DESIGN AND IMPLEMENTATION OF A NEONATAL WARMING BLANKET WITH TEMPERATURE REGULATION FUNCTIONALITY

by

Morgan Warneford

A thesis submitted in partial fulfillment of the requirements for the degree of

Bachelor of Engineering (Biomedical) (Honours) / Master of Engineering (Biomedical)

Flinders University

2018

DECLARATION

I declare that this thesis does not incorporate, without acknowledgment, any material previously submitted as part of a degree or diploma in any university, and that, to the best of my knowledge, it does not contain any material previously published except where due reference is included in the text.

ABSTRACT

Core body temperature needs to be within a normothermic range, but premature neonates may have trouble regulating their own body temperature, as they have an underdeveloped autonomic nervous system. If the neonate is using energy to maintain a normal body temperature, this means that growth and development of the neonate is likely to be reduced. This is of crucial importance with premature neonates due to their inability to stay warm, and their smaller size (Okken, 1991).

Neonatal enclosed incubators, or open radiant warmers, are commonly used to ensure normothermic body temperature. Neonatal staff usually select a desired body temperature for the neonate and the heater output energy is automatically managed to maintain this desired body temperature. Enclosed incubators may also provide supplemental humidity and oxygen and help reduce external noise and light from the hospital environment (Bird, 2018). However, while neonatal incubators are ideal in terms of temperature regulation and environmental control, they prevent contact between parents and the neonate, and this can lead to adverse emotional and physical affects.

During extended hospital stays, neonates are rarely able to be cared for full-time by their family and this may lead to depression and anxiety in parents. Furthermore, neonates from the Neonatal Intensive Care Unit (NICU), once older, are also more likely to display "less adaptability, greater impulsiveness, more temper tantrums, shorter attention span, increased rigidity (and) decreased smiling," (Hunt, 2011).

The output of the thesis has the potential to reduce the occurrence of temperature-related neonatal problems and to increase the development of these neonates, both mentally and physically. Kangaroo care[1] may also become more common, and this would increase the wellbeing of the neonate and parents or guardians during this crucial development stage.

Kangaroo care provides many benefits including (Brusie, 2017):

[1] Kangaroo care is a term used to describe "a method of holding a premature baby that involves skin-to-skin contact" (*Kangaroo Care & Babies*, no date).

- Stabilised infant weight
- Lower rates of infection
- Lower mortality rates
- Higher breastfeeding rates
- Increased bonding
- Increased milk supply
- Decreased stress in the baby

Furthermore, the increased contact between neonates and parents has led to in-hospital benefits such as "shorter hospital stays for babies and increased staff satisfaction for doctors and nurses looking after the babies" (*Circle Of Care Optimising Outcomes for Newborns (COCOON)*, no date).

As an additional benefit, the incorporation of humidity monitoring in a new design could prove significant in terms of pressure sore prevention and maintenance. Literature supports the idea that humidity plays a significant role in pressure sore formation (Suriadi *et al.*, 2008), so real-time humidity data can ensure that any initial sores will be quickly identified, and rectified.

The heart rate of the neonate will also be monitored as heart rate is a significant indicator of wellbeing in neonates. As will be discussed in the thesis, heart rate can also be an indicator of the risk of infant mortality due to Sudden Infant Death Syndrome (SIDS).

The project will also lead to the production of a blanket which is more flexible and less labour intensive than those currently available in the market. Any time spared in the NICU is beneficial as the neonates are in such a fragile health state, and require continuous monitoring and care. The design considers the delicate and complex network of factors that need to be monitored and controlled to support the life of a premature or very low birth weight neonate. These factors include the temperature and humidity of the surrounding environment, as well as the heart rate and core body temperature of the neonate. The designed blanket also aims to incorporate the sensing and monitoring capabilities of neonatal incubators while maintaining the portability and flexibility of neonatal warming blankets.

The temperature output of the blanket is monitored and regulated using a compensation system. This sets the design apart from others in the market as the temperature of the embedded heating elements varies depending on the temperature of the neonate at a given time. Visual and audio alarm systems have also been put in place to ensure the safety of the neonate. The introduction of this design at the Flinders Medical Centre (FMC) is expected to decrease the temperature-related neonatal complications in the NICU.

To design and create the neonatal warming blanket, continuous dialogue was maintained between the student, FMC staff and the student's academic supervisor. A dedicated Printed Circuit Board (PCB) was designed and built which enabled all components to be mounted on one electronic board. Software was then written by the student which provided a means of interacting with the PCB and components, in order to obtain humidity, pulse and temperature data relating to the neonate and their surrounding environment.

