DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
THE UNIVERSITY OF TEXAS AT ARLINGTON

ARCHITECTURAL DESIGN SPECIFICATION
CSE 4316: SENIOR DESIGN I
FALL 2023

SIM BREATHE REPEAT
BREATHE EASY SIM

DAVID GOMEZ
ELISABETH HARRIS
KYLE HENRY
ETHAN SPRINKLE
MAICOL ZAYAS

Revision History

Revision	Date	Author(s)	Description
0.1	10.23.2023	DG, EH, KH, ES, MZ	document creation
0.2	11.01.2023	DG, EH, KH, ES, MZ	complete draft
1.0	11.04.2015	DG, EH, KH, ES, MZ	official release
1.1	11.05.2023	KH, ES	added design review requests
1.2	11.06.2023	KH	document submission
1.3	04.15.2024	MZ	peripherals adaptations
1.4	04.28.2024	ES	updated document based on final project turn in

CONTENTS

List of Figures

List of Tables

1 Introduction

The Breathe Easy Sim system is designed to simulate the functionality of the AVEA Ventilator currently used in the University of Texas at Arlington (UTA) SMART Hospital to provide resident students with hands-on experience for treating respiratory failure in a patient. The system is intended to be used alongside a SimMan mannequin, instead of a human patient. SimMan mannequins are advanced patient simulators designed to replicate a human respiration, pulse, reactions, among other things. The Breathe Easy Sim will be calibrated to protect the components of the SimMan mannequins by controlling the amount of airflow into the bladders of the mannequins. Users of the Breathe Easy Sim will be able to touch the screen and turn the knobs to simulate the functionality of the AVEA Ventilator. A fully functioning system consists of three main parts: a Breathe Easy Sim, a SimMan and the user.

The Breathe Easy Sim will have one external input and two external outputs. The one external input to the system is the user. An user is expected to provide all parameters to the system including rate, pressure and flow cycle of the airflow. The first output of the ventilator will be the user itself. The GUI inside the system will display the necessary information for the user to comprehend the status of the procedure and apply its expertise to continue the process. The second external output of the system will be the mannequin patient. Since this product is intended to be used as an educational tool, safety is a top priority. For that reason all input parameters provided by the user will have limited impact on the output to the mannequin, this way we can ensure the protection of all tools.

The main features included in the Breathe Easy Sim system are the Touch Screen, Processing Units, Air Pump System, Oxygenation Tubing, and AC Power Connector. The Touch Screen will be utilized as the main interface for the user. This screen will allow the user to control any parameter with ease through a GUI. The screen will be attached and powered by the main processing unit. The Processing Unit consists of the main processing unit, which is in charge of the Graphical User Interface as well as the Touch Screen. The micro-controller is used to facilitate the handling of user inputs and outputs and control the air pump system. The Oxygenation Tubing will be attached to the patient simulator mannequin. Once successfully attached, this tubing will direct the airflow straight to the mannequin lung bladders and continue the cycle until it is detached or the system is turned off. Similarly, the tubing of the system will mirror that of the AVEA Ventilator. Lastly, the entire system will be powered by the AC Power Connector, which is an alternating current connector straight from a 120V outlet source. This current will be converted into DC using a power converter to then power the control system.

There are a number of key requirements for the Breathe Easy Sim system. First is that the system should closely resemble a real ventilator, with an aesthetic design inspired by the AVEA Ventilator. This visual resemblance is vital to provide students with an immersive and authentic educational experience. Second is that the software inside the Breathe Easy Sim shall be designed to be as similar as possible to the interface and functionality of the AVEA Ventilator. This approach aims to ensure that students can comfortably use the Sim Vent, given its resemblance to the actual ventilator. Another key requirement is that the Breathe Easy Sim shall provide a constant and steady airflow, replicating the behavior of a real ventilator. This constant airflow is essential for creating an accurate and effective simulation for educational purposes. Also, the Breathe Easy Sim shall be capable of simulating live readings on a mannequin. The readings should be generated by the ventilator, replicating the data and measurements that would be observed in a real clinical setting. The final key requirement for the Breathe Easy Sim system is that it shall be designed to operate when plugged into a power source. The ability to use the Breathe Easy Sim while connected to a power supply is a key customer requirement to ensure continuous operation.

