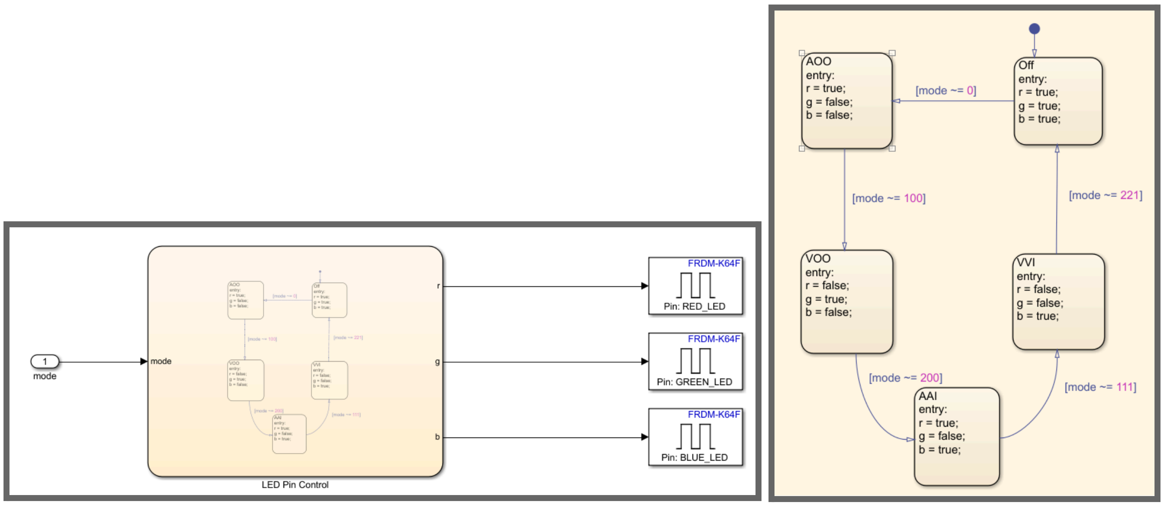

Figure 4.4.1.a: Inside and outside of LED Pin Control Chart

4.4.2 Sensing Pin Control

The Sensing Pin Chart is used to turn on pins required for sensing natural heartbeats. The Sensing Pin Control chart, the inputs to this chart are the variables from the pacemaker control chart senseAtrium and senseVentricle, and the inputs atrial and ventricular sensitivity duty cycle. The outputs from the chart are the values of the 3 pins required for sensing natural heartbeats. When the charging state is senseAtrium the SenseAtrium state is active. When the charging state is senseVentricle the SenseVentricle state is active. When neither is active, no sensing is done. See Table 4.4.2.a to see the pins that states turn on.

Table 4.4.2.a: Pins turned on by the charging states involved in sensing.

State	Pin Values
SenseAtrium	FRONT_END_CTRL = true VENT_CMP_REF_PWM = 0 ATR_CMP_REF_PWM = atrialSensitivityDutyCycle
SenseVentricle	FRONT_END_CTRL = true VENT_CMP_REF_PWM = ventricularSensitivityDutyCycle ATR_CMP_REF_PWM = 0
Default	FRONT_END_CTRL = true VENT_CMP_REF_PWM = 0 ATR_CMP_REF_PWM = 0

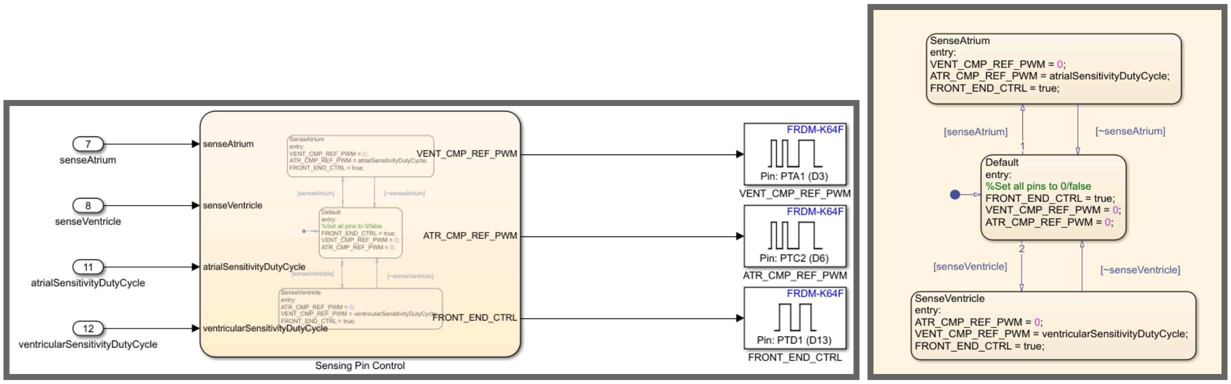

Figure 4.4.2.a: Inside and outside of Sensing Pin Control chart.