Testing of each sensing component was carried out to determine the accuracy of each sensor relative to a known reference. However, testing on neonates in the NICU was outside the scope of this project. As a result of this thesis, a functional warming blanket prototype was constructed which successfully integrated humidity, pulse and temperature sensors. The accuracy of these sensors was verified using sensors of known accuracy, such as those used in the hospital environment.

It is important to note that the project objective was to design a prototype, and there are still areas that could be developed further in future work. While clinical testing was outside the scope of this project, this will be the next step in ensuring the safety of the neonate, parents and neonatal staff using the blanket in the future. In addition to this, increased budget in the future could lead to more comprehensive analysis into the sensors available in the market. This could, in turn, improve the overall accuracy of the sensors in the system.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would sincerely like to thank Associate Professor Kenneth Pope for his endless support throughout my university degree and his continued support during the development of this thesis. Your knowledge and passion in the field is inspiring, and I cannot thank you enough for the wisdom that you have imparted on all your students. I would like to extend the same gratitude to Mr. David Hobbs, who has also inspired me to become a successful and genuine biomedical engineer. Your enthusiasm for what you do is evident to all of your students, and I aspire to be like that in the future. Professor John Costi, you have also helped to shape me into a dedicated and committed engineer. You have been incredibly patient as I have developed my skills in academic research and I thank you for the massive role you have played in building my passion for the biomedical engineering field. Kenneth, David and John, you have all offered me endless opportunities to advance my career and education as an engineer and to support me professionally and personally, and I appreciate that more than you know.

I would also like to offer my sincere gratitude to Dr. Tony Carlisle, who assisted in initial phase of the conceptualization, and continually assisted in the revision of project deliverables. I would also like to thank Mr. Craig Peacock for his endless patience and kindness when assisting me with the planning and assembly of hardware throughout the project.

I would especially like to thank my parents, John and Allison Warneford, for their unbelievable support and kindness which is shown in everything that they do. Their interest and excitement over this thesis (and my degree as a whole) was no exception, and I am incredibly grateful to have such wonderful and inspirational role models in my life. You inspire me to become the best version of myself academically, professionally and personally and I am incredibly grateful for you both.

I would lastly like to thank my amazing partner, Cade Wittwer, for his support, patience and generosity over the past few years. He continues to support me in all my endeavors, and has been my strength throughout this process. He is also a very successful engineer, and has inspired me to continue to develop my skills and aim to be the best engineer, and best person, that I could possibly be.

Thank you all so much; I can honestly say that without your input and support, this thesis, and the completion of my degree, would not have been achievable.

ABBREVIATIONS

Abbreviation	*Definition*
A	Amperes
BFSL	Best Fit Straight Line
BPM	Beats Per Minute
bps	Bits Per Second
cm	Centimetres
ECG	Electrocardiogram
EEG	Electroencephalogram
EMG	Electromyogram
FMC	Flinders Medical Centre
GUI	Graphical User Interface
I2C	Inter-Integrated Circuit (Communication Protocol Port)
IDE	Integrated Development Environment
iOS	Internetwork Operating System

Abbreviation	*Definition*
LDR	Light Dependent Resistor
LED	Light Emitting Diode
MER	Medical Emergency Response
MOSFET	Metal-Oxide Semiconductor Field-Effect Transistor
NAN	Not a Number
NICU	Neonatal Intensive Care Unit
NTE	Neutral Thermal Environment
OLED	Organic Light Emitting Diode
OP-AMP	Operational Amplifier
PC	Personal Computer
PID	Proportional Integration Derivative
PRBS	Pseudo-Random Binary Sequence
PV	Process Variable

Abbreviation	*Definition*
QA	Quality Assurance
RMS	Root of the Mean Squared
RM	Registered Midwife
RN	Registered Nurse
RSS	Root of the Sum Squared
SCL	Serial Clock
SDA	Serial Data
SIDS	Sudden Infant Death Syndrome
SP	Set Point
SUDI	Sudden Unexpected Death of Infants
USB	Universal Serial Bus
VDD	Supply Voltage
WBS	Work Breakdown Structure

1 BACKGROUND

The term neonate is used to describe new-born children, typically less than a month old. Some of these neonates are born prematurely, or experience difficulties during the birthing process, and therefore require specialised medical care to help them become healthy.