2 System Overview

The Breathe Easy Sim system will consist of 4 layers: the Processing Layer, User Software Layer, Peripherals Layer, and the Air Pump Layer. The Processing Layer is the middle-man layer that connects the other layers together. The Air Pump Layer directly interacts with the Processing Layer, but not the Peripherals Layer or the User Interface Layer. The Peripherals Layer also only directly interacts with the Processing Layer, but not the Air Pump Layer or the User Interface Layer. The User Interface layer does not directly interact with the Peripherals Layer or the Air Pump Layer, rather it only directly interacts with the Processing Layer. Thus, the Air Pump, Peripherals, and User Interface Layers all rely on the Processing Layer to send signals and information between each them. Below is a diagram and detailed description of the layers and how they interact with each other.

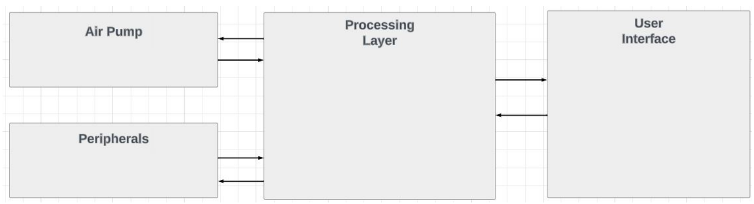

Figure 1: A simple architectural layer diagram

2.1 Processing Layer

The Processing Layer is the main layer that all systems interact with. It does three main jobs. First, the processing layer receives signals from the peripheral layer, air pump layer, and user interface layer. These signals are then processed and/or translated depending on the content of the signals. Then the signals are sent to their destination layers. Air pump layer and peripheral layer signals are sent to the user interface layer and vice versa.

2.2 User Software

The User Software Layer is the interface through which users interact with the system. It provides a user-friendly environment for controlling and monitoring the simulation, offering access to settings, system parameters, patient simulations, and event detection. This layer ensures that users can easily configure the system, observe simulations, and respond to critical events, enhancing the educational experience in respiratory care and ventilation.

2.3 Peripherals

The Peripheral layer represents the physical inputs and outputs that form the system. The inputs will be coming from Push-buttons, Encoders, and a Touch-Screen displaying the User Interface. There are two main outputs, not counting the screen display, LED's and a Speaker.

2.4 Air Pump

The Air Pump system is responsible for providing air to the simulated mannequin through the use of a motor controlled pump and bellows system. The Processing Layer will interface with the motor and limit switches to regulate air flow.

3 Subsystem Definitions & Data Flow

There are a total of 12 components that comprise the Breathe Easy Sim system. These components are referred to as subsystems, which are part of the four main layers discusses in the previous section.

The Air Pump Layer contains two components: the Airflow Controller and the System Feedback. The Airflow Controller receives input from the Data Processing component in the Processing Layer on air flow rate and motor position to control the pump and bellows system. A limit switch communicates with the Data Processing layer to indicate position and calibrate the air pump.

The Peripherals Layer contains three components. The Physical Input component takes input from the user in the form of push buttons, knobs, and the touch screen. For all types of input, the input information will be sent to the Data Processing component in the Processing Layer. The Alarm component and the LED component both receive input from the Signal Handler component in the Processing Layer. The Alarm component will deliver the information received through audio, using a speaker. The LED component will deliver the information received through LED lights on the body of the ventilator.

The Processing Layer contains three components. The Data Processing Component is the heart of the layer. It serves as the hub where all the data and information is sent to be processed and then sent out to the correct component. It takes input from the Physical Input component and the System Feedback component, and it outputs information to the Settings Screen component and the Biometrics Simulation component. The Biometrics Simulation component is a part of the Processing Layer, as it sends information to the Patient Simulation Screens component in the User Interface Layer. The Signal Handler component in the Processing Layer is not connected to the Data Processing Layer, rather it takes input from the Event Detection component in the User Interface Layer regarding a triggered event from the Settings Screen. Then the Signal Handler send that information to either the Alarm or LED component in the Peripheral Layer.

The User Interface Layer contains four components. The Settings Screen receives input from the Data Processing Component about information regarding non-integral patient behaviors. That information is sent to the System Parameters component, where the system information is stored. The Settings Screen also receives input from the System Parameters component when information about the system has been changes. When an event that requires physical output is triggered in the Settings Screen, it outputs that to the Event Detection component, where those triggers are sent to the Signal Handler component. The Patient Simulation Screens send changes to the System Parameters component, and when that happens the System Parameters component and it receives changes from the System Parameters component. However, the main source of input for the Patient Simulation Screens is the Biometrics Simulation component in the Processing Layer. Like mentioned earlier, the System Parameters component sends and receives information from the Settings Screen and the Patient Simulation Screens. The Event Detection component receives input from the Settings Screen and the Patient Simulation Screens about triggered events, then sends that information to the Signal Handler to process that information.

