Alberta E-Vent Control Software

Initial Release and Development Overview

April 24, 2020

Document History

Date	Document Status	Originator	Approver
April 24, 2020	Draft	C. Hill & D. Quinn	

Approval

Date	Name	Role	Signature

Contents

1 Summary

TODO:

2 Acronyms

AC Assist Control

BPM Breaths per Minute

IT Inspiratory Time

PEEP Positve End-Expiratory Pressure

PIP Peak Inspiratory Pressure

RR Respiratory Rate

TP Trigger Pressure

TV Tidal Volume

VC Volume Control

3 Software Classification

The Alberta E-Vent has been designed as a emergency ventilation system which requires human supervision. The intent of the system is to offload the physical burden of manually running a bag-valve-mask setup while still utilizing the trained supervision of medical professional. Further, it is expected that the device will be used on patients capable of spontaneous breathing which requires the Assist Control Mode of ventilation.

The Alberta E-Vent utilizes an external mechanical relief valve to limit the maximum operating pressure of the ventilator. This valve prevents the ventilator from generating pressures that would cause severe harm to the patient. Further, the ventilator acts as an aid to patient's breathing but does not prevent them from spontaneous inhalation or exhalation during operation. The anticipated operating conditions of the ventilator, as detailed in the preceding paragraph, would mean that a software malfunction will not cause suffocation. The primary risks associated with a software failure relate to asynchronous breathing and the associated lung injury associated with this operating condition. This rationalization provides the justification for treating the software as Class B in accordance with IEC 62304.2006.

4 IEC 62304.2006 Overview

The Alberta E-Vent Software has designated Class B. See Section 3.

Ref.	Software Lifecycle Process	Class			Relevant Section
		A	B	C	
4.1.0	Quality Management Systems	X	X	X	See Section 5
4.2.0	Risk Management	X	X	X	See Section 9
4.3.0	Software safety classification	X	X	X	See Section 3
5.1	**Software development planning**				
5.1.1	Software Development Plan	X	X	X	N/A; See footnote 1
5.1.2	Software Development Plan maintenance	X	X	X	N/A; See footnote 1
5.1.3	Software development plan reference to system design and development	X	X	X	N/A; See footnote 1
5.1.4	Software development standard, methods and tools planning			X	N/A; Class B software
5.1.5	Software integration and integration testing planning		X	X	N/A; See footnote 3
5.1.6	Software verification planning	X	X	X	See Section 10
5.1.7	Software risk management planning	X	X	X	See Section 9
5.1.8	Documentation planning	X	X	X	N/A; See footnote 1
5.1.9	Software configuration management planning	X	X	X	N/A; See footnote 1
5.1.10	Supporting Items to be controlled		X	X	See Section 5
5.1.11	Software configuration item controlled before verification		X	X	See Section 5
5.2	**Software requirements analysis**				
5.2.1	Define and document software requirements from system requirements	X	X	X	See Section 7
5.2.2	Software requirements content	X	X	X	See Section 7
5.2.3	Include risk control measures in software requirements		X	X	See Section 7
5.2.4	Re-evaluate medical device risk analysis	X	X	X	See Section 9
5.2.5	Update requirements	X	X	X	See Section 7
5.2.6	Verify software requirements task	X	X	X	See Section 10
5.3	**Software architectural design**				
5.3.1	Transform software requirements into an architecture		X	X	See Section 8
5.3.2	Develop an architecture for the interfaces of software items		X	X	See Section 8
5.3.3	Specify functional and performance requirements of SOUP item		X	X	N/A; See footnote 2
5.3.4	Specify system hardware and software required by SOUP item		X	X	N/A; See footnote 2
5.3.5	Identify segregation necessary for risk control			X	N/A; Class B software

5.3.6	Verify software architecture		X	X	N/A; See footnote 1
5.4	**Software detailed design**				
5.4.1	Subdivide software into software units		X	X	N/A; See footnote 3
5.4.2	Develop detailed design for each software unit			X	N/A; Class B software
5.4.3	Develop detailed design for interfaces			X	N/A; Class B software
5.4.4	Verify detailed design task			X	N/A; Class B software
5.5	**Software unit implementation**				
5.5.1	Implement each software unit	X	X	X	N/A; See footnote 3
5.5.2	Establish software unit verification process		X	X	N/A; See footnote 3
5.5.3	Software unit acceptance criteria		X	X	N/A; See footnote 3
5.5.4	Additional software unit acceptance criteria			X	N/A; Class B software
5.5.5	Software unit verification		X	X	N/A; See footnote 3
5.6	**Software Integration Testing**				
5.6.1	Integrate software units		X	X	N/A; See footnote 3
5.6.2	Verify software integration		X	X	N/A; See footnote 3
5.6.3	Software integration testing		X	X	N/A; See footnote 3
5.6.4	Software integration testing content		X	X	N/A; See footnote 3
5.6.5	Evaluate software integration test procedures		X	X	N/A; See footnote 3
5.6.6	Conduct regression tests		X	X	N/A; See footnote 3
5.6.7	Integration test record contents		X	X	N/A; See footnote 3
5.6.8	Use software problem resolution process		X	X	N/A; See footnote 3
5.7	**Software system testing**				
5.7.1	Establish tests for software requirements	X	X	X	See Section 10
5.7.2	Use software problem resolution process	X	X	X	See Section 10
5.7.3	Retest after changes	X	X	X	See Section 10
5.7.4	Evaluate software system testing	X	X	X	See Section 10
5.7.5	Software system test record contents	X	X	X	See Section 10
5.8	**Software release**				
5.8.1	Ensure software verification is complete	X	X	X	See Section 10
5.8.2	Document known residual anomalies	X	X	X	See Section 10.4
5.8.3	Evaluate known residual anomalies		X	X	See Section 10.4
5.8.4	Document released versions	X	X	X	Addressed by this document in its entirety
5.8.5	Document how released software was created		X	X	Addressed by this document in its entirety
5.8.6	Ensure activities and tasks are complete		X	X	Addressed by this document in its entirety
5.8.7	Archive software	X	X	X	See Section 5

5.8.8	Assure reliable delivery of released software	X	X	X	See Section 5
6	**Software maintenance process**				
6.1	Establish software maintenance plan	X	X	X	See Section 11; See footnote 4
6.2	**Problem and Modification**				
6.2.1	Document and evaluate feedback	X	X	X	See Section 11; See footnote 4
6.2.1.1	Monitor feedback	X	X	X	See Section 11; See footnote 4
6.2.1.2	Document and evaluate feedback	X	X	X	See Section 11; See footnote 4
6.2.1.3	Evaluate problem report's affects on safety	X	X	X	See Section 11; See footnote 4
6.2.2	Use software problem resolution process	X	X	X	See Section 11; See footnote 4
6.2.3	Analyze change requests	X	X	X	See Section 11; See footnote 4
6.2.4	Change request approval	X	X	X	See Section 11; See footnote 4
6.2.5	Communicate to users and regulators	X	X	X	See Section 11; See footnote 4
6.3	**Modification Implementation**				
6.3.1	Use established process to implement modification	X	X	X	See Section 11; See footnote 4
6.3.2	Re-release modified software system	X	X	X	See Section 11; See footnote 4
7	**Software risk management process**				
7.1.1	Identify software items that could contribute to a hazardous situation		X	X	See Section 9
7.1.2	Identify potential causes of contribution to a hazardous situation		X	X	See Section 9
7.1.3	Review published SOUP anomaly lists		X	X	See Section 7.1
7.1.4	Document potential causes		X	X	See Section 9
7.2	**Risk control measures**				
7.2.1	Define Risk control measures		X	X	See Section 9
7.2.2	Risk control measures implemented in software		X	X	See Section 9
7.3	**Verification of Risk control measures**				
7.3.1	Verify Risk control measures		X	X	See Section 10.3
7.3.3	Document traceability		X	X	Addressed by this document in its entirety
7.4	**Risk management of software changes**				
7.4.1	Analyze changes to medical device software with respect to safety	X	X	X	See Section 11; See footnote 4

7.4.2	Analyze impact software changes on existing risk control measures		X	X	See Section 11; See footnote 4
7.4.3	Perform risk management activities based on analyses		X	X	See Section 11; See footnote 4
8	**Software configuration management Process**				
8.1.1	Establish means to identify configuration items	X	X	X	See Section 7.1
8.1.2	Identify SOUP	X	X	X	See Section 7.1
8.1.3	Identify system configuration documentation	X	X	X	See Section 7
8.2	**Change control**				
8.2.1	Approve Change requests	X	X	X	See Section 11; See footnote 4
8.2.2	Implement changes	X	X	X	See Section 11; See footnote 4
8.2.3	Verify changes	X	X	X	See Section 11; See footnote 4
8.2.4	Provide means for traceability of change	X	X	X	See Section 11; See footnote 4
8.3	**Configuration status accounting**				
8.3.1	Configuration status accounting task	X	X	X	See Section 11; See footnote 4
9	**Software problem resolution Process**				
9.1	Prepare problem reports	X	X	X	See Section 11; See footnote 4
9.2	Investigate the problem	X	X	X	See Section 11; See footnote 4
9.3	Advise relevant parties	X	X	X	See Section 11; See footnote 4
9.4	Use change control process	X	X	X	See Section 11; See footnote 4
9.5	Maintain records	X	X	X	See Section 11; See footnote 4
9.6	Analyze problems for trends	X	X	X	See Section 11; See footnote 4
9.7	Verify software problem resolution	X	X	X	See Section 11; See footnote 4
9.8	Test documentation contents	X	X	X	See Section 11; See footnote 4

Footnotes:

1. The development of the Alberta E-Vent occurred over 1-month period to respond to the immediate need for emergency ventilation systems for use in the COVID-19 pandemic. The rapid development schedule meant that not all aspects of the IEC 62304 software development criteria could be followed. The Alberta E-Vent hardware and software was developed using an

agile development procedure with a small team. The focus of the development procedure has been to iterate quickly, solicit clinician feedback, and implement requested changes.

