EECS4312 Isolette Assignment

Hovhannes Khachikyan (hovokhc@cse.yorku.ca)
Varsha Ragavendran (varshar@cse.yorku.ca)

November 5, 2017

Prism account used for submission: varshar

- You may work on your own or in a team of no more than two students. You must provide both names and Prism logins in the specified space above this.
- **Submit only one document under one Prism account.**
- Keep track of your revisions in the table below.

Requirements Document: Temperature control for an Isolette

Revisions

Date	Revision	Description
22 October 2017	1.0	Initial requirements document
??	2.0	Add more here if needed

Contents

List of Figures

List of Tables

1. System Overview

The System Under Development (SUD) is a computer controller for the thermostat of an Isolette.[1] An Isolette is an incubator for for an infant that provides controlled temperature, humidity and oxygen (Fig. 3). Isolettes are used extensively in Neonatal Intensive Care Units for the care of premature infants.

This requirements document is specifically for the control of temperature. The purpose of the Isolette computer controller is to maintain the air temperature of an Isolette within a desired range. It senses the current temperature of the Isolette and turns the heat source on and off to warm the air as needed. If the temperature falls too far below or rises too far above the desired temperature range, it activates an alarm to alert the nurse. The system allows the nurse to set the desired temperature range and to set the alarm temperature range outside the desired temperature range of which the alarm should be activated. This requirements documents follows the specification in [**?**] (Appendix A) except where noted.

Figure 1: Isolette

Many babies have dies due to faulty incubators. There is thus a standard that manufacturers must satisfy. Modern incubators are equipped with alarms for air temperature, skin temperature, oxygen concentration and humidity. The alarms are both visual such

[1] The image in Fig 3 is from: `www.nufer-medical.ch`.

as red warning lamps, and audio such as beep signals. Once measured measured values exceed permitted limits as well as when faults occur in sensors. For one such incident leading to death see "Medical Devices: Use and Safety" shown in Fig. 2.

CASE 6:2 Baby dies through overheating in incubator

An underdeveloped baby was being treated in an incubator with skin temperature control. When the baby was being washed, the skin sensor was removed and left hanging outside the incubator after the washing. Thus the sensor started measuring the room temperature (approx. 25°C). The control circuits therefore increased the heat to maximum level, and the temperature in the incubator rose to more than 45°C. The baby died.

For increased safety, incubators must be constructed with an extra control circuit that prevents overheating in case the skin sensor is misplaced. The incubator in question was indeed equipped with such a safety circuit, but the circuit was defective.

Figure 2: Incubator Safety Problems [**?**, p98]

2. Goals

The high-level goals (G) of the system are:

- G1—The Infant should be kept at a safe and comfortable temperature.
- G2—The Nurse should be warned if the Infant becomes too hot or too cold.
- G3—The cost of manufacturing the computer controller for the thermostat should be as low as possible.

3. Context Diagram

See Fig. A-1 in [**?**]. The System Under Description (SUD) is a computer *controller* to regulate the temperature of the Isolette. Everything else including the Operator Interface (described in [**?**]) is in the ecosystem (i.e. in the environment of the controller). The monitored variables and controlled variables for the controller are in Table 1 and Table 2, respectively. For clarity, simplicity and safety, there are some differences between the specifications in this document and the descriptions in [**?**].[2]

Placeholder for your Context Diagram

[2]Documented in the write-up to this assignment: `assign1-spec.pdf`.

4. Monitored Variables

The monitored variables are a subset of those described in [?].[3] There is a single status variable *m_st* that is *invalid* whenever any one of the operator inputs or temperature sensor are in a failed state. Otherwise types and ranges are as in [?].

Name	Type	Range	Units	Physical Interpretation
m_tm	$\mathbb{R}$	68.0 .. 105.0	°F	actual temperature of Isolette air temperature from sensor
m_dl	$\mathbb{Z}$	97 .. 99	°F	desired lower temperature set by operator
m_dh	$\mathbb{Z}$	98 .. 100	°F	desired higher temperature set by operator
m_al	$\mathbb{Z}$	93 .. 98	°F	lower alarm temperature set by operator
m_ah	$\mathbb{Z}$	99 .. 103	°F	higher alarm temperature set by operator
m_st	Enumerated	{valid, invalid}		status of sensor and operator settings
m_sw	Enumerated	{on, off}		switch set by operator

Table 1: Monitored Variables

[3] With some change of nomenclature. Monitored variables have an "m" prefix.

5. Controlled Variables

Ensure that the table below is complete.