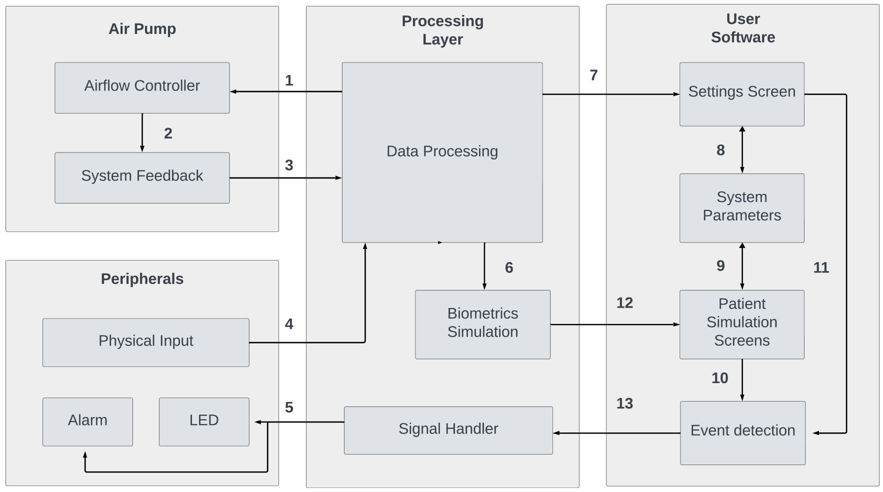

Figure 2: A simple data flow diagram

4 Processing Layer Subsystem

The Processing Layer consists of 3 subsystems: Data Processing, Biometrics Simulation, and Signal Handler. The Data Processing subsytem translates and interprets information from subsystems in the Air Pump and Peripheral Layers and pass information into the User Interface and Air Pump subsystems. The Biometrics Simulation subsystem receives information from the Data Processing subsystem and updates the Patient Simulation Screens accordingly. The Signal Handler receives signals to display through the attached peripherals (LED's, alarms, etc) from the User Interface subsystem.

4.1 Data Processing

The Data Processing subsystems' purpose is to take in the information given by the System Feedback and Physical Input subsystems and interpret/send the data to the subsystem it is needing to go to. The three subsystems that this subsystem sends information to is Airflow Controller, Settings Screen, and Biometrics Simulation.

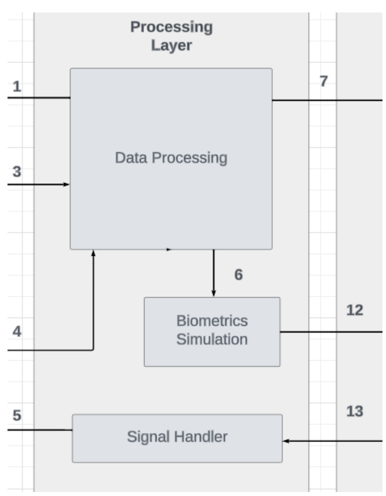

Figure 3: Processing Layer subsystem description diagram

4.1.1 Assumptions

There are 2 main assumptions made for this subsystem. The first assumption is that all of the hardware-based layers this system relies on are functional. This means that the air pump, LED's, alarms, knobs, touch screen, and buttons are all functioning and communicating values properly. The second assumption is that the Biometrics Simulation and Setting Screen subsystems are online to receive and handle the data processed by data processing.

4.1.2 Responsibilities

This layers responsibilities are as follows. It is is supposed to update the Air Pump Layer with the airflow instructions to be carried out. It is also supposed to send physical input signals to the User Interface to update the screens that the users are seeing to the expected user-desired output.