Each year in Australia, there are around 800 neonatal deaths out of around 300,000 live births ('Perinatal deaths in Australia', 1993). This is a neonatal mortality rate of around 2.6 per cent. Furthermore, Sudden Unexpected Death of Infants (SUDI) is a condition leading to the deaths of around one infant in every three thousand births in 2016 (*Facts & Figures on Births, Perinatal Deaths; SUDI | Red Nose*, no date). While this is a rate 85% less than in 1989 (*Facts & Figures on Births, Perinatal Deaths; SUDI | Red Nose*, no date), these are still preventable deaths and, with the number of births equating to 311,104 in Australia in 2016, this is still 94 deaths caused by SUDI alone. Together with the over-represented neonatal deaths, the World Health Organization predicts that deaths caused by SUDI and neonatal complications can largely be prevented through the introduction of a unique warming blanket that incorporates 'intelligent' sensing (Hsia, 2013).

Such an 'intelligent' warming blanket would require a vital sign monitoring and regulation system. Temperature, humidity and pulse sensors would provide data to a microcontroller[2]. This microcontroller would interface with a computer, whereby algorithms would be in place to control the temperature output of heating elements, and to display sensor data to the user. While neonates requiring intensive care that can only be achieved by an incubator will not benefit from the device, those requiring less critical care can be more effectively monitored than current systems allow. This is the underlying concept of the proposed design, and will be discussed in subsequent sections of this thesis.

1.1 PROBLEM DEFINITION

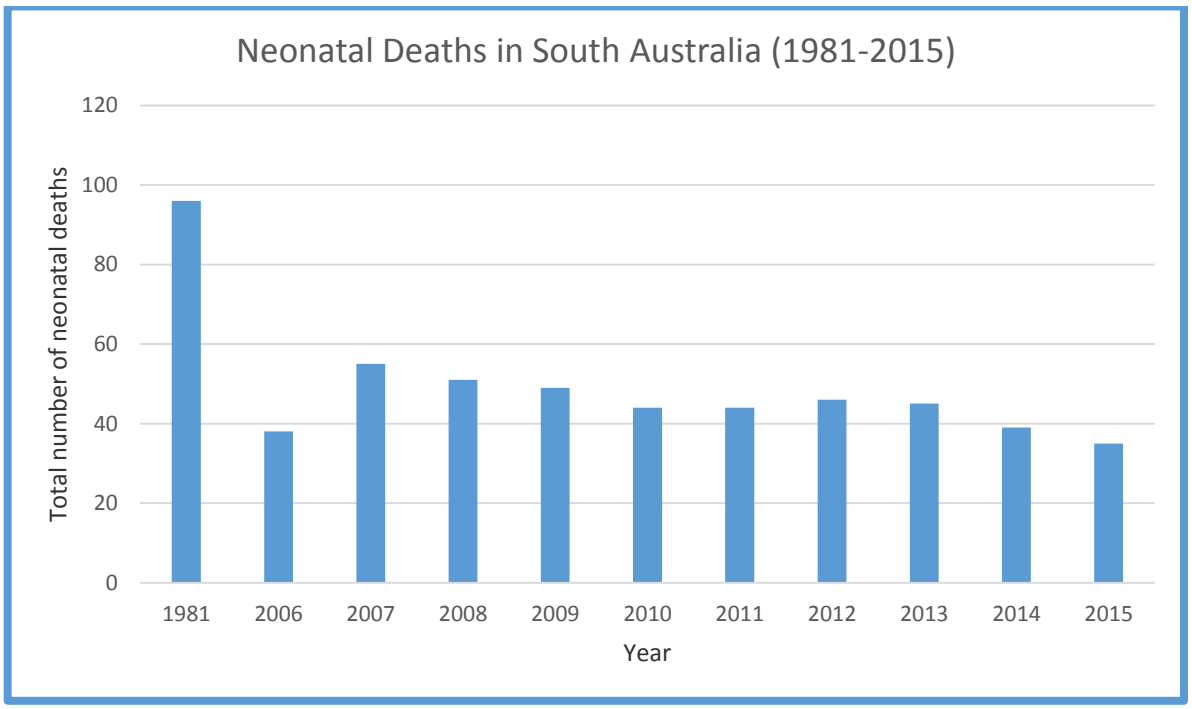
"To accelerate progress in child survival, focusing on the newborn is critical since the share of all under-five deaths occurring in the neonatal period (the first 28 days of life) is increasing" (Wardlaw *et al.*, 2014). A significant factor that contributes to this neonatal mortality rate is the array of

[2] "A microcontroller is a computer present in a single integrated circuit which is dedicated to perform one task and execute one specific application" (*What is a Microcontroller?*, no date).

difficulties that neonates face with temperature regulation. The body surface of a neonate is around three times greater than an adult when compared to the weight of the person (*Warmth and Temperature Regulation*, no date). Therefore, they are much more susceptible to heat loss from their skin to the environment.