The controlled variables are a subset of those described in [?].[4] In addition, there is a mode display *c_md* and a message display *c_ms*.[5]

Name	Type	Range	Units	Physical Interpretation
c_hc	Enumerated	{on, off}		heat control: command to turn heat source on or off
c_td	$\mathbb{Z}$	$\{0\} \cup \{68 \,..\, 105\}$	°F	displayed temperature of Isolette (zero when Isolette is off)
c_al	Enumerated	{off, on}		sound alarm to call nurse
c_md	Enumerated	{off, init, normal, failed}		mode of Isolette operation (failed if $m\_st = invalid$)
c_ms	Enumerated			messages to display to nurse

Table 2: Controlled Variables

[4]With some change of nomenclature. Controlled variables have a “c” prefix.

[5]The mode “off” is added to that of Fig. A-4 in [?], and the mode transitions have been changed.

6. Mode Diagram

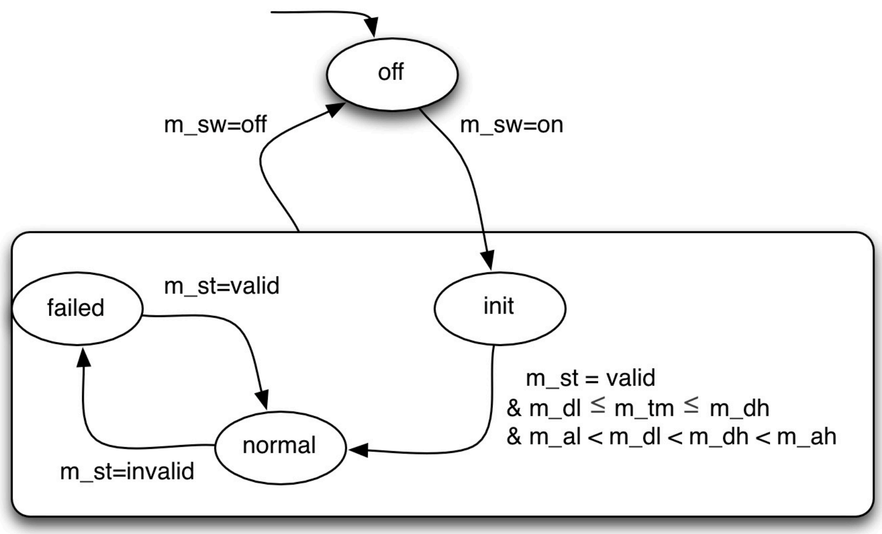

Figure 3: Statechart for the modes variable c_md

TODO: Provide rationale for the statechart.

7. R-Descriptions

We have already elicited the following R-descriptions.

REQ1	The *controller* shall operate in one of four modes: *off*, *init*, *normal* and *fail*.	See statechart in Fig. ??.

REQ2	In the *normal* mode, the temperature controller shall maintain current temperature inside the Isolette within a set temperature range (the *desired* range).	The *desired* temperature range is *m_dl* .. *m_dh*. If the current temperature *m_tm* is outside this range, the controller shall turn the heater on or off via the controlled variable *m_hc* to maintain the desired state.

Rationale: The *desired temperature range* will be set by the nurse to the desired range based on the infant's weight and health. The controller shall maintain the current temperature within this range under normal operation.

The following relevant hazard was identified through the safety assessment process:

- **H1**: Prolonged exposure of Infant to unsafe heat or cold;
- *Classification*: catastrophic;
- *Probability*: $< 10^{-9}$ per hour of operation.

To ensure that probability of hazard H1 is 10^{-9} per hour of operation, the following derived safety requirement shall apply to the Isolette controller:

REQ3	In *normal* mode, the controller shall activate an alarm whenever • the current temperature falls outside the *alarm* temperature range (either through temperature fluctuation or a change in the alarm range by an operator), or • a failure is signalled in any of the input devices (temperature sensor and operator settings).	The alarm temperature range is $m\_al \mathinner{..} m\_ah$. Monitored variable m_st shows "invalid" when any of the input signals fail.

REQ4	Once the alarm is activated, it becomes deactivated in one of two ways: • The nurse turns off the Isolette; • The alarm has lasted for 10 seconds, and after 10 seconds or more the alarm conditions are removed.	Refer to the relevant tables of monitored and/or controlled variables and function tables.

On the next page, you must add the next three most important R-Descriptions. Provide a brief rationale for each R-Description. Include any remaining R-Descriptions in an appendix to this document.

Additional three R-Descriptions with brief rational (one page)

8. E-descriptions

ENV1	The current temperature received from the sensor is a a real number in the range 68.0 to 105.0°F.	??

ENV5	The desired and alarm temperatures received from the operator are all in increments of 1°F.	Refer to the relevant tables of monitored and/or controlled variables and function tables.

Provide the next 3 most important E-descriptions on this page. The the rest in the appendix.