2. Software of Unknown Pedigree (SOUP) has been avoided where possible, however, the software developed utilizes the Arduino suite of IDE, compiler, and libraries. The use of an a widely available platform intended for rapid development, like Arduino, was necessitated by the short development timelines associated with this project. A selection of additional open-source libraries not associated with the Arduino suite have been used when necessary. Full details of the the SOUP utilized in the Alberta E-Vent is given in Section 7.1 of this document. Risks associated with using SOUP has been reduced by extensively testing the full ventilator to ensure it functions as expected in reliable manner.

3. The software developed for the Alberta E-Vent is relatively compact (less than 10k lines). As such, the entire software suite required to operate the ventilator has been treated as a single unit. The software has been developed and tested as a single unit. The small size of the code base enabled this type of testing to be effective for debugging and verification testing.

4. Items related to ongoing maintenance of the software are not addressed in detail by the current set of documents. A brief overview of the maintenance polices to be employed is given in Section 11. The intent of the current set of documents is to outline the software development procedure.

5 Quality Management

Quality management of Alberta E-Vent software is paramount to ensuring that all software delivered with the ventilator is of high-quality. Accomplishing this outcome relies on a number of important components:

- Documented software design and risk analysis that ensures the ventilator will operate in a safe manner under the foreseeable operating conditions
- Rigorous verification and validation testing
- Revision control processes that ensure documentation and consistent published software quality

This document provides a summary of the steps that have been taken to fulfill the requirements of the listed components. Software specifications and functionality documentation is provided in Sections 6 through 8. The combination of these sections provides a clear outline of the purpose and scope of the software utilized by the Alberta E-Vent. A failure mode and effects analysis is provided in Section 9 and outlines the strategies that have been implemented to minimize the risk of software and hardware related failures. A complete summary of the verification and validation testing that was conducted in collaboration with a variety of medical professional, including RRTs and MDs, is provided in Section 10. Finally, the revision control processes to be used during the maintenance period of this software are outlined in Section 11. Note that the primary intent of this document is to cover the initial development and testing of the Alberta E-Vent software and the maintenance plan outlined herein is a high-level overview only. The 1-month long time frame for the development of the software has necessitated that a fully detailed software support and maintenance plan will need to be developed at a later date.

6 Functionality Overview

The Alberta E-Vent is a positive displacement ventilator which uses feedback control to deliver a set volume of air/oxygen to a patient. The software supports two modes of ventilation:

- Assist Control (AC) Mode
- Volume Control (VC) Mode

Volume Control mode is the most basic mode offered by the ventilator. When VC mode is selected, the ventilator delivers breaths at a set rate and volume which are defined by the following properties:

1. Tidal Volume - the total volume delivered to the patient during inspirations
2. Respiratory Rate - the number of breaths delivered per minute
3. Inspiratory Time - the duration, in seconds, over which the inhalation portion of the breath is delivered

These three settings fully define the behavior of the ventilator. Note that the expiratory time and corresponding inhalation:exhalation (I:E) ratio are implicitly set by these three parameters and as such, neither exhale time or I:E ratio are offered as user adjustable settings. Full details on the operation of VC Mode are given in Section 8.3.

In Assist Control (AC) Mode, the ventilator will respond to patient triggered-breaths in addition to delivering the base-line ventilation rate set by the respiratory therapist. A patient-triggered breath occurs when the breathing circuit pressure drops by a set amount below the PEEP pressure. This set amount is user adjustable and is referred to as the trigger pressure. Note that the trigger pressure is always taken to be relative to the measure PEEP pressure and should not be considered an absolute pressure. Full details on the operation of AV Mode are given in Section 8.2.

7 Software Specifications

This section outlines the performance of the ventilator software including supported ventilation modes, user adjustable parameters, and built in system alarms and checks. Verification and validation of these specifications are provided in Section 10.

7.1 Software of Unknown Pedigree (SOUP)

TODO: need to add elapsedMillis, Roboclaw, LiquidCrystal etc.

7.1.1 Publish Anomaly Lists for SOUP

Published anomaly lists are not available for many of the open-source software libraries used. As such, verification of these libraries has relied on testing specific to the Alberta E-Vent to ensure that the functionality expected is delivered over the full range of ventilator operating conditions. Further details on the verification and validation testing performed for the ventilator can be found in Section 10.

7.2 Supported Modes

The ventilator software supports the following features and specifications:

1. Assist Control Mode
2. Volume Control Mode

7.3 Adjustable Ventilation Parameters

Parameter	**Range**	**Increment**	**Default**
Tidal Volume	50-100%*	1%	50%
Respiratory Rate	10-30 BPM	1 BPM	20 BPM
Inspiration Time	0.5-3.0 s	0.1 s	1.5 s
Trigger Pressure**	1-5 cm H_2O	1 cm H_2O	1 cm H_2O

* Corresponds to a tidal volume range of approximately 400-800 milliliters
** Only applies during AC Mode

7.4 Alarms and Fault Detection

Alarm	**Range**	**Increment**	**Default**
High PIP	30-70 cm H_2O	1 cm H_2O	40 cm H_2O
Low PIP	2-20 cm H_2O	1 cm H_2O	10 cm H_2O
High PEEP	15-35 cm H_2O	1 cm H_2O	20 cm H_2O
Low PEEP	0-10 cm H_2O	1 cm H_2O	3 cm H_2O
High Respiratory Rate*	15-40 BPM	1 BPM	35 BPM
Mechanical Failure	N/A	N/A	N/A
Homing Failure	N/A	N/A	N/A

* Only applies during AC Mode

7.5 Software and Compiler Versions

All verification and validation of the Alberta E-Vent software has been carried out using the following versions and tools:
IDE/Compiler: Arudino IDE Version 1.8.12
Alberta E-Vent Software: Version 010520

8 Architecture Description

8.1 Startup and Calibration

Upon power-up, the ventilator LCD screens will display the Albert E-Vent name and the software version for 2-seconds. The ventilator will then proceed into a calibration procedure:

- Arms will move outwards (away from the bag) until a limit switch is reached. An audible click will be heard when the limit switch is depressed by the arms.
- Arms will stop momentarily and then begin to move inwards towards the edge of the bag. They should stop just at the edge, or in slight contact, with the bag. The ventilator will stop in this position for approximately 1-second before transitioning into ventilation.
- This location at the edge of the bag is referred to as the ventilator's zero point.

The full state flow diagram for the motor calibration procedure is provided in Appendix A.

8.2 Assist Control Mode

Assist Control Mode is characterized primarily by the ability to detect patient-triggered breaths. When set to AC Mode, the ventilator will adhere to the following procedure:

- A waiting period occurs in which the ventilator monitors the breathing circuit for a drop in pressure indicative of the patient demanding a breath.
- If this patient trigger is detected, or a sufficient period of time passes such that a breath is required to maintain the baseline respiratory setpoint, the ventilator delivers a breath at the set tidal volume and inspiration time.
- After completing the inhale, the motor arms retract to the zero point location. This retraction period occurs over 20% of the nominal expiration time to allow for the arms to be sufficiently positioned to deliver another breath before the end of the nominal expiration period. Upon returning to the zero point, the ventilator cycle back to the initial waiting period.

The full state flow diagram for AC Mode operation can be found in Appendix A.

8.3 Volume Control Mode

Volume Control mode functions identically to the assist control mode except that patient triggers are not monitored for.

- The ventilator delivers a breath at the set tidal volume and inspiration time.
- The motor arms retract to the zero point location and pause for the duration of the nominal expiration period (determined based on the inspiration time and respiratory rate).
- Upon pausing for the required expiration time period, the cycle restarts and the ventilator delivers another breath.

The full state flow diagram for VC Mode operation can be found in Appendix A.