4.1.3 Subsystem Interfaces

Table 2: Data Processing Subsystem interfaces

ID	Description	Inputs	Outputs
#1	Airflow instructions are sent to the Air Pump to control airflow rates.	Airflow Controller	Data Proccessing
#3	The Air Pump location state is fed into data processing via a limit switch.	Data Proccessing	System Feedback
#4	A button, knob, or touch screen signal is sent to data processing to be translated to an instruction for the user interface and, potentially, the air pump to carry out.	Data Proccessing	Physical Input
#6	Biometrics simulations are generated and simulated using data in the Processing Layer	Biometrics Simulation	Data Proccessing
#7	Physical inputs are sent to the settings screen to update the specific values on the user interface based off user input.	Settings Screen	Data Proccessing

4.2 Biometrics Simulation

The Biometrics Simulation subsystem controls the the simulated patient biometrics going into the user interface. The output is dependent upon the processed input from the System Feedback subsystem..

4.2.1 Assumptions

There are 4 assumptions made for this subsystem. The first is that the Air Pump Layer is online and functioning properly. The second assumption is that the Data Processing Layer is online. The next assumption assumes that the Data Processing subsystem communicates over the state of the air pump in a small window of time (less than a second). The last assumption made is that the Biometrics Simulation subsystem can put together graph coordinates of various breathing information based on simulated data.

4.2.2 Responsibilities

This subsystem has 2 responsibilities. The first responsibility is to generate coordinates for a time graph that correspond to how the air pump is "inhaling" and "exhaling" so to speak. The second responsibility is to send that information to the Patient Simulation Screens.

4.2.3 Subsystem Interfaces

Table 3: Biometrics Simulation Subsystem interfaces

ID	Description	Inputs	Outputs
#6	Biometrics simulations are generated and simulated using data in the Processing Layer	Biometrics Simulation	Data Processing
#12	The calculated coordinates and information are sent to be displayed in the GUI of the Patient Screens	Patient Simulation Screens	Biometrics Simulation

4.3 Signal Handler

The Signal Handler works as an intermediary between the hardware outputs components and the User Interface. This module receives its input from the Event Detection subsystem and sends output signals to both Alarm and LED peripherals. In case of the LED outputs, the Signal Handler simply uses GPIO pins to send digital signals that along with the circuit, manipulates each LED. The case of the Alarm is more elaborate.

4.3.1 Assumptions

The Signal Handler connects directly with the Alarm subsystem.

The Signal Handler will use interrupt functions to prioritize output peripherals.

The Signal Handler works as a bus between the LED's and User Interface, however the alarm inputs require extra processing.

4.3.2 Responsibilities

This sublayer is in charge of sending the user the expected output through the use of peripherals. It connects the Event Detector sublayer to the output hardware, being an intermediary and data processing module.

4.3.3 Subsystem Interfaces

The input to this system comes from the Event Detector sublayer in User Interface layer. The output of the system goes directly to the Alarm and LED's sublayers inside the Peripherals layer.

Table 4: Signal Handler Subsystem interfaces

ID	Description	Inputs	Outputs
#5	Signal Handler sends output signals to Alarm and LED layers to notify user.	LED/Alarm	Signal Handler
#13	Event Detection triggers a response that notifies the Signal Handler	Signal Handler	Event Detection

5 User Software Subsystem

The User Software Layer consists of 4 subsystems: Settings Screen, System Parameters, Patient Simulation Screens, and Event Detection.

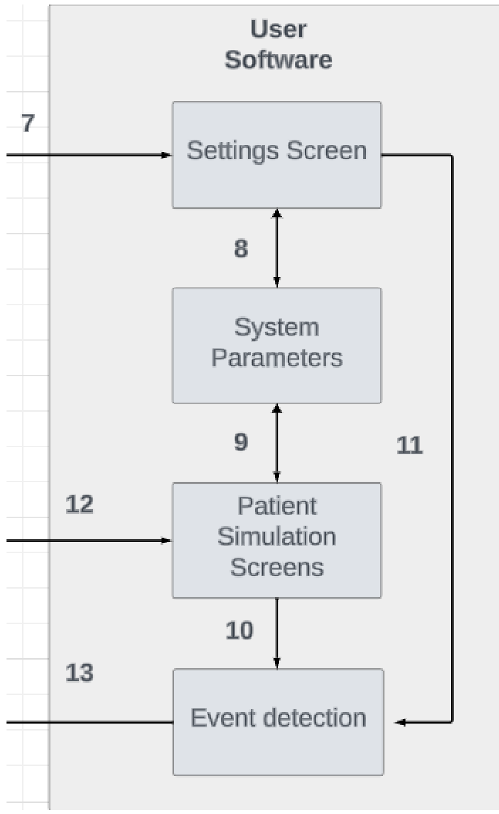

Figure 4: User Software Subsystem

5.1 Settings Screen

The Settings Screen subsystem is in charge of showing ventilator behavior that are non integral to the patient. This subsystem receives input from the Data Processing subsystem under the Processing Unit layer, which informs the Settings Screen what to display on the screen. Some of the behavior changes include changes that require a physical output on the ventilator, thus in that instance the Settings Screen will output to the Event Detection subsystem to handle that event. The Settings Screen also receives input and sends output to the Systems Parameters subsystem.