4.4.3 Pacing Pin Control

The Pacing Pin Control chart turns on pins required for pacing both the atrium and ventricles. The inputs to the chart are the variables from the pacemaker control chart: both preparePaceAtrium/Ventricle, both PaceAtrium/Ventricle, and reset, and both atrial/ventricularShockDutyCycle. The appropriate state will only be active when the associated charging state is on due to the pacemaker control chart deciding behaviour for the modes.

The way the chart is designed allows code reuse to be maximized, as both AOO and AAI use the states for preparing and pacing the atrium and both VOO and VVI use the states for preparing and pacing the ventricle.

Another approach is to have the charging states directly linked to the states to pace and sense this would decrease coupling but would increase code duplication drastically with VOO and VVI both having their own set of the preparePaceVentricle and paceVentricle. However, as only 1 of these states is active at a time there is no need to have multiple copies of these states. See Table 4.4.3.a to see the pins that states turn on.

Table 4.4.3.a: Pins turned on by the charging states involved in pacing.

State	Pin Values
PreparePaceAtrium	VENT_PACE_CTRL = false ATR_PACE_CTRL = false PACING_REF_PWM = atrialShockDutyCycle PACE_CHARGE_CTRL = true PACE_GND_CTRL= true Z_VENT_CTRL = false Z_ATR_CTRL = false ATR_GND_CTRL = true VENT_GND_CTRL = false
PreparePaceVentricle	ATR_PACE_CTRL = false VENT_PACE_CTRL = false PACING_REF_PWM = ventricularShockDutyCycle PACE_CHARGE_CTRL = true PACE_GND_CTRL= true

	Z_ATR_CTRL = false Z_VENT_CTRL = false VENT_GND_CTRL = true ATR_GND_CTRL = false
PaceAtrium	PACE_CHARGE_CTRL = false PACE_GND_CTRL= true VENT_PACE_CTRL = false VENT_GND_CTRL = false Z_VENT_CTRL = false Z_ATR_CTRL = false ATR_PACE_CTRL = true ATR_GND_CTRL = false PACING_REF_PWM = 0
PaceVentricle	PACE_CHARGE_CTRL = false PACE_GND_CTRL= true ATR_PACE_CTRL = false ATR_GND_CTRL = false Z_ATR_CTRL = false Z_VENT_CTRL = false VENT_GND_CTRL = false VENT_PACE_CTRL = true PACING_REF_PWM = 0
Default	VENT_PACE_CTRL = false ATR_PACE_CTRL = false PACING_REF_PWM = 0 PACE_CHARGE_CTRL = false PACE_GND_CTRL= false Z_VENT_CTRL = false Z_ATR_CTRL = false ATR_GND_CTRL = false VENT_GND_CTRL = false