8.4 Alarm Conditions

Alarm conditions result in an audible (LED) and visual (piezoelectric buzzer) alert. Additionally, the alarm LCD screen displays the currently triggered alarm. If the alarm condition is not triggered on two subsequent breaths, the alarm auto-resets and deactivates the visual, audio and LCD indicators. The exception to this auto-reset feature are the mechanical fault and homing timeout alarms which require a motor calibration or power cycle respectively in order to reset.

An alarm silence button mutes all active alarms for a period of 2-minutes. If a new alarm occurs during this period the audio alert returns and any subsequent pushes of the alarm silence button will silence all active alarms for an additional two-minutes.

Additional details on the functioning of the alarms is provided by the state flow diagrams in Appendix A.

8.4.1 High Peak Inspiratory Pressure (PIP)

The PIP is measured at all times when the motor arms are moving inwards to deliver a breath. If the pressure in the breathing circuit exceeds the high PIP alarm set-point, the breath is immediately aborted. During a breath abort, the motor arms are commanded to stop and immediately move outwards to zero point at the edge of the bag. A high PIP alarm may be triggered at any points during the inhale phase, including the plateau pressure pause that occurs at the top of the breath. After aborting the breath, ventilation will continue with additional high PIP alarms triggering additional aborted breaths. After two consecutive breaths that do not trigger the alarm, the alarm is auto-reset.

8.4.2 Low Peak Inspiratory Pressure (PIP)

The low PIP condition is checked immediately before the pause phase at the top of the breath. This period is where the highest inspiratory pressure is found and if the measured value falls below the low PIP alarm set-point, the alarm will be triggered. Ventilation continues as normal under this alarm but operator attention is required since a persistent low PIP alarm can be indicative of a leak or disconnection in the breathing circuit. After two consecutive breaths that do not trigger the alarm, the alarm is auto-reset.

8.4.3 High Peak End-Expiratory Pressure (PEEP)

The PEEP pressure is measured at either the end of the motor return period (AC Mode) or after the nominal exhalation time (VC Mode). The measured PEEP value is then compared against the high PEEP alarm set-point. If the measured PEEP pressure exceeds the high PEEP alarm set-point, the alarm is activated. After two consecutive breaths that do not trigger the alarm, the alarm is auto-reset.

8.4.4 Low Peak End-Expiratory Pressure (PEEP)

The PEEP pressure is measured at either the end of the motor return period (AC Mode) or after the nominal exhalation time (VC Mode). The measured PEEP value is then compared against the low PEEP alarm set-point. If the measured PEEP pressure is less than the low PEEP alarm set-point, the alarm is activated. After two consecutive breaths that do not trigger the alarm, the alarm is auto-reset. Similar to the low PIP alarm, the low PEEP alarm can be indicative of a leak or disconnection.

8.4.5 High Respiratory Rate (AC Mode Only)

The respiratory rate is calculated during AC mode to account for breaths that are spontaneously triggered by the patient in addition to breaths delivered by the baseline rate set on the ventilator. A new average respiratory rate is calculated every 5-breaths. If the respiration rate calculated during this time exceeds the high respiratory alarm set-point, the alarm will be triggered. If the next 5-breaths result in a calculated respiratory rate below the set-point, the alarm is auto-reset.

8.5 Fault Conditions

Fault conditions require a departure from ventilation in order to correct. There are two fault conditions that are monitored for by the software.

8.5.1 Mechanical Failure

A mechanical failure occurs when the commanded motor position is not reached in the allocated time. Such a failure can occur for a number of reasons such as:

- Faulty wiring
- Encoder failure
- Mechanical obstructions preventing arm motion
- Motor controller failure

If a mechanical failure condition is detected, ventilation is stopped and the motor attempts to re-calibrate. If the calibration is successful, ventilation will continue. The software keeps and internal count of the number of mechanical failures that have occurred. In the event that two mechanical failure conditions occur, the ventilator immediately proceeds to the homing failure alarm which requires a full system restart to resolve.

8.5.2 Homing Failure

A homing failure alarm will occur if two mechanical failure conditions are detected during the operation of the ventilator, or if the system times out while waiting for a limit switch activation during calibration. Either of these scenarios will activate the visual and audio annunciation of the alarm and display a message on the LCD requiring the operator to power cycle the device.

Table 2: Motor Calibration Phase Risk Controls

Process Failure	Effect on Patient	Possible Cause	Control
Stuck in Prehome (outward movement not stopped)	Ventilation does not start	Limit switch stuck open	Homing timeout
		Wiring fault	Homing timeout
		Mechanical fault	Homing timeout
		Hardware fault	None
		Software fault	None
Stuck in Prehome (no outward motion)	Ventilation does not start	Motor wiring fault	Homing timeout
		Mechanical fault	Homing timeout
		Hardware fault	None
		Software fault	None
Stuck in Zeroing (continuous inward motion)	Ventilation does not start	Encoder fault	Motor position check
		Hardware fault	None
		Software fault	None
Stuck in homed position (no inward motion)	Ventilation does not start	Wiring fault	Motor position check
		Mechanical fault	Motor position check
		Hardware fault	None
		Software fault	None
Post-home position incorrect (not touching bag)	Tidal volume setting inaccurate	Encoder fault	Motor position check; visual inspection
		Mechanical fault	Motor position check
		Hardware fault	Visual inspection
		Software fault	Visual inspection
Post-home position incorrect ()inwards of bag edge)	Tidal volume setting inaccurate	Encoder fault	Motor position check; visual inspection
		Mechanical fault	Motor position check
		Hardware fault	Visual inspection
		Software fault	Visual inspection

9 Risk Management

The ventilator software is composed of a number of discrete states (See Appendix A for further details). Failure modes in each of these states have been analyzed and control measures, either external or internal to the software, have been added to mitigate risk.

9.1 Motor Calibration

Occurs at power-up and after the detection of a mechanical fault. See Section 8.1 for further details. This section examines the possible failure modes of the motor calibration phase and identifies the risk control measures that have been implemented to reduce the risk.

9.2 Inhale

Both the AC and VC Modes of operation include a inhale phase. Refer to Sections 8.2 and 8.3 for further details. This section examines the possible failure modes of the inhale phase and identifies the risk control measures that have been implemented to reduce the risk.

Table 3: Inhale Phase Risk Controls

Process Failure	**Effect on Patient**	**Possible Cause**	**Control**
Stuck in inhale (no motion)	Ventilation interrupted	Timer calculation overflow	Code review; limit run time to 14-days
		Timer not working	watchdog timer
		Wiring fault	Motor position check
		Encoder fault	Motor position check
		Mechanical fault	Low PIP alarm
		Software fault	None
Inhale duration shorter (due to timing)	Higher PIP	Embedded timer runs fast	High PIP alarm
		Motor controller fault	High PIP alarm
		Motor too fast due to software fault	PIP relief valve
Inhale duration shorter (due to position)	Reduced tidal volume	Motor stops before desired position	Motor position check
Inhale duration longer (due to timing)	Lower PIP; higher PEEP	Embedded timer runs slow	Watchdog timer
		Motor controller fault	High PEEP alarm
		Software fault	PEEP valve
Inhale duration longer (due to position)	Tidal volume setting inaccurate	Motor exceeds desired position	Motor position check
Other inhale failures	Ventilation interrupted	Motor fault	Motor position check
		Wiring fault	Motor position check
		Mechanical fault	Motor position check
		Encoder fault	PIP and PEEP alarms
		Bag failure	PIP and PEEP alarms

Table 4: Exhale Phase Risk Controls

Process Failure	**Effect on Patient**	**Possible Cause**	**Control**
Stuck in exhale (no motion)	Ventilation interrupted	Timer calculation overflow	Code review; limit run-time to 14-days
		Timer not working	watchdog timer
		Wiring fault	Motor position check
		Encoder fault	Motor position check
		Mechanical fault	High PEEP alarm
		Software fault	None
Exhale duration shorter (due to position)	Asynchronous breathing	Embedded timer runs fast	High PEEP alarm
		Motor controller fault	Motor position check
		Software fault	High PEEP alarm
Exhale duration shorter (due to timing)	No effect	Software or controller fault	None
Exhale duration longer (due to position)	Inaccurate tidal volume on next breath	Mechanical fault	Motor position check
		Motor controller fault	Motor position check
		Software fault (arms too wide)	Visual inspection required
Exhale duration longer (due to timing)	Inaccurate tidal volume on next breath	Embedded timer runs slow	Watchdog timer
Other exhale failures	Ventilation interrupted	Motor fault	Motor position check
		Wiring fault	Motor position check
		Mechanical fault	Motor position check
		Encoder fault	Visual inspection required

9.3 Exhale

Both the AC and VC Modes of operation include a exhale phase. Refer to Sections 8.2 and 8.3 for further details. This section examines the possible failure modes of the exhale phase and identifies the risk control measures that have been implemented to reduce the risk.