5.1.1 Assumptions

The Settings Screen assumes that the Data Processing subsystem receives input from the touch screen. The Settings Screen does not directly receive input from the touch screen, rather the Data Processing subsystem is the middle-man between the two. The Settings Screen also assumes that the Event Detection subsystem will send the correct information to the correct subsystem in a timely manner such that the physical output of alarms or LEDs is not interrupted.

5.1.2 Responsibilities

The Settings Screen subsystem is responsible for reflecting the changes in the ventilator behavior based on inputs from the user and the patient, which are received from the Data Processing subsystem and the System Parameters subsystem respectively. It is also responsible for signaling to the Event Detection subsystem when the ventilator needs to emit some sort of physical output, such as alarms or LEDs.

5.1.3 Subsystem Interfaces

There are two inputs to the Settings Screen subsystem: the Data Processing subsystem and the System Parameters subsystem. The input received from the Data Processing subsystem informs the Settings Screen what behaviors of the ventilator were changed so it can accurately reflect those changes. The Settings Screen then takes those changes and outputs them to the System Parameters subsystem. Also, when certain behaviors are changed from the input received from the Data Processing subsystem, the Settings Screen subsystem then outputs information regarding physical outputs of the ventilator, like LEDs and alarms, to the Event Detection subsystem. Also, input for the Settings Screen comes from the System Parameters subsystem when there is information that is changed from the Patient Simulation Screens that needs to be reflected on the Settings Screen.

Table 5: Settings Screen Subsystem interfaces

ID	Description	Inputs	Outputs
#7	Settings Screen receives data from the Data Processing Layer for display	Settings Screen	Data Processing
#8	Settings Screen continuously interfaces and provides feedback with System Parameters	Settings Screen	System Parameters
#11	Event detection is handled and acknowledged on the Settings Screen	Event Detection	Settings Screen

5.2 System Parameters

The System Parameters subsystem is a vital component within the Simulated Ventilator architecture, residing in the Processing Layer. This subsystem is responsible for managing and maintaining critical system settings and parameters that dictate the behavior of the simulation. It serves as the central hub for configuring ventilation settings and patient simulation scenarios.

5.2.1 Assumptions

The System Parameters subsystem assumes that it will receive accurate and real-time data from the Data Processing subsystem to base its calculations and system parameters on. It assumes that the User Software Layer will correctly implement user inputs for configuring ventilation and patient scenarios. It expects the Event Detection subsystem to use the system parameters to evaluate alarm conditions and trigger appropriate responses.

5.2.2 Responsibilities

Ventilation Settings Configuration: The System Parameters subsystem allows users to configure ventilation settings.
Real-Time Parameter Update: The subsystem is responsible for real-time updates of system parameters, ensuring that changes in ventilation settings or patient scenarios are immediately reflected in the simulation.
Alarm Threshold Definition: It sets and maintains alarm thresholds based on user-defined parameters

and ventilation settings. This includes defining limits for physiological parameters such as heart rate, oxygen saturation, and more.

5.2.3 Subsystem Interfaces

There are two inputs to the System Parameters subsystem: the Patient Simulation Screens and the Settings Screen. The input received from the Patient Simulation Screens informs the System Parameters.

Table 6: System Parameters Subsystem interfaces

ID	Description	Inputs	Outputs
#8	Settings Screen continuously interfaces and provides feedback with System Parameters	Settings Screen	System Parameters
#9	Simulation Screens display information based on System Parameters.	System Parameters	Patient Simulation Screens

5.3 Patient Simulation Screens

The Patient Simulation Screens subsystem is in charge of showing and changing ventilator behavior that are integral to the patient. It receives input from both the Biometrics Simulation subsystem found in the Processing Layer as well as the System Parameters subsystem. It outputs to both the System Parameters subsystem and the Event Detection subsystem.

