## Part 1 - Simulink

* ~~AOO, VOO, AAI, VVI,~~ DOO, AOOR, VOOR, AAIR, VVIR, DOOR (Jingming Liu)
  + Read details for each mode
  + DCM capability to dynamically change between modes without restarting the device.
  + Add paceLocation = 3 for D modes. ✔
* Rate adaptive modes must track activity using the onboard accelerometer.
  + Research typical BPM for sleeping heart rate/resting heart rate/active heart rate.

|  |  |  |
| --- | --- | --- |
| **Age** | **Target HR Zone 50-85%** | **Average Maximum Heart Rate, 100%** |
| 20 years | 100-170 beats per minute (bpm) | 200 bpm |
| 30 years | 95-162 bpm | 190 bpm |
| 35 years | 93-157 bpm | 185 bpm |
| 40 years | 90-153 bpm | 180 bpm |
| 45 years | 88-149 bpm | 175 bpm |
| 50 years | 85-145 bpm | 170 bpm |
| 55 years | 83-140 bpm | 165 bpm |
| 60 years | 80-136 bpm | 160 bpm |
| 65 years | 78-132 bpm | 155 bpm |
| 70 years | 75-128 bpm | 150 bpm |

* + 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.
  + Response factor (how much to change by)
    - Level 0: LRL, Level 1: LRL +1\*Response, Level: LRL + 2\*Response factor
  + Response time (how long to get to new rate after a change in activity is sensed)
* Add Serial Number (send on Connect)✔
  + + request parameters?

## Part 2 - DCM

* DCM need to include all modes and parameters ✔
* Serial Communication between DCM and Pacemaker ✔
* Add DEPLOY Button to bottom of main page ✔
* Store, set, transmit parameters and verify with Pacemaker (data is specified by srsVVI rev 2 document & complete set in PACEMAKER Table 7) ✔
* DCM read from Pacemaker
  + Pull values from Pacemaker\*\* if we have time
* \*Display egram data (atrium & ventricle) that received by serial link from Pacemaker ✔
* Parameters:
  + Decide data types for variables ✔
  + use typecast() ✔
  + Round to the nearest decimal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parameter** | Programmable Values | Increment | Nominal/Default | Tolerance |
| A or V Pulse Amplitude Regulated | Off, 0.1-5.0V | 0.1V | 5V | 12% |
| A or V Pulse Width | 0.1-30 ms | 0.1 ms | 1 ms | 0.2 ms |
| Sensitivity Adjustment | 0-5V | 0.1 V | - | +/-2% |

* 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 parameter data.
* Notify when a different device is connected. ✔
* Maybe store parameters after the device is disconnected ✔
* Add a reset function? ✔
* Add new parameters: ✔
  + Maximum sensor rate: how high it can go during activity ✔
  + Activity threshold: value from the accelerometer ✔
  + Response factor: how much heartbeat increases ✔
  + Response time: how long to get to a higher upper rate ✔
  + Recovery time: rate of decreasing after activity stopped ✔
  + Sensitivity Adjustment ✔
    - A means shall be provided for the physician to manually adjust the sensing threshold of the device for both the ventricular and atrial sense channels. ✔
  + AV Delay ✔
* Add error handling for pacemaker\_serial not defined. ✔
  + Don’t allow user to deploy or open graph unless pacemaker\_serial is connected
  + Add better errors ✔
* Use new genericError() popup. ✔

## Part 3 - Testing

* Prove correctness in functionality by designing sufficient test-sets
* While documenting provide:
  + Purpose/Test Justification (Brief)
  + System Input
  + Expected Output
  + Actual Output
  + Result (Pass/Fail)

## Bonus

* DDDR mode with the proper AV delay (don't forget to make it rate adaptive).
  + 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.
  + If sense in atrium, pace vent after AV delay
  + AV delay work in rate adaptive mode!
* 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.

Feedback from Assignment 1:

* Max value check is important for DCM Params. ✔
* Better use of Subsystems for pin mapping
* Decomposition details are not very explicit in the document. \*\*?
* Better representation of models in the document will help
* Better representation of states, and state transitions in the document
* Everything/all classes in the same file – in general is not a good idea. Each window/form seems to be a module. Can you find a better way to organize modules \*\*Sorry too late lol
* For testing: screenshots missing. better representation using tables will be good.