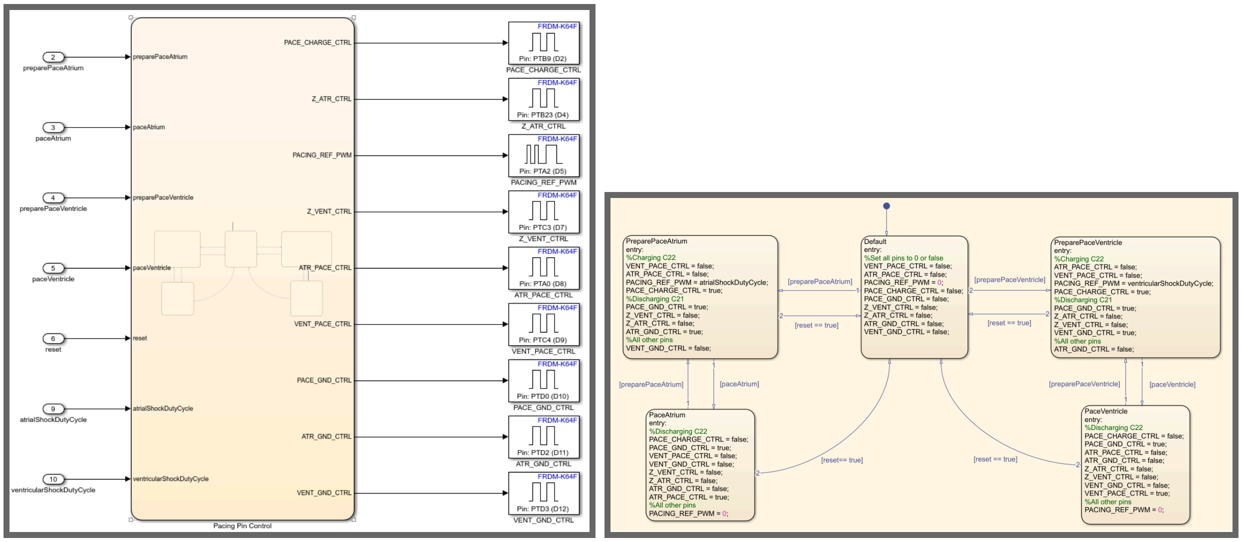

Figure 4.4.3.a: Inside and outside the Pacing Pin Control chart.

4.5 Expected Changes

4.5.1 Input Subsystem Changes

- In the input subsystem most constants will be changed into values uploaded from the DCM, with the current way the Simulink is set up making the input constants into serial input should be very simple.
- Mode selection will need to be changed to use the DCM instead of the on board button which will likely also mean that the mode selection chart may become obsolete.
- More modes will need to be created meaning that additional variables may be required.

4.5.2 Pacemaker Chart Changes

- With more modes being added, more inputs to the chart will be included.
- With more modes being added, more charging states may need to be created.
- More modes will be created and as such more subcharts will need to be added for each mode.

4.5.3 Output Subsystem Changes

- With more modes being added, more inputs to the subsystem will be included.
- With more modes being added, more charts may need to be created.
- Values will need to be transmitted to the DCM.

5 Validation and Verification

The following outlines the justification of Pacemaker Firmware requirements and all testing of the Pacemaker Firmware.

5.1 Validation

The requirements that are created for the Pacemaker Firmware are based on the PACEMAKER requirement documentation given to the project team. Based on sections 3.4 which specifies programmable parameters of pulse amplitude and pulse width, 3.5 lists required modes, and 5 has a table of required parameters for the modes. Section 3.6.5 gives values for the refractory period used in the requirements. Table A lists value ranges for lower rate limits, refractory periods, pulse amplitudes, and pulse widths. However, these values are not specific to the hardware that we are using, causing the values to possibly be slightly different for the requirements than the values that are in the document.

The requirements were generated from these formal specifications making the needed modifications when necessitated by the specifics of this assignment. Considering the modifications required to complete the assignment, comparing the requirements with the formal specifications it can be said that the requirements are valid for creating the Pacemaker Firmware and what was created is in relative compliance with the formal specifications.

5.2 Verification

The following provides verification of each of the requirements outlined separately in this document. These tests ensure that the Pacemaker Firmware operates as intended and meets all requirements. Requirements are referred to by the numbering present in section 3 Requirements.

5.2.1 Implicit Testing

Requirements 1-3 do not require formal testing as the design directly implements these requirements. As previously discussed, the five modes Off, AOO, VOO, AAI, & VVI have been added and can be cycled using the button with the LED colour indicating the current mode. Specific testing for each mode is below. Additionally, it can be inferred that the microcontroller representing the heart is being shocked using electrical pulses given that the paces are generated by charging and discharging a capacitor. The remaining requirements are all far more specific and testable and as such formal tests and their results follow.

5.2.2 Programmable Parameter Testing

The programmable parameters outlined by requirement 4 are tested below. These tests were done using the various pacemaker modes with different input parameters. Results were measured using the HeartView monitoring tool. Though many of these tests are unsuccessful, it is believed that the programmable parameters are implemented correctly and hardware errors and limitations are causing many of the discrepancies.

5.2.2.1 Refractory Periods

The following test cases described in Table 5.2.2.1.a are intended to test the refractory period programmable parameters. These tests are conducted using the HeartView monitoring tool and the AAI and VVI modes. The other Pacemaker Firmware modes do not need to be included in the test as the refractory period does not affect these modes. Interesting test results are further discussed following the table.