5.3.1 Assumptions

The Patient Simulation Screens assumes that it is receiving real-time and accurate data from the Biometrics Simulation subsystem so that it can accurately represent that on the appropriate screens and output the appropriate information to the appropriate subsystem. The Patient Simulation Screens also assumes that the Event Detection subsystem will send the correct information to the correct subsystem in a timely manner such that the physical output of alarms or LEDs is not interrupted.

5.3.2 Responsibilities

The responsibility of the Patient Simulation Screens is first to boot up when the ventilator is first turned on. Then it will continually show simulated diagnostics for the screen on the ventilator to show. It will also show changes made from the physical inputs of the touch screen and knobs through receiving input from the Biometrics Simulation subsystem. When certain physical inputs are changed or when certain behaviors are recorded from the SimMan mannequin, the Patient Simulation Screens will then send output to the Event Detection subsystem to signal that there is physical output that the ventilator needs to emit.

5.3.3 Subsystem Interfaces

There are two inputs to the Patient Simulation Screens subsystem: the Biometrics Simulation subsystem and the System Parameters subsystem. The input received from the Biometrics Simulation subsystem informs the Patient Simulation Screens what input is being received from the SimMan mannequin behaviors so it can accurately reflect that information on the screens. The Patient Simulation Screens then takes those changes and outputs them to the System Parameters subsystem. Also, when certain behaviors are received from the input received from the Biometrics Simulation subsystem, the Patient Simulation Screens subsystem then outputs information regarding physical outputs of the ventilator, like LEDs and alarms, to the Event Detection subsystem. Also, input for the Patient Simulation Screens comes from the System Parameters subsystem when there is information that is changed from the Settings Screen that needs to be reflected in the Patient Simulation Screens.

Table 7: Patient Simulation Screens Subsystem interfaces

ID	Description	Inputs	Outputs
#9	Simulation Screens display information based on System Parameters.	System Parameters	Patient Simulation Screens
#12	The Biometrics simulation data is displayed on the Simulation Screen	Patient Simulation Screens	Biometrics Simulation
#10	Events from the Simulation screen are sent to Event detection subsystem for handling	Event Detection	Patient Simulation Screens

5.4 Event Detection

The Event Detection Subsystem is a pivotal component within the Simulated Ventilator (Sim Vent) architecture, housed in the User Software Layer. This subsystem is responsible for monitoring the simulation in real-time, detecting critical events and alarm conditions, and initiating appropriate responses. It ensures the safety and educational effectiveness of the Sim Vent system.

5.4.1 Assumptions

The Event Detection Subsystem assumes that the User Software Layer will provide accurate and appropriate inputs from system parameters and alarm thresholds. It relies on the System Parameters subsystem to maintain and update alarm threshold settings based on user-defined configurations.

5.4.2 Responsibilities

Real-Time Event Monitoring: The Event Detection Subsystem continuously monitors the simulation, analyzing data and parameters to detect critical events in real-time.

Alarm Generation: It generates alarms when specific alarm conditions are met, such as deviations from normal ranges, malfunctions, or predefined clinical events.

Alarm Notifications: The subsystem notifies users and relevant parties of alarms through visual, auditory, or textual notifications, ensuring prompt and appropriate responses.

Response Triggers: It triggers responses to alarms, such as adjustments to ventilation settings, and initiation of predefined interventions.

5.4.3 Subsystem Interfaces

The Event Detection subsystem takes input from both the Patient Simulation Screens and the Settings Screen. It outputs the information to the Signal Handler to then send to the needed places.

Table 8: Event Detection Subsystem interfaces

ID	Description	Inputs	Outputs
#10	Events from the Simulation screen are sent to Event detection subsystem for handling	Event Detection	Patient Simulation Screens
#11	Event detection is handled and acknowledged on the Settings Screen	Event Detection	Settings Screen
#13	Event detection is sent to Signal Handler	Signal Handler	Event Detection

6 Peripherals Subsystem

The Peripheral Layer consists of hardware devices that function as the main input and output of the whole system. These devices include Push buttons, Touchscreen, Speakers, LED, etc. The user will control the system from the Peripheral layer. This layer includes only inputs from the user. The purpose of this layer is to separate the hardware components of the system from the software to facilitate its analysis.

The three subsystems included in this layer are Physical Inputs, Alarms, and LED.