9.4 Alarms and Fault Detection

The ventilator includes a number of alarms. Refer to Sections 8.4 and 8.5 for further details. This section examines the possible failure modes these alarms and identifies the risk control measures that have been implemented to reduce the risk.

Table 5: Alarm and Fault Detection Risk Controls

Alarm Failure	**Effect on Patient**	**Possible Cause**	**Control**
Mechanism failure not detected	Inaccurate tidal volume and/or inspiration time	Mechanical fault affecting arm motion but not the encoder reading	Visual inspection required
High PIP alarm condition not detected	Risk of barotrauma	Faulty pressure sensor	Mechanical high pressure relief valve
Low PIP alarm condition not detected	Insufficient ventilation	Faulty pressure sensor	None (blood gases monitoring)
High PEEP alarm condition not detected	Persistent occurrence could lead to pnemeauthorax	Faulty pressure sensor	Mechanical PEEP valve
Low PEEP alarm condition not detected	Collapse of patient lungs	Faulty pressure sensor	Mechanical PEEP valve
High respiratory rate not detected	TODO	TODO	TODO

10 Verification and Validation

TODO: COMPLETE THIS SECTION

10.1 Servo Lung Testing

To be completed with finalized software.

10.1.1 Ventilation Testing

To be completed with finalized software.

10.1.2 Alarm Testing

To be completed with finalized software.

10.1.3 Clinical Feedback

To be completed with finalized software.

10.2 Longevity Testing

To be completed with finalized software.

10.3 Verification of Risk Control Methods

10.4 Anomalies

11 Maintenance

Software maintenance may become necessary if the released software has unacceptable anomalies identified. The risk of this occurring has been minimized through extensive testing seeking to verify and validate the performance of the ventilator. Further details on this testing can be found in Section 10. The aim of this section is to outline the procedure that will be followed if software modifications have to be made to the released software.

11.1 Software Problem Identification

Any software anomalies identified will need to undergo a multi-step process to determine the severity of the anomalies and the corrective action required.

- Evaluate the risk to patient health posed by the anomaly. Issues deemed high-risk must be rectified immediately.
- Identify the phase of the ventilator operation the anomaly is associated with. Refer to state flow diagrams as required.
- Correct the issue in the associated code based on the phase of operation identified above.
- Ensure the entire functionality of the ventilator has been maintained using a holistic test procedure.

11.2 Verification and Release of Modified Software

Any modification to the Alberta E-Vent software will required a complete repeat of the verification tests performed for the initial release. Only after a succesful completition of this full suite of tests will any modified software be released. Revision tracking will be maintained by utilizing the following procedures:

- All software revisions will be stored in a repository such as GitHub
- Each software revision that is released will be assigned a unique software version code based on the date it is published (see section 5)
- Each software revision will require the following documentation:
  - Date of published revision
  - Unique revision identifier that is displayed at startup
  - Location of the revision in the code repository
  - Reason for revised code, including supporting documentation such as error reports or risk analysis documents
  - Summary of revised code, including supporting documentation such as state flow diagrams
  - Originator of the revised code
  - Independent checker of the revised code
  - Documentation of the verification testing procedure followed to confirm performacne of the ventilator is maintained after the modification have been implemented
  - Final approver of the revised code

An example template to be used for this revision and change tracking procedure is provided in Appendix C.

Appendix A Software Flow Charts

State flow diagrams summarize the behavior of the software. Each diagram provides an overview of a particular aspect of the program.

1. The overall program state flow diagram is shown in Fig. A.2.
2. The motor homing and zeroing (calibration) state flow diagram is shown in Fig. A.3.
3. The Assist control Mode state flow diagram is shown in Fig. A.4.
4. The Volume Control Mode state flow diagram is shown in Fig. A.5.

Note that the combined behavior of all of these state flow diagrams is necessary for a complete understanding of how the software interacts.