Table 5.2.2.1.a: Refractory period test cases verifying requirement 4.1.

Test	Expected Result	Pass/Fail	Comment
No natural atrial pulsing when ARP is set to 500 ms and the lower rate limit is set higher than 120 BPM	Atrial artificial pacing at approximately 120 BPM	Pass	Indicates that pacing is inhibited following the refractory period.
No natural atrial pulsing when ARP is set to 1000 ms and the lower rate limit is set higher than 120 BPM	Atrial artificial pacing at approximately 120 BPM	Pass	Indicates that the upper range limit is functioning and limiting ARP to 500 ms.
No natural ventricular pulsing when VRP is set to 500 ms and the lower rate limit is set higher than 120 BPM	Ventricular artificial pacing at approximately 120 BPM	Pass	Indicates that pacing is inhibited following the refractory period.
No natural ventricular pulsing when VRP is set to 1000 ms and the lower rate limit is set higher than 120 BPM	Ventricular artificial pacing at approximately 120 BPM	Pass	Indicates that the upper range limit is functioning and limiting VRP to 500 ms.
Natural atrial pulsing is higher than 120 BPM when ARP is set to 500 ms and the lower rate limit is set higher than the natural heart rate	No pacing output	Pass	Indicates that the refractory period is not inhibiting natural pulse sensing.

Natural ventricular pulsing is higher than 120 BPM when VRP is set to 500 ms and the lower rate limit is set higher than the natural heart rate	No pacing output	Pass	Indicates that the refractory period is not inhibiting natural pulse sensing.

As indicated by all of these tests the refractory periods are functioning properly. The upper range limit for the refractory periods is verified as an input value of 500 ms and 1000 ms renders the same resulting maximum refractory period of 500 ms. The lower range limit is not testable as the heart rate cannot reach a high enough value in which the period between pulses is less than 150 ms and the minimum refractory period would make a difference. When the refractory period is greater than the lower rate limit, it limits pacing to the refractory period as intended. Finally, during our project demonstration, it was instructed that the Pacemaker Firmware should continue to sense natural pacing during the refractory period which is verified as being true. As all tests verify the expected behavior, the refractory period programmable parameters are considered to be correct.

5.2.2.2 Sensitivities

Sensitivity is not a required adjustable parameter for this assignment. As such it has been dialed into a specific value to work most effectively to ensure it can detect the natural pulses without detecting noise. No further testing for the sensitivities is provided.

5.2.2.3 Pulse Widths

The following test cases outlined in Table 5.2.2.3.a are intended to test the pulse width programmable parameters. These tests are conducted using the HeartView monitoring tool and the AOO and VOO modes. These modes are used because they allow for testing of both the atrial and ventricular pulse widths and utilize identical pacing code to the AAI and VVI modes. Pulse width is measured as the time between the rising edge of the pulse and the initial falling edge as it crosses 0V. Interesting test results are further discussed following the table.

Table 5.2.2.3.a: Pulse width test cases verifying requirement 4.2.

Test	Expected Result	Pass/Fail	Comment
The pulse width is set below 0.05 ms	The pulse width is the same as setting it to 0.05 ms	Pass	Tested with both AOO and VOO. Indicates that the lower range limit is functioning and limiting the pulse width to 0.05 ms.
The pulse width is set above 1.9 ms	The pulse width is the same as setting it to 1.9 ms	Pass	Tested with both AOO and VOO. Indicates that the upper range limit is functioning and limiting the pulse width to 1.9 ms.

The atrial pulse width is set between 0.05 ms and 1.9 ms	The atrial pulse width measured in HeartView is approximately its set value	Fail	Tested at 0.05, 0.1, 0.4, 1, 1.25, 1.5, 1.7, and 1.9 ms. All measured to either approximately 1.0 or 1.5 ms.
The ventricular pulse width is set between 0.05 ms and 1.9 ms	The ventricular pulse width measured in HeartView is approximately its set value	Fail	Tested at 0.05, 0.1, 0.4, 1, 1.25, 1.5, 1.7, and 1.9 ms. All measured to either approximately 1.0 or 1.5 ms.