6.1 Physical Input

The Physical Input sublayer consists of all three external inputs of the system. These three inputs are Push-Buttons, a Rotary Encoder (Knobs), and the Touch Screen. All three inputs will have similar data flow, sending their signals directly to the Data Processing sublayer under the Processing Unit layer. These signals will be translated and sent to the corresponding programs under the User Interface.

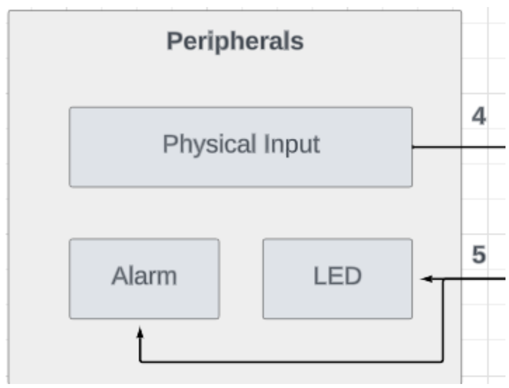

Figure 5: Peripherals subsystem description diagram

6.1.1 Assumptions

This sublayer only consists of the hardware and circuits that surround the physical inputs. All code utilized to read and process the data is found on the Processing Unit layer under Data Processing. It is assumed that the code will utilize interrupts to properly prioritize the generated signals.

Although this sublayer considers the Touch Screen as part of the physical input, it is only considered so because of the touch capabilities. The actual display of the Screen is not considered a physical input or output, it is instead assumed to be part of the User Interface layer under the multiple screens.

6.1.2 Responsibilities

The input peripherals provide the user the ability to interact with the system software. For the most part, push buttons will be utilized to interface the multiple screens that the GUI provides, while the rotary encoder modifies the parameters of the system, such as air pressure. The touchscreen input will be useful for navigating the GUI with ease.

6.1.3 Subsystem Interfaces

This sublayer only has one output destination which is the Data Processing. No other inputs are connected to this sublayer. The only input to the layer comes from the user itself.

Table 9: Physical Input Subsystem interfaces

ID	Description	Inputs	Outputs
#4	Physical Input signals from the user are sent to the Data Processing Layer	Data Processing	Physical Inputs

6.2 Alarm

The Alarm sublayer will inform the user if the system is properly working, alerting them of any communication or booting errors. This information is being delivered as audio, utilizing a speaker that can be toggled on or off at any moment. The input to the speaker comes from the Signal Handler sublayer under the Processing Unit layer.

6.2.1 Assumptions

Any assumptions made in the definition of the subsystem should be listed and described. Pay particular attention to assumptions concerning interfaces and interactions with other layers.

The Alarm sublayer does not include any data processing of the output, it is assumed that the Signal Handler will receive the interrupt from the User Interface program and send the analog signal needed.

The sound level of the speaker will be adjusted inside the Settings Screen on the User Interface. This value will be stored at the System Parameters and whenever an alarm interrupt occurs inside the Patient Simulation Screen, it will be sent to Event Detection and then to Signal Handler. At this point, the signal handler will receive the alert along with the requested sound level.

The User can toggle on or off the use of the speaker inside the Settings Screen. The speaker won't have an in-circuit switch, instead, it will be entirely controlled by software. Once turned on, it will start receiving input once again.

6.2.2 Responsibilities

The Alarm is responsible for informing the user of the state of the system under specific circumstances. The most common use for the alarm is to inform the User of the state of operation of the system.

6.2.3 Subsystem Interfaces

Any input to this system will come from the Signal Handler sublayer. The Signal Handler sublayer will acquire its input from the Event Detection, inside the User Interface, and send the necessary signals to the speaker.

Table 10: Alarm Subsystem interfaces

ID	Description	Inputs	Outputs
#5	Alarm notification triggered base on Signal Handler interrupts	Alarm	Signal Handler

6.3 LED

The LED sublayer is very intuitive, it represents a single LED lights placed in the body of the Ventilator. The purpose of the LED is to notify the user when the ventilator is on and can be used as part of an alarm.

6.3.1 Assumptions

The LED will be controlled by the User Interface layer.

6.3.2 Responsibilities

The LED has the sole responsibility of providing the user additional ventilator information.

6.3.3 Subsystem Interfaces

The only input to this system will be the Signal Handler, which sends digital signals directly to the circuits through the GPI controller pin.