9. Abstract variables needed for the Function Table

		lo(i)
i=0		off
i>0	$m\_tm(i) < m\_al(i)$	on
	$m\_tm(i) \geq m\_al(i) \wedge m\_tm(i) < (m\_al(i) + EPS)$	lo(i-1)
	$m\_tm(i) \geq (m\_al(i) + EPS)$	off

Table 3: Abstract variable lo function table

		hi(i)
i=0		off
i>0	m_tm(i) >m_ah(i)	on
	$m\_tm(i) \leq m\_ah(i) \wedge m\_tm(i) \geq (m\_ah(i) - EPS)$	hi(i-1)
	$m\_tm(i) \leq (m\_ah(i) - EPS)$	off

Table 4: Abstract variable hi function table

	alarm(i)
$lo(i) = on \vee hi(i) = on$	on
$\neg(lo(i) = on \vee hi(i) = on)$	off

Table 5: Abstract variable alarm function table (combines lo and hi)

Where EPS = 0.5 (Timing resolution)

10. Function Tables

Starting on the next page, provide one function table for each control variable (in Table 2). Each control variable should have its own sub-section heading and its own page.

10.1. Function Table for heat control: c_hc

Function table goes here on this page. Other function tables each on their own page

<table>
<tr><td colspan="4"></td><td>c_hc(i)</td></tr>
<tr><td>i=0</td><td colspan="3"></td><td>off</td></tr>
<tr><td rowspan="5">i>0</td><td colspan="3">$c\_md(i) = init \vee c\_md(i) = off$</td><td>off</td></tr>
<tr><td rowspan="4">$\neg(c\_md(i) = init \vee c\_md(i) = off)$</td><td rowspan="3">c_md(i) = normal</td><td>$m\_tm(i) < m\_dl(i)$</td><td>on</td></tr>
<tr><td>$m\_tm(i) > m\_dh(i)$</td><td>off</td></tr>
<tr><td>m_dl(i) $\leq m\_tm(i) \leq m\_dh(i)$</td><td>c_hc(i-1)</td></tr>
<tr><td colspan="2">$\neg(c\_md(i) = normal(i))$</td><td>off</td></tr>
</table>

Table 6: Function table for heat control c_hc

10.2. Function Table for modes: c_md

<table>
<tr><td colspan="4"></td><td>c_md(i)</td></tr>
<tr><td>i=0</td><td colspan="3"></td><td>off</td></tr>
<tr><td rowspan="8">i>0</td><td>m_sw(i) = off</td><td colspan="2"></td><td>off</td></tr>
<tr><td rowspan="7">m_sw(i) = on</td><td colspan="2">c_md(i-1) = off</td><td>init</td></tr>
<tr><td rowspan="2">c_md(i-1) = init</td><td>$m\_st(i) = valid$
$\wedge\ m\_dl(i) \leq m\_tm(i) \leq m\_dh(i)$
$\wedge\ m\_al(i) < m\_dl(i) < m\_dh(i) < m\_ah(i)$</td><td>normal</td></tr>
<tr><td>$not(m\_st(i) = valid$
$\wedge\ m\_dl(i) \leq m\_tm(i) \leq m\_dh(i)$
$\wedge\ m\_al(i) < m\_dl(i) < m\_dh(i) < m\_ah(i))$</td><td>c_md(i-1)</td></tr>
<tr><td rowspan="2">c_md(i-1) = normal</td><td>m_st(i) = valid</td><td>c_md(i-1)</td></tr>
<tr><td>m_st(i) = invalid</td><td>failed</td></tr>
<tr><td rowspan="2">c_md(i-1) = failed</td><td>m_st(i) = invalid</td><td>c_md(i-1)</td></tr>
<tr><td>m_st(i) = valid</td><td>normal</td></tr>
</table>

Table 7: My caption

10.3. Function Table for temperature displayed: c_td

	c_td(i)
c_md(i) = normal	floor(m_tm + 0.5)
$c\_md(i) = off \vee c\_md(i) = init \vee c\_md(i) = failed$	0

Table 8: My caption

11. Validation

To be Done. Proof of completeness and disjointness and validation of the requirements using PVS.

Include the PVS sources in the appendix to this document but summarize the proofs here.

12. Use Cases

See Section A2 of [?] for some use cases. The use cases need to be adapted to the revised descriptions of the previous sections of this document. Provide one Use Case (a) informally and (b) formally in PVS.

13. Acceptance Tests

In this section, the use cases have to be converted into precise acceptance tests (using the function table to describe pre/post conditions) to be run when the design and implementation are complete. Describe one acceptance test

14. Traceability

Matrix to show which acceptance tests passed, and which R-descriptions they checked. No need to do this for this assignment.

15. Glossary

The definition of important terms is placed in this section. You are not required to complete this.

A. Appendix Title??

Appendix goes here. PVS sample below. Format so that there is no line wrapping

```
alert: THEORY
BEGIN
  delta: posreal = 0.5 % TR = 0.5 seconds
  IMPORTING Time[delta]

  p:      [DTIME -> real]  % Pressure
  alarm: [DTIME -> bool]

  hi: real
END
```