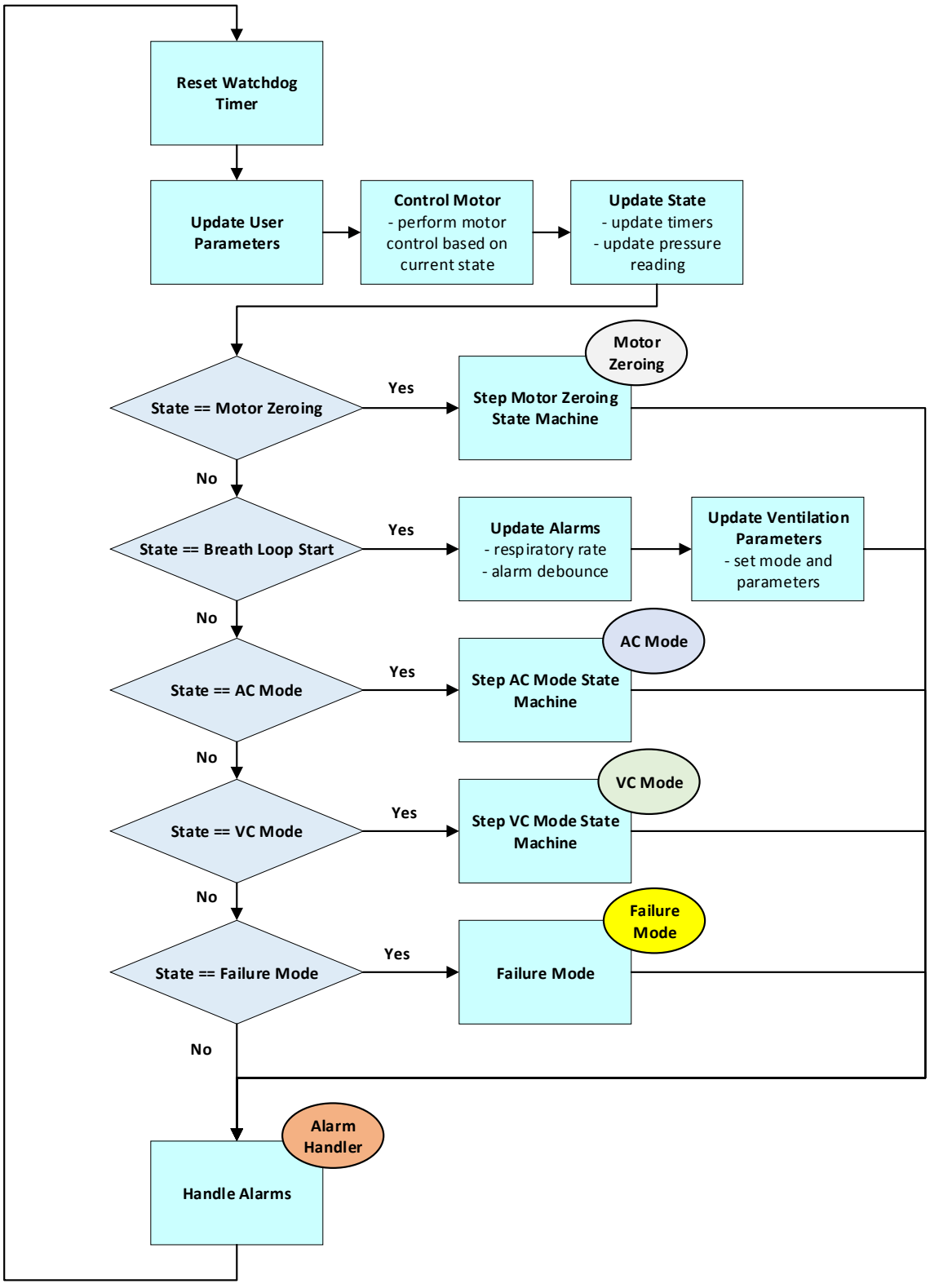

Figure A.1: Main Program Loop

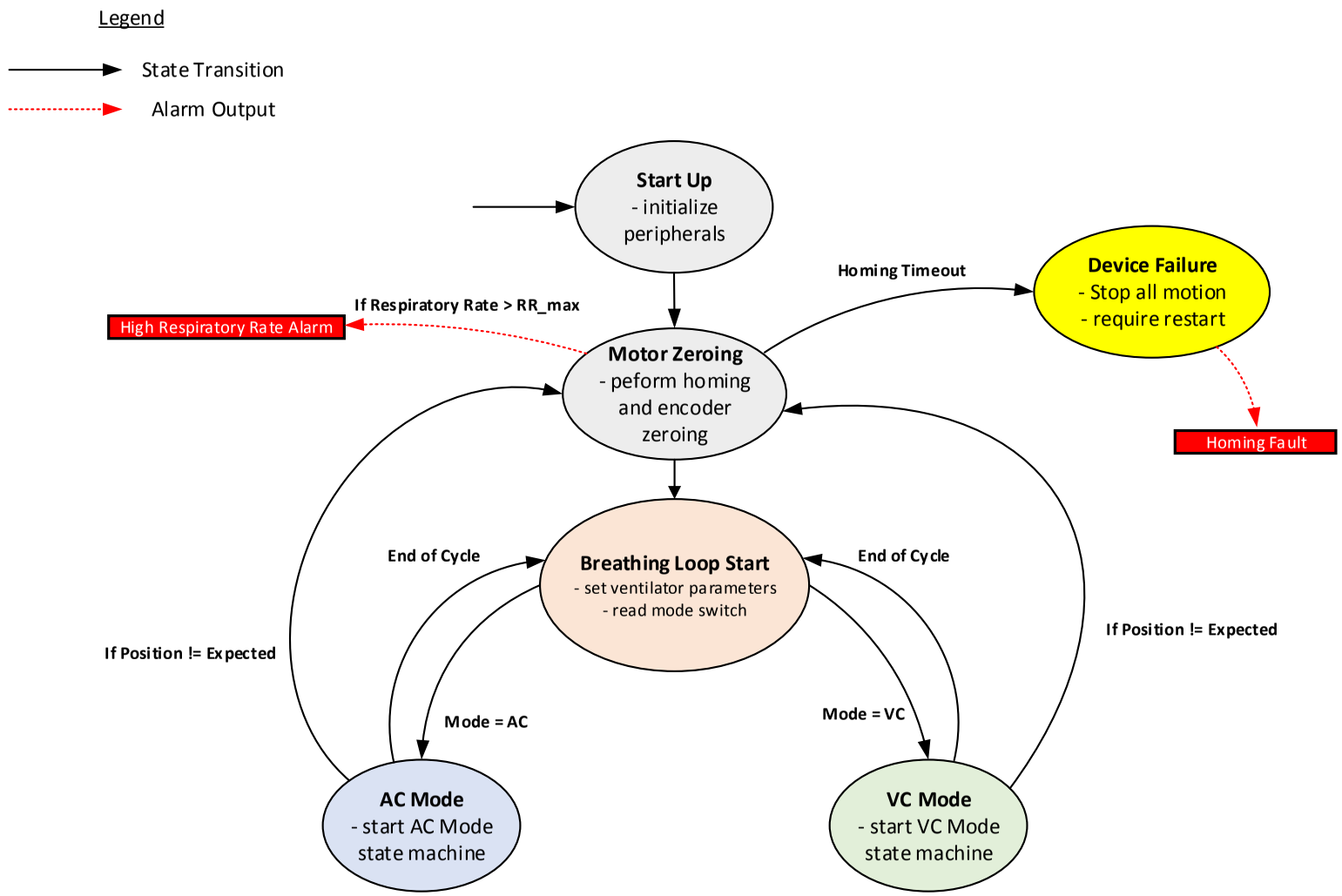

Figure A.2: Overall Program State Flow Diagram

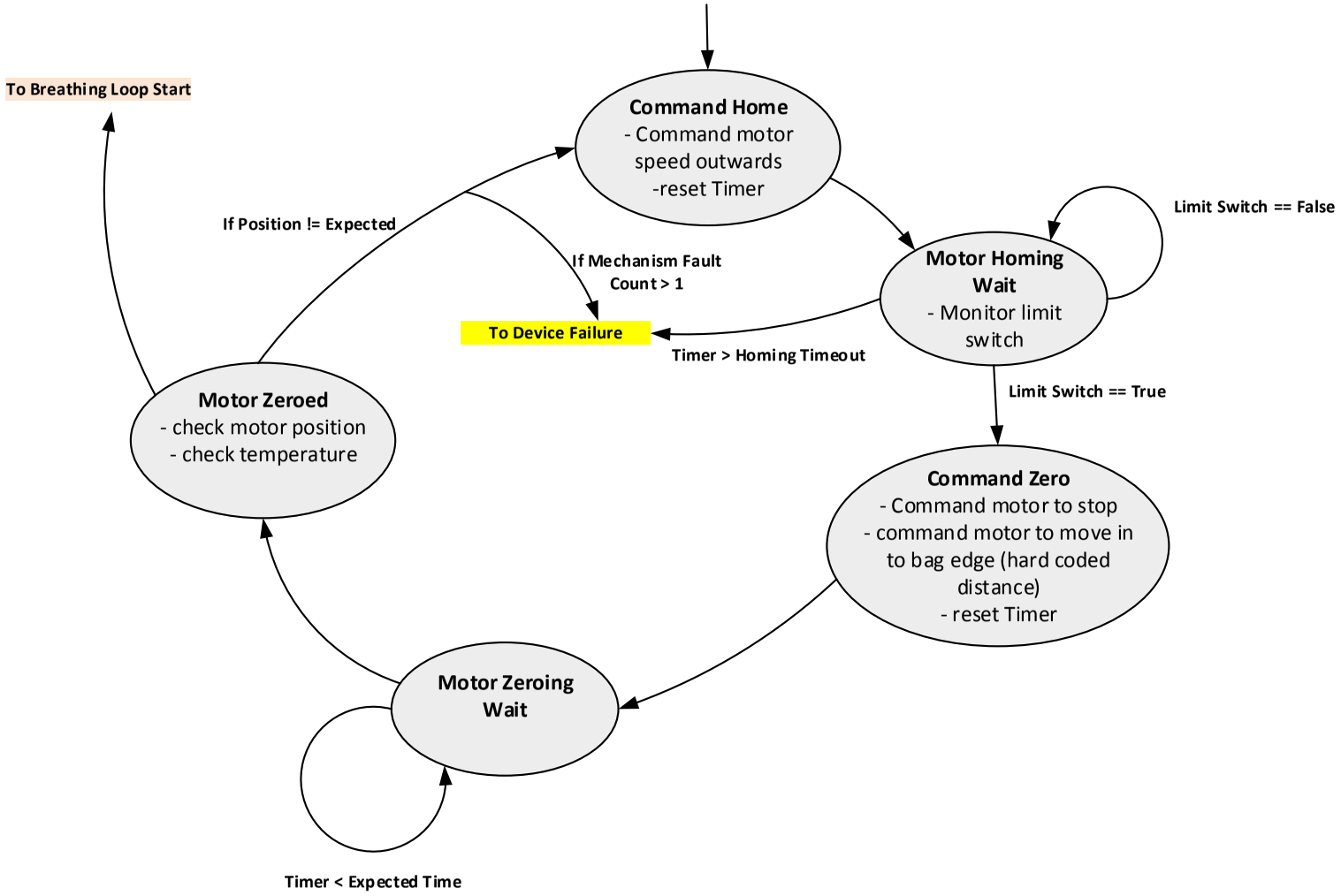

Figure A.3: Motor Zeroing State Flow Diagram

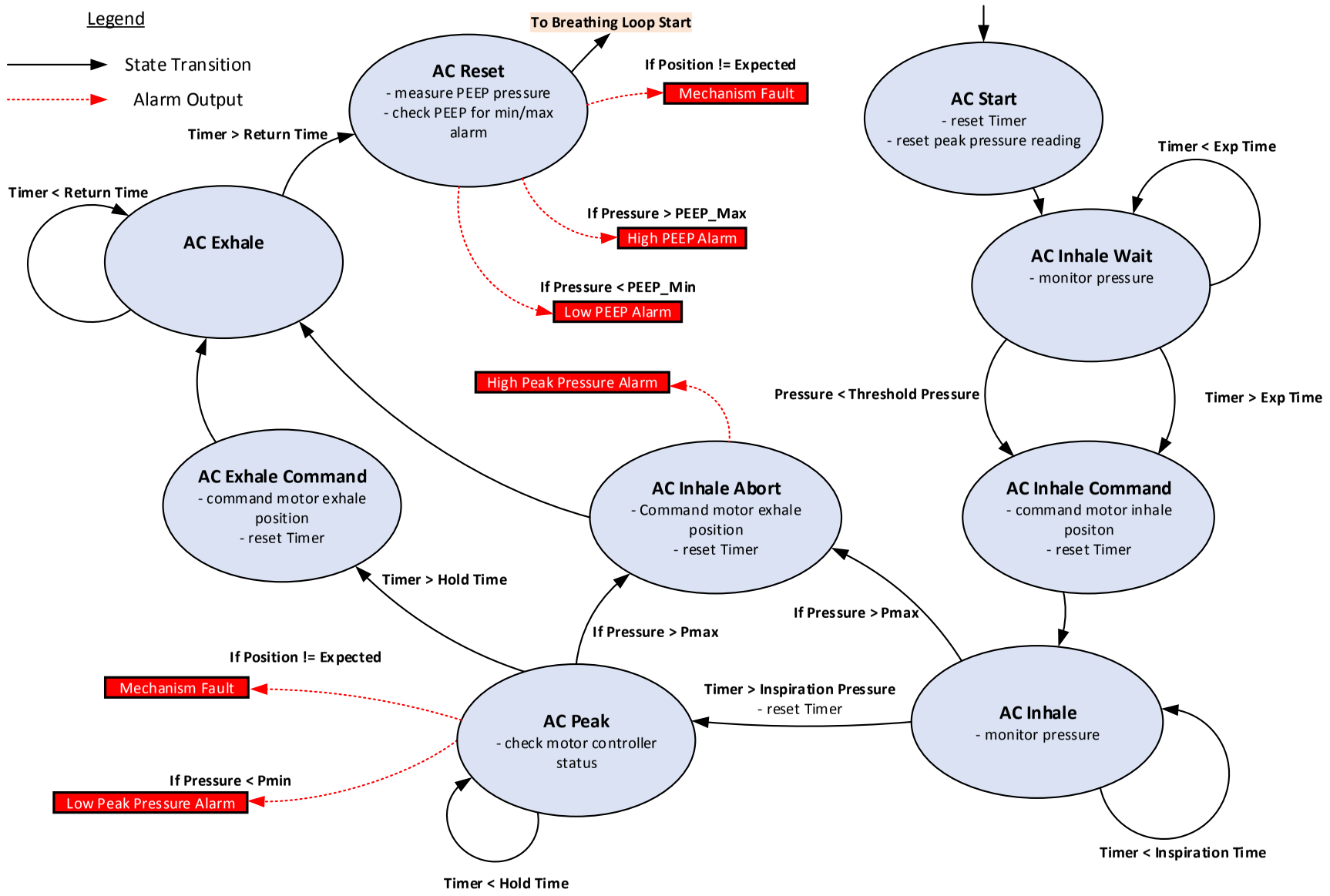

Figure A.4: Assist Control Mode State Flow Diagram

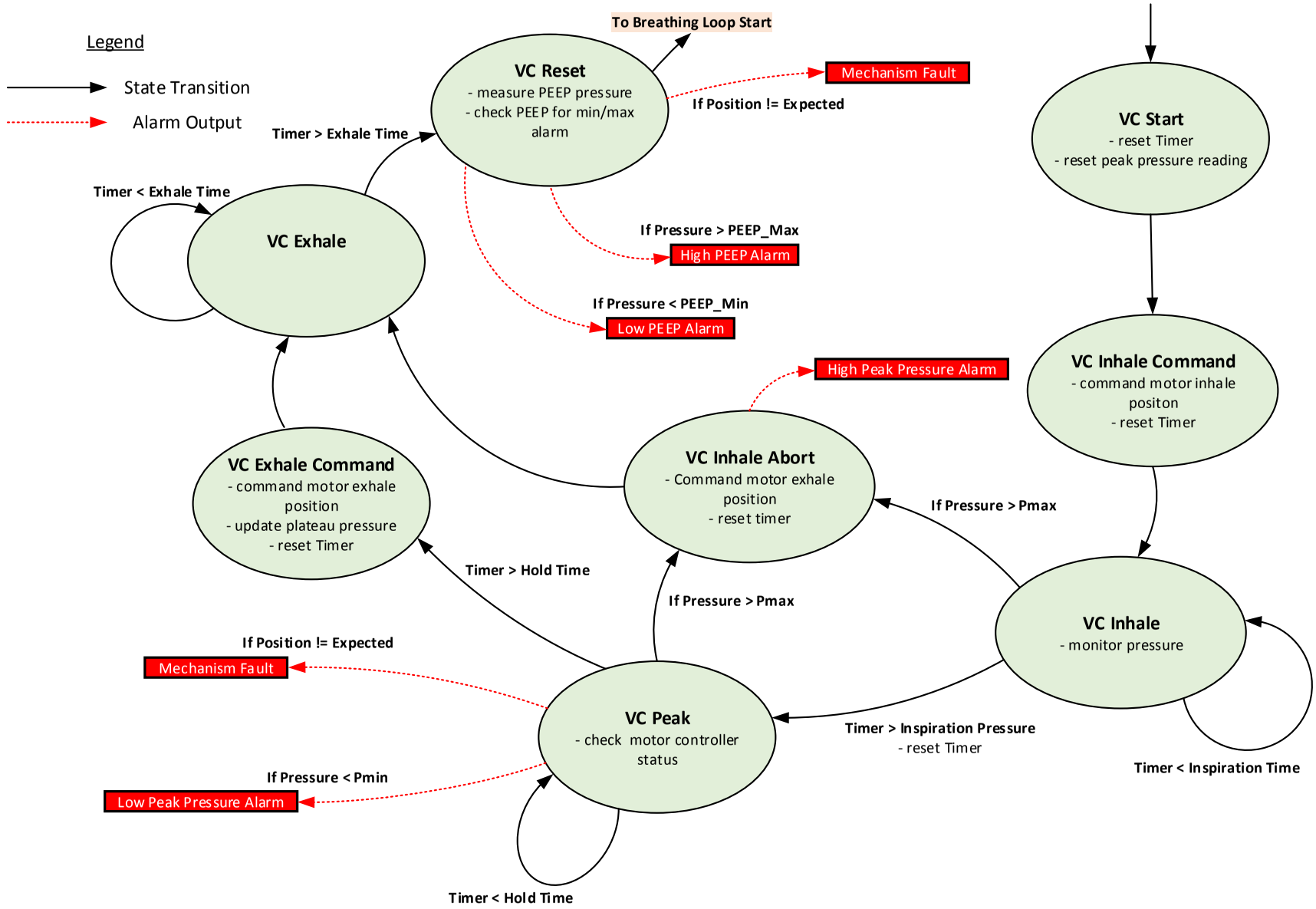

Figure A.5: Volume Control Mode State Flow Diagram

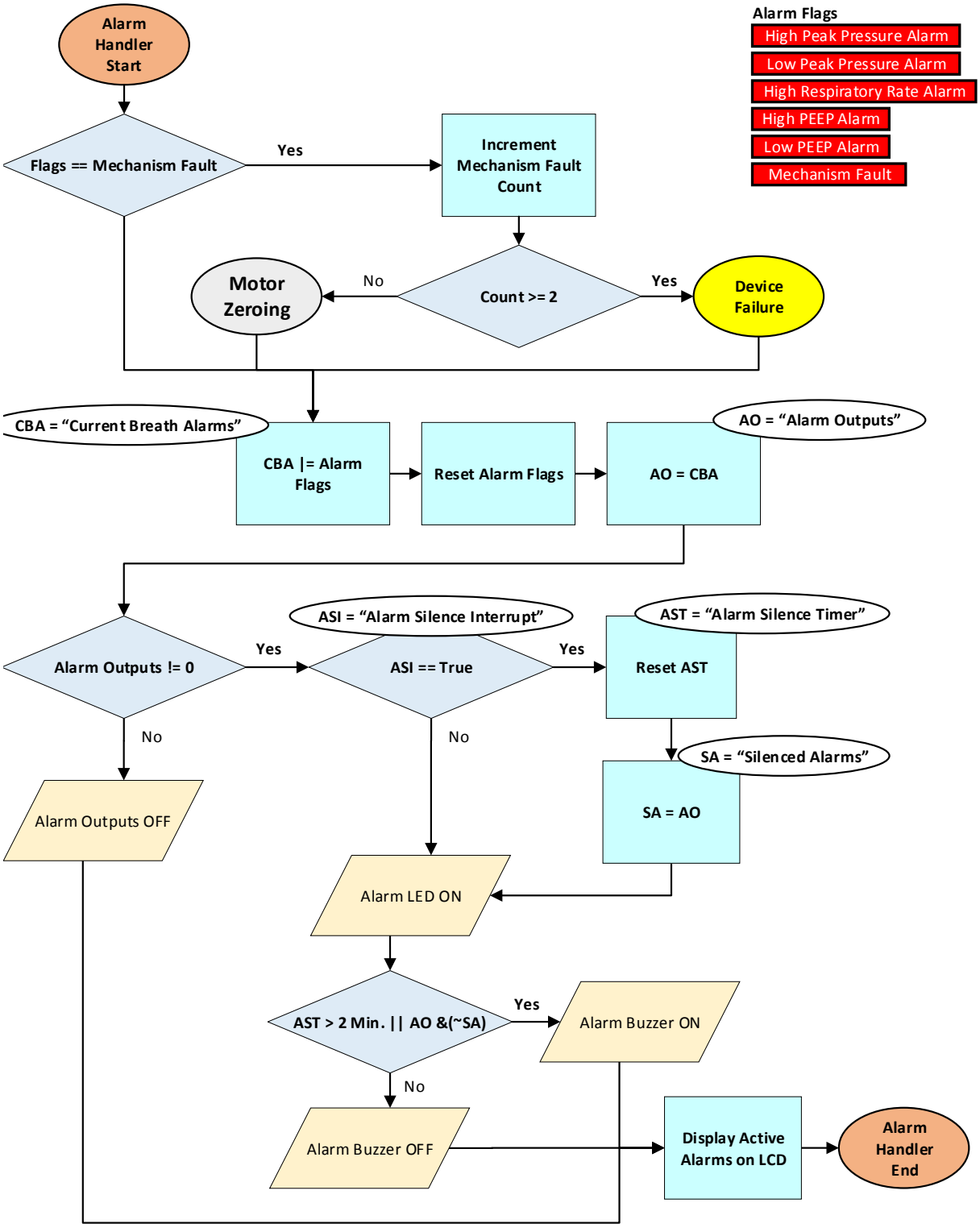

Figure A.6: Alarm Handler

Appendix B Risk Assessment

The figures contained in this section outline the overall risk assessment of the ventilator system. Not all of the analysis in this section pertains to the software aspect of the ventilator. For an examination of the software-specific risk mitigation implemented refer to Section 9 of this document. This appendix provides the following Figures:

- Figure B.1 provides the risk matrix utilized in evaluation of the risk associated with the ventilator system. The risk matrix employed is based on the ISO 14971 standard.
- Figures B.2 through B.4 outline the unmitigated risk associated with the ventilator.
- Figures B.5 through B.8 detail the risk mitigation steps that have been taken to minimize risk.