Table 11: LED Subsystem interfaces

ID	Description	Inputs	Outputs
#5	LED	LED	Signal Handler

7 Air Pump Subsystem

The Air Pump subsystem is responsible for providing air flow to the simulated mannequin through the use of a motor controlled pump and bellows system. Positive and negative air pressure will be controlled through the use of a stepper motor in conjunction with a position sensor to ensure air flow is delivered at the appropriate rate and speed. The Air Pump system will also provide System Feedback to the Processing Layer in order to properly calculate and display air flow rate and alert the user to any errors with the Air Pump system.

7.1 Airflow Controller

The Airflow Controller is comprised of a stepper motor controlled bellows system to provide positive and negative air pressure to the simulated mannequin. Compressing the bellows results in positive air pressure (breathing in) while decompressing the bellows results in negative air pressure (breathing out). The presence of additional valves will help ensure that the positive and negative air pressure amounts do not exceed the limit of the mannequin's air bladders.

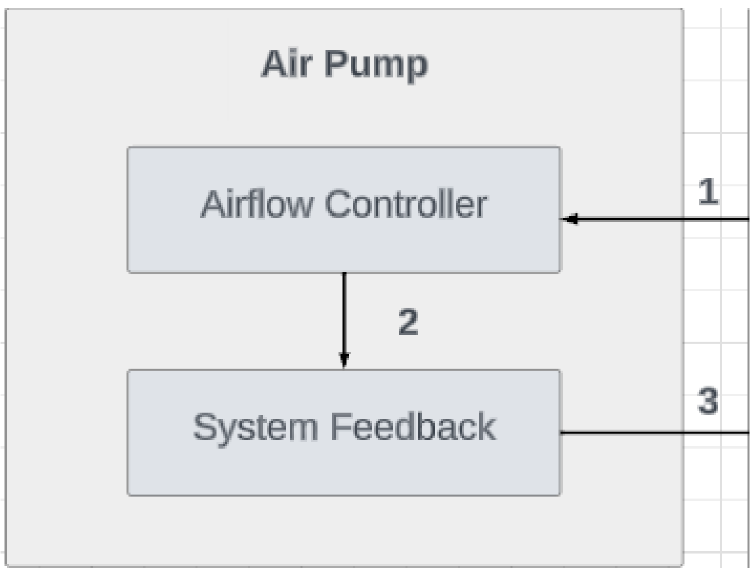

Figure 6: Subsystem description diagram of the Air Pump System

7.1.1 Assumptions

The Airflow Controller will be calibrated to the bladders provided by the customer to ensure that the air flow amounts do not exceed or damage the integrity of the bladder. Additional maintenance and calibration will be required in the event the bladder design is changed.

7.1.2 Responsibilities

The responsibility of the Airflow Controller is to provide air flow to the simulated mannequin and to provide diagnostics to the feedback system.

7.1.3 Subsystem Interfaces

The only input to the Airflow Controller comes from the Data Processing subsystem, where motor position and air flow rates are controlled. The Airflow Controller outputs information that is then gathered by the System Feedback subsystem.

Table 12: AirFlow Controller Subsystem interfaces

ID	Description	Inputs	Outputs
#1	Air Flow and Rate are controlled from the Data Processing Layer	Airflow Controller	Data Processing
#2	Air Pump Position is sent to System Feedback	System Feedback	Airflow Controller

7.2 System Feedback

The System Feedback subsystem gathers information from the Airflow Controller and reports it to the Data Processing subsystem through the use of a position sensor.

7.2.1 Assumptions

The System Feedback subsystem assumes that all sensors used to provide information will be operating properly and are properly calibrated. Low level limits will be established to help diagnose potential problems in the Air Pump system, but not every contingency can be accounted for.

7.2.2 Responsibilities

The responsibility of the System Feedback subsystem is to gather information from the Airflow Controller and report it to the Data Processing subsystem. The System Feedback subsystem does not provide feedback to the Airflow Controller directly, but instead passes the information to the Data Processing subsystem for action.

7.2.3 Subsystem Interfaces

The only input to the System Feedback subsystem comes from the Airflow Controller, where information regarding the controller is gathered. The System Feedback subsystem then outputs said information to the Data Processing subsystem for analysis and further action.

Table 13: System Feedback Subsystem interfaces

ID	Description	Inputs	Outputs
#2	Air Pump Position is sent to System Feedback	System Feedback	Airflow Controller
#3	Air Pump Position is sent to Data Controller	Data Processing	System Feedback

REFERENCES