This set of tests yielded interesting and unexpected results. The first two tests passed because they yielded the same measured values as the parameters they were expected to be rounded to. However, the other two tests both failed as all the parameters seemed to produce a measured value of either 1.0 or 1.5 ms, whichever was closer. This is likely due to hardware errors and limitations regarding the time required to discharge the capacitor. Assuming this is what is causing the error, changing the parameter does slightly modify the pulse width meaning that the parameter is doing its function and it is likely implemented correctly.

5.2.2.4 Pulse Amplitudes

Formal testing of the pulse amplitudes is difficult due to large amplitude inconsistencies between pulses. This is considered to be a limitation of either the hardware or HeartView monitoring tool and as such will eliminate the possibility for formula testing. It can be shown that an amplitude of 4V is relatively larger than an amplitude of 1V. However, there is no way to repetitively and empirically test this. Considering this issue is a result of hardware error and changing the pulse amplitude does cause a visible relative change within HeartView, the pulse amplitude programmable parameters are considered correct.

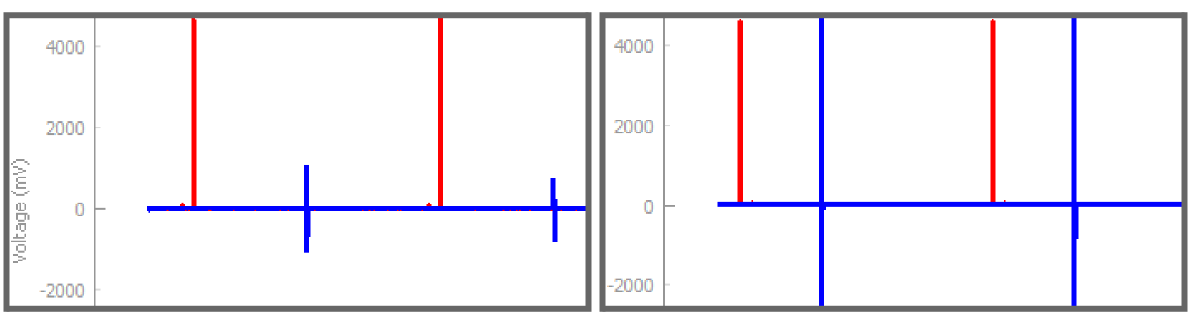

Figure 5.2.2.4.a: Comparison of the 1V (left) and 4V (right) pulse amplitude in HeartView.

5.2.2.5 Lower Rate Limit

The following test cases outlined in Table 5.2.2.5.a are intended to test the lower rate limit programmable parameter. These tests are conducted using the HeartView monitoring tool and the AOO and VOO modes. These modes are used because they are not affected by the natural pulsing of the heart and the natural and artificial beats can be set to the same frequency to observe any changes in phase shift over time. Interesting test results are further discussed following the table.

Table 5.2.2.5.a: Lower rate limit test cases verifying requirement 4.4.

Test	Expected Result	Pass/Fail	Comment
Natural pulsing is set to 30 BPM and the lower rate limit is set below 30 BPM.	Natural and artificial pulsing remain at the same phase shift over time.	Pass	Tested with both AOO and VOO. Indicates that the lower range limit is functioning and limiting the lower rate limit to 30 BPM.
Natural pulsing is set to 175 BPM and the lower rate limit is set above 175 BPM.	Natural and artificial pulsing remain at the same phase shift over time.	Pass	Tested with both AOO and VOO. Indicates that the upper range limit is functioning and limiting the lower rate limit to 175 BPM.
Natural atrial pulsing is set between 30 and 175 BPM and the lower rate limit is set to the same value.	Natural and artificial atrial pulsing remain at the same phase shift over time.	Pass	Tested at 30, 31, 47, 50, 60, 70, 90, 120, 150, 170, and 175 BPM. Indicates that natural and artificial atrial pulsing are consistently at the same rate.
Natural ventricular pulsing is set between 30 and 175 BPM and the lower rate limit is set to the same value.	Natural and artificial ventricular pulsing remain at the same phase shift over time.	Pass	Tested at 30, 31, 47, 50, 60, 70, 90, 120, 150, 170, and 175 BPM. Indicates that natural and artificial ventricular pulsing are consistently at the same rate.

All of the tests yield the expected results as anticipated because of the correction factor implemented. Older versions of the Pacemaker Firmware would not have achieved these results which is why the correction factor was implemented. Further details regarding this are provided in 4.2.1.5 Lower Rate Limit. It is important to note that given enough time there will be a change in phase shift as the two rates are not identical, however, over a reasonable amount of time they appear to be the same. Due to the successful testing, the lower rate limit is considered correct.