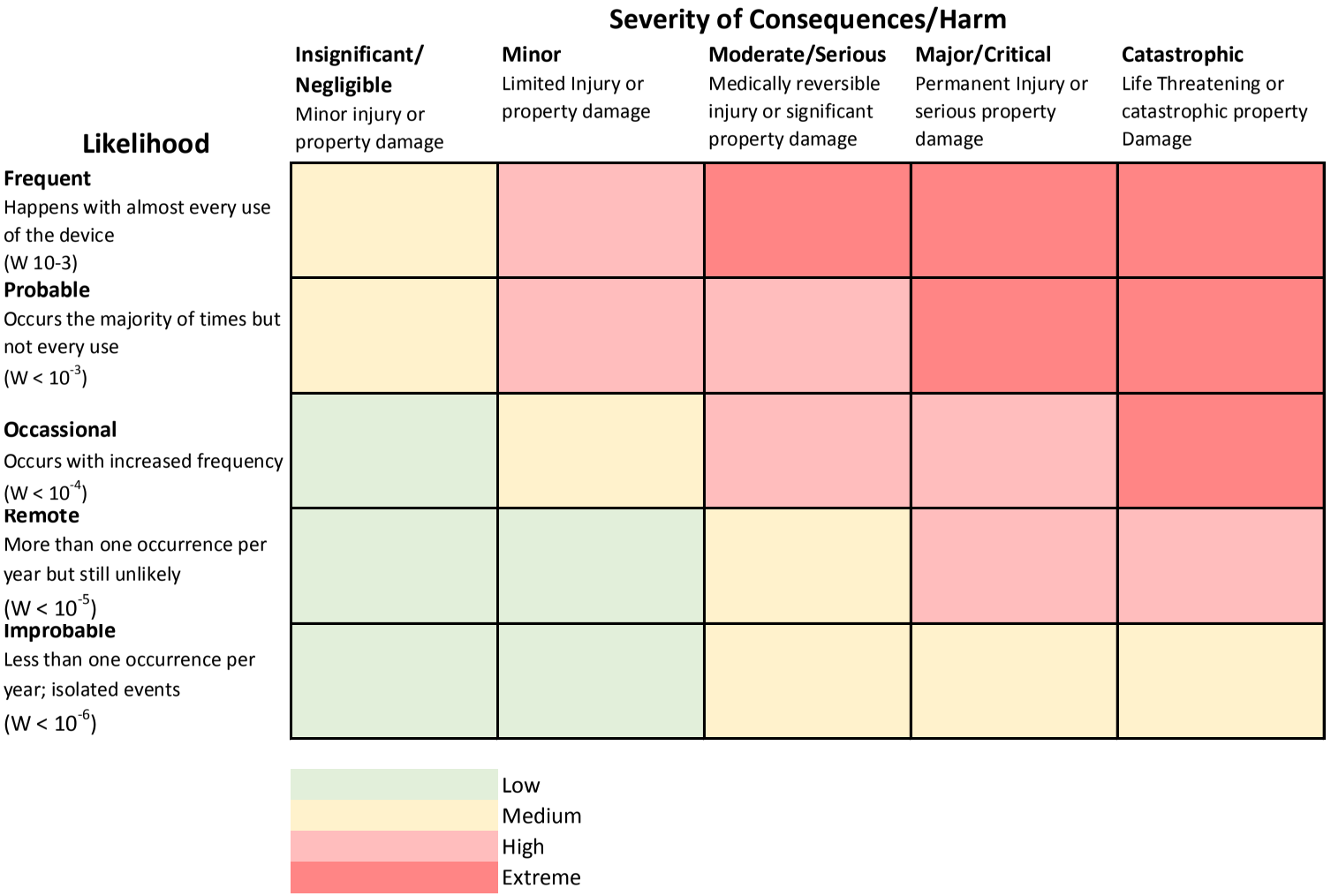

Figure B.1: Risk Matrix based on ISO 14971

Risk Description	Causes	Consequences	Consequence	Likelihood	Overall Unmitigated Risk
Patient circuit disconnect	- Accidental disconnection - Patient movement - Ventilator falls from table	- Ventilation Interrupted, patient not receiving adequate ventilation - if left uncorrected long enough, can lead to possible injury or death	Catastrophic	Occassional	Extreme
Bag rupture	- pre-existing damage - excessive wear and abrasion	- Ventilation Interrupted, patient not receiving adequate ventilation - if left uncorrected long enough, can lead to possible injury or death	Catastrophic	Remote	High
Bag no longer reinflating	- repetitive movement induced plastic fatigue - age of bag (older bags tend to stiffen)	- Ventilation Interrupted, patient not receiving adequate ventilation - if left uncorrected long enough, can lead to possible injury or death	Catastrophic	Occassional	Extreme
Oxygen leak or not enough O2	- incorrectly attached oxygen hose - damaged hose to connection - bag O2 reservoir damaged - incorrect O2 blending	- lower oxygen to patient than expected - possible fire hazard with potential spark from motor contacts	Critical	Remote	High
EMI emissions from motor and electronics interacting with surrounding equipment	- EMI emissions from the motor or electronics	- interference with life support systems could cause serious injury or death	Catastrophic	Remote	High
EMI interference with controller	- EMI emissions from the motor or surrounding medical equipment causing a failure of the electronics	- Possible ventilation Interrupted, patient not receiving adequate ventilation - if left uncorrected long enough, can lead to possible injury or death	Catastrophic	Remote	High
Power loss (unit has no backup battery installed)	- Damaged power supply - Loss of main power (and back-up power) - Accidental disconnection of cable - Accidental switched off (bumped into switch)	- Ventilation Interrupted, patient not receiving adequate ventilation - if left uncorrected long enough, can lead to possible injury or death	Catastrophic	Occassional	Extreme

Figure B.2: Unmitigated Risk Analysis

Risk Description	Causes	Consequences	Consequence	Likelihood	Overall Unmitigated Risk
Power loss (unit has no backup battery installed)	- Damaged power supply - Loss of main power (and back-up power) - Accidental disconnection of cable - Accidental switched off (bumped into switch)	- Ventilation Interrupted, patient not receiving adequate ventilation - if left uncorrected long enough, can lead to possible injury or death	Catastrophic	Occassional	Extreme
Gear drive damage	- erosion of the gears overtime will create slack. - foreign object gets into the gears or between arms	- lower tidal volume than expected - possible total mechanical failure, and inability to home arms, leading to ventilation interruption, patient not receiving adequate ventilation.	Serious	Occassional	High
Motor failure	- damage to motor (e.g. overheated, power surge, controller failure) - defective motor - encoder damaged	- Ventilation Interrupted, patient not receiving adequate ventilation - if left uncorrected long enough, can lead to possible injury or death	Catastrophic	Remote	High
Control Box Electronics Failure (see also PFMEA) - Arduino Controller failure	- damage to card (e.g. overheated, power surge) - defective component	- Ventilation Interrupted, patient not receiving adequate ventilation - if left uncorrected long enough, can lead to possible injury or death	Catastrophic	Occassional	Extreme
Accidental change in settings	- tired/fatigued clinicians - hit the wrong button by accident	- worst case loss of ventilation or too high pressure - if left uncorrected long enough, can lead to possible injury or death	Catastrophic	Ocassional	Extreme
Pinch points	- loose clothing or hair can get caught in the gears / mechanism - fingers caught in the gears	- injury to fingers - serious injury in the case of hair or clothing	Serious	Remote	Medium