5.2.3 Off Mode Testing

The following test cases described in Table 5.2.3.a are intended to test the Pacemaker Firmware Off mode and requirement 5. These tests are to be conducted using the HeartView monitoring tool. References to low and high pulse rates refer to the tested pulse rate being lower or higher than the lower rate limit. Interesting test results are further discussed following the table.

Table 5.2.3.a: Off mode test cases verifying requirement 5.

Test	Expected Result	Pass/Fail	Comment
No natural atrial pulsing	No pacing output	Pass	No comment
Low natural atrial pulsing	No pacing output	Pass	No comment
High natural atrial pulsing	No pacing output	Pass	No comment
No natural ventricular pulsing	No pacing output	Pass	No comment
Low natural ventricular pulsing	No pacing output	Pass	No comment
High natural ventricular pulsing	No pacing output	Pass	No comment

As expected Off mode never paces the heart despite any possible heart activity. All tests returned the expected results and Off mode is considered correct.

5.2.4 AOO Mode Testing

The following test cases described in Table 5.2.4.a are intended to test the Pacemaker Firmware AOO mode and requirement 6. These tests are to be conducted using the HeartView monitoring tool. References to low and high pulse rates refer to the tested pulse rate being lower or higher than the lower rate limit. Interesting test results are further discussed following the table.

Table 5.2.4.a: AOO mode test cases verifying requirement 6.

Test	Expected Result	Pass/Fail	Comment
No natural atrial pulsing	Atrial pacing at the lower rate limit	Pass	No comment
Low natural atrial pulsing	Atrial pacing at the lower rate limit	Pass	No comment
High natural atrial pulsing	Atrial pacing at the lower rate limit	Pass	No comment
No natural ventricular pulsing	Atrial pacing at the lower rate limit	Pass	No comment
Low natural	Atrial pacing at the	Pass	No comment

ventricular pulsing	lower rate limit		
High natural ventricular pulsing	Atrial pacing at the lower rate limit	Pass	No comment

As expected AOO mode always paces the atrium at the lower rate limit ignoring heart activity. All tests returned the expected results and AOO mode is considered correct.

5.2.5 VOO Mode Testing

The following test cases described in Table 5.2.5.a are intended to test the Pacemaker Firmware VOO mode and requirement 7. These tests are to be conducted using the HeartView monitoring tool. References to low and high pulse rates refer to the tested pulse rate being lower or higher than the lower rate limit. Interesting test results are further discussed following the table.

Table 5.2.5.a: VOO mode test cases verifying requirement 7.

Test	Expected Result	Pass/Fail	Comment
No natural atrial pulsing	Ventricular pacing at the lower rate limit	Pass	No comment
Low natural atrial pulsing	Ventricular pacing at the lower rate limit	Pass	No comment
High natural atrial pulsing	Ventricular pacing at the lower rate limit	Pass	No comment
No natural ventricular pulsing	Ventricular pacing at the lower rate limit	Pass	No comment
Low natural ventricular pulsing	Ventricular pacing at the lower rate limit	Pass	No comment
High natural ventricular pulsing	Ventricular pacing at the lower rate limit	Pass	No comment

As expected VOO mode always paces the ventricle at the lower rate limit ignoring heart activity. All tests returned the expected results and VOO mode is considered correct.

5.2.6 AAI Mode Testing

The following test cases described in Table 5.2.6.a are intended to test the Pacemaker Firmware AAI mode and requirement 8. These tests are to be conducted using the HeartView monitoring tool. References to low and high pulse rates refer to the tested pulse rate being lower or higher than the lower rate limit. Interesting test results are further discussed following the table.

Table 5.2.6.a: AAI mode test cases verifying requirement 8.

Test	Expected Result	Pass/Fail	Comment
No natural atrial pulsing	Atrial pacing at the lower rate limit	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.
Low natural atrial pulsing	Atrial pacing lower rate limit time after the previous pulse either natural or artificial	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.
Natural atrial pulsing one BPM below the lower rate limit	One artificial pace immediately before each natural pulse	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM. At 120 and 175 BPM the artificial pace comes ~6 and ~10 ms after the natural pulse but still one artificial pace per natural pulse.
Natural atrial pulsing at the lower rate limit	No pacing output	Fail	Tested at a lower rate limit of 31, 60, 120, 175 BPM. Only at 30 BPM was every artificial pace consistently inhibited.
Natural atrial pulsing one BPM above the lower rate limit	No pacing output	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.
High natural atrial pulsing	No pacing output	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.
No natural ventricular pulsing	Atrial pacing at the lower rate limit	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.
Low natural ventricular pulsing	Atrial pacing at the lower rate limit	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.
High natural ventricular pulsing	Atrial pacing at the lower rate limit	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.

For the most part, the AAI mode passes each test at the lower rate limits tested. The significant failure of this set of tests is that the Pacemaker Firmware generates inconsistent pacing when the atrial heart rate in HeartView is set to the lower rate limit. The only tested heart rate that passes this specific test is a lower rate limit of 30 BPM. The project team suspects this is the case because at greater heart rates minor errors become more significant due to the shorter period between pulses. As a result, very minor differences in the generated natural heart rate due to hardware errors associated with the heart microcontroller may cause the Pacemaker Firmware to sometimes pick up the natural pulse and other times to not. More regarding why this design decision was made is spoken about in 4.3.3 AAI Mode. However, as this inconsistent pacing is happening effectively simultaneously with the natural heartbeats, this should not cause an issue and is therefore not considered to be a complete failure of the mode. Additionally, it is interesting that at higher rates the artificial pace comes slightly after the natural pulse when testing at one BPM less than the lower rate limit. This is believed to be the case because of the incredibly short time between the Pacemaker Firmware finishing waiting for a natural pulse to begin generating an artificial pulse and the heart generating a natural pulse. During this small time period after the Pacemaker Firmware has finished waiting for a natural pulse and begun to send the artificial pulse, the natural pulse occurs. As the heart rate increases this time period becomes even shorter and this is suspected to be the reason that 175 BPM has a greater delay than 120 BPM. As the artificial and natural paces effectively happen at the same time this delay is not considered harmful. Increasing the HeartView pulse width parameter during these tests causes all of the minor delays discussed to happen during this longer natural pulse. Finally, as intended the ventricular heart activity does not affect the AAI mode. Considering all that is discussed here the AAI mode is considered to be correct for the purpose of this assignment.

5.2.7 VVI Mode Testing

The following test cases described in Table 5.2.7.a are intended to test the Pacemaker Firmware VVI mode and requirement 9. These tests are to be conducted using the HeartView monitoring tool. References to low and high pulse rates refer to the tested pulse rate being lower or higher than the lower rate limit. Interesting test results are further discussed following the table.

Table 5.2.7.a: VVI mode test cases verifying requirement 9.

Test	Expected Result	Pass/Fail	Comment
No natural ventricular pulsing	Ventricular pacing at the lower rate limit	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.
Low natural ventricular pulsing	Ventricular pacing lower rate limit time after the previous pulse either natural or artificial	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.
Natural ventricular pulsing one BPM below the lower rate limit	One artificial pace immediately before each natural pulse	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM. At 60, 120, and 175 BPM the artificial pace comes ~2, ~16, and ~22 ms after

			the natural pulse but still one artificial pace per natural pulse.
Natural ventricular pulsing at the lower rate limit	No pacing output	Fail	Tested at a lower rate limit of 31, 60, 120, 175 BPM. Only at 30 BPM was every artificial pace consistently inhibited.
Natural ventricular pulsing one BPM above the lower rate limit	No pacing output	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.
High natural ventricular pulsing	No pacing output	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.
No natural atrial pulsing	Ventricular pacing at the lower rate limit	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.
Low natural atrial pulsing	Ventricular pacing at the lower rate limit	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.
High natural atrial pulsing	Ventricular pacing at the lower rate limit	Pass	Tested at a lower rate limit of 31, 60, 120, 175 BPM.

Testing the VVI mode yields nearly identical results to testing the AAI mode, as such the explanation of interesting results can be found in 5.2.6 AAI Mode Testing. The only significant difference between the VVI mode testing is that at one below the lower rate limit of 60 BPM the artificial pace is delayed compared to the natural pulse. However, for the same reasons as the AAI mode, this is not considered to be a test failure. As the AAI mode is considered correct for this assignment, and the VVI mode renders similar testing results then the VVI mode is also considered correct for the purpose of this assignment.