Figure B.3: Unmitigated Risk Analysis

Risk Description	Causes	Consequences	Consequence	Likelihood	Overall Unmitigated Risk
Bag displacement (bag moves off the holders)	- repetitive movement causing bag to move (not experienced during testing) - bag not placed properly or secured properly	- Ventilation Interrupted, patient not receiving adequate ventilation - if left uncorrected long enough, can lead to possible injury or death	Catastrophic	Remote	High
Mechanism falls from table or cart	- repetitive movement causing mechanism to move (not experienced during testing) - mechanism is bumped accidentally	- Ventilation Interrupted, patient not receiving adequate ventilation - if left uncorrected long enough, can lead to possible injury or death	Catastrophic	Remote	High
High PIP Pressure	- Obstruction - Cough; - Deterioration of patient status - Pressure sensor failure - Plugged or damaged pressure sensing hose - Modified PEEP valve (precompressing spring) as a pressure relief valve fails - Software coding error or library	Exceeding H Alarm causes risk of barotrauma (lung damage)	Critical	Probable	Extreme
Software code failure - alarms fail to sound	- Error in software coding - Two Arduino libraries are not possible to review and may contain a bug (although they are widely used and tested). - Device will not have been long term tested to catch buffer overflow and similar code issues	- Exceeding H Alarm causes risk of barotrauma (lung damage) - Ventilation Interrupted, patient not receiving adequate ventilation - if left uncorrected long enough, can lead to possible injury or death	Catastrophic	Probable	Extreme

Figure B.4: Unmitigated Risk Analysis

Risk Description	Risk Control Mitigations (RCM)	Consequence	Likelihood	Residual Risk	Comments and Recommendations
Patient circuit disconnect	1. Pressure sensor identifies loss of pressure; Alarm will sound and light. 2. Close supervision is advised and convert to manual BVM	Catastrophic	Remote	High	No further risk reduction with device, ALARP. recommend clinical procedures to prevent patients from disconnecting. Likely the same with more advanced ventilators See pressure sensor failure risk
Bag rupture	1. Pressure sensor identifies loss of pressure; Alarm will sound and light. 2. Close supervision is advised and convert to manual BVM 3. User Manual procedure of visually inspecting bag on a regular basis, to ensure bag is not damaged. Unlikely to occur quickly	Catastrophic	Improbable	Medium	No further risk reduction necessary, but must enable controls and procedures
Bag no longer reinflating	1. Pressure sensor identifies loss of pressure; Alarm will sound and light. 2. close supervision is advised and convert to manual BVM 3. User Manual procedure of visually inspecting bag on a regular basis, to ensure bag is not damaged. Unlikely to occur quickly	Catastrophic	Improbable	Medium	No further risk reduction necessary, but must enable controls and procedures
Oxygen leak or not enough O2	1. This risk is native to the manual resuscitation bag and is addressed by the bag instructions and existing hospital procedures and training. 2. The bag is exposed and any leaked oxygen will dissipate 3. Blood gases are measured and will show declining oxygen levels 4. User manual indicates that FiO2 will be manually sampled on a regular basis using hand held probe (hospital procedures)	Critical	Improbable	Medium	No further risk reduction necessary, but must enable controls and procedures

Figure B.5: Mitigated Risk Analysis

Risk Description	Risk Control Mitigations (RCM)	Consequence	Likelihood	Residual Risk	Comments and Recommendations
EMI emissions from motor and electronics interacting with surrounding equipment	1. Formal EMI testing conducted to ensure low emissions 2. Electronics are contained in steel enclosure 3. Low voltage system with a low motor load	Catastrophic	Improbable	Medium	No further risk reduction necessary, but must enable controls and procedures If formal testing shows higher than acceptable emissions, consider a faraday cage on motor
EMI interference with controller	1. Board traces have been minimized and grounded 2. Controller is in a metal enclosure to shield from outside interference 3. Close supervision is advised and convert to manual BVM	Catastrophic	Improbable	Medium	No further risk reduction necessary, but must enable controls and procedures
Power loss (unit has no backup battery installed)	1. Power loss alarm, which will sound for 30 seconds 2. Power switch has a guard to avoid accidental bumping 3. Power supply is of medical grade and tested in manufacturing for defects 4. Close supervision is advised and convert to manual BVM 5. Power loss alarm is part of user manual checks	Catastrophic	Improbable	Medium	No further risk reduction with the device, ensure procedures and supervision is in place to hear the alarm If further risk reduction is desired or plans to move patient on ventilator, consider use of an uninterruptible power supply (UPS) with an alarm
Gear drive damage	1. Mechanical failure alarm and/or homing failure alarm will sound and light, indicating damage 2. If arms stop moving completely it can be easily seen visually from a distance 3. User manual has a step to check arms for looseness, indicating gear slack (i.e. erosion) 4. Close supervision is advised and convert to manual BVM	Serious	Remote	Medium	No further risk reduction necessary, but must enable controls and procedures
Motor failure	1. System will not arrive at a desired position; Encoder will detect failure to reach position; Homing alarm will sound 2. Close supervision is advised and convert to manual BVM	Catastrophic	Improbable	Medium	No further risk reduction necessary, but must enable controls and procedures

Figure B.6: Mitigated Risk Analysis

Risk Description	Risk Control Mitigations (RCM)	Consequence	Likelihood	Residual Risk	Comments and Recommendations
Control Box Electronics Failure (see also PFMEA) - Arduino Controller failure	1. Power loss alarm (which requires Arduino to be functioning), which will sound for 30 seconds 2. Close supervision is advised and convert to manual BVM 3. Arduino controller will be tested during QA/QC testing 4. Power loss alarm is part of user manual checks	Catastrophic	Improbable	Medium	No further risk reduction necessary, but must enable controls and procedures
Accidental change in settings	1. Changing the setting is a three part step - 1. press to select parameter, 2. encoder dial to desired value on screen, 3. press to confirm - but does not affect a physical change until confirm button is pressed 2. All controls will be clearly labeled and steps 1-2-3 are clear as well	Catastrophic	Improbable	Medium	No further risk reduction necessary, but must enable controls and procedures
Pinch points	1. Mechanism and gears are covered; Arms project through a slot 2. User manual procedures 3. Emergency Stop on the panel will be accessible to "hit the big red button" to stop the mechanism	Serious	Improbable	Low	No further risk reduction necessary, but must enable controls and procedures, including tying back hair and no loose fitting clothing.
Bag displacement (bag moves off the holders)	1. Pressure sensor identifies loss of pressure; Alarm will sound and light. 2. close supervision is advised and convert to manual BVM 3. User Manual procedure of visually inspecting bag on a regular basis, to ensure bag is not damaged. Unlikely to occur quickly 4. elastic bands are used to secure the bag in place	Catastrophic	Improbable	Medium	No further risk reduction necessary, but must enable controls and procedures
Mechanism falls from table or cart	1. Rubber feet are attached to the mechanism to prevent slippage. 2. Alarms for Mechanical failure, low pressure, and potentially loss of power will all sound and light 3. close supervision is advised and convert to manual BVM	Catastrophic	Improbable	Medium	No further risk reduction necessary, but must enable controls and procedures

Figure B.7: Mitigated Risk Analysis

Risk Description	Risk Control Mitigations (RCM)	Consequence	Likelihood	Residual Risk	Comments and Recommendations
High PIP Pressure	1. High PIP Alarm will sound, and auto reset with one complete breathing cycle with no over pressure 2. QA/QC on pressure sensor in manufacturing to ensure working 3. change out hoses with new patients, and inspect pressure sensing lines on occasion 4. HME filter prevents plugging material from entering the pressure sensing lines 5. Pressure relief valve will protects over pressure. Each unit is tested and clearly marked to avoid confusion with standard PEEP valves. Spring ajust knob (lid) will be glued in place to prevent tampering with the spring compression.	Critical	Improbable	Medium	No further risk reduction necessary, but must enable controls and procedures
Software code failure - alarms fail to sound	1. Code is being verified and validated (however no long term testing is possible with short time line) using testing procedure to ensure accuracy and stability, including alarms 2. close supervision is advised and convert to manual BVM 3. We will do some long term operation once the production model is in manufacturing to identify any issues before the units ship. 4. Arduino libraries are very common and have been user tested 5. If Arduino card crashes to the point requiring reboot, the alarm will sound	Catastrophic	Improbable	Medium	With any software code, there can be errors, however the software will be verified and tested to reduce likelihood of failure. It is important that the devices be closely monitored, especially upon the first long term operation in a clinical setting. Any found code defects will be corrected with a recall and a code fix uploaded to the Arduino. There is an opening in the control panel case to insert a USB cable to update from a laptop.

Figure B.8: Mitigated Risk Analysis

Appendix C Software Change Request Tracking

The figures contained in this section are examples of the documentation that will used to track changes made to the Alberta E-Vent software. These records will be maintained as part of the software repository and will provide traceability for all modifications that are made to the software.

Date	Revision #	Development Branch Name	Reason for Change Document #	Summary of Changes Document #	Originator	Checker	Test Protocol Followed	Approver

Figure C.1: Software Change Request Tracking