Changes in temperature are responded to with a change in oxygen consumption. "If skin temperatures drop just one degree from the ideal 36.5 degrees Celsius, a baby's oxygen use can increase by ten per cent" (*Warmth and Temperature Regulation*, no date). For these reasons, temperature variations can have a dramatic effect on the oxygen requirements and overall wellbeing of premature and low birth weight neonates. Therefore, there needs to be a reliable and controllable temperature regulation system in place which will ensure that the temperature of the neonate will remain in the normothermic range[3] when they transition from neonatal incubators to less critical care.

It is important to note that advances in medical technology, such as the use of neonatal incubators, have ensured that the Australian fatality rate of premature neonates has dropped considerably. Figure 1-1 highlights this steady decrease in mortality from 1981 to 2015.



[3] Normothermia is "a body temperature within normal limits. The term is used mainly in contexts in which hypothermia is a possibility or a risk" (*Normothermia*, no date).

Figure 1-1: Neonatal mortality based on birth weight from 1981 to 2015 (Scheil et al., 2017).

Figure 1-1 shows a steady decrease in neonatal mortality since 1981. In 2015 specifically, 7.6% of all births occurred with neonates having low birth weights, meaning that there were 1,540 low birth weight births in total over the year (Scheil *et al.*, 2017). Referring to Figure 1-1 above, 35 of these births resulted in neonatal deaths, which is significantly less than the 96 neonatal deaths in 1981 (Scheil *et al.*, 2017). This decrease can be attributed to wider knowledge regarding the harmful side-effects of smoking and drinking alcohol during pregnancy, but also with technological advancements in neonatal care and thermoregulation.

It is important to consider that the cause of death can also have a significant impact on the number of neonates requiring care in the NICU. In Australia, around 32.3% of annual neonatal deaths are due to congenital abnormality, 33.5% are due to extreme prematurity, 7.6% are due to cardio-respiratory disorders, 5.8% are due to infection, 11.1% are due to neurological complications, 2.3% from gastrointestinal complications, with the remainder being of another classification, or not stated ('Perinatal deaths in Australia', 1993). The relative differences between these diagnoses are highlighted in Figure 1-2.

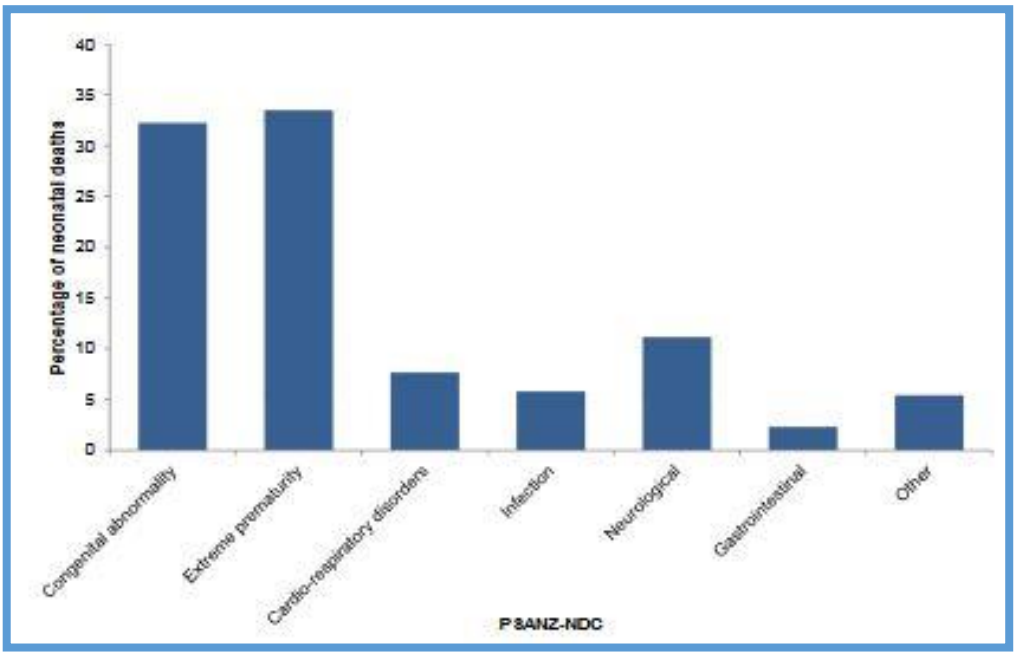


Figure 1-2: Causes of neonatal deaths, with extreme prematurity being significantly over-represented along with congenital abnormality ('Perinatal deaths in Australia', 1993).

Furthermore, the mortality of neonates based on gestational age can be seen in Figure 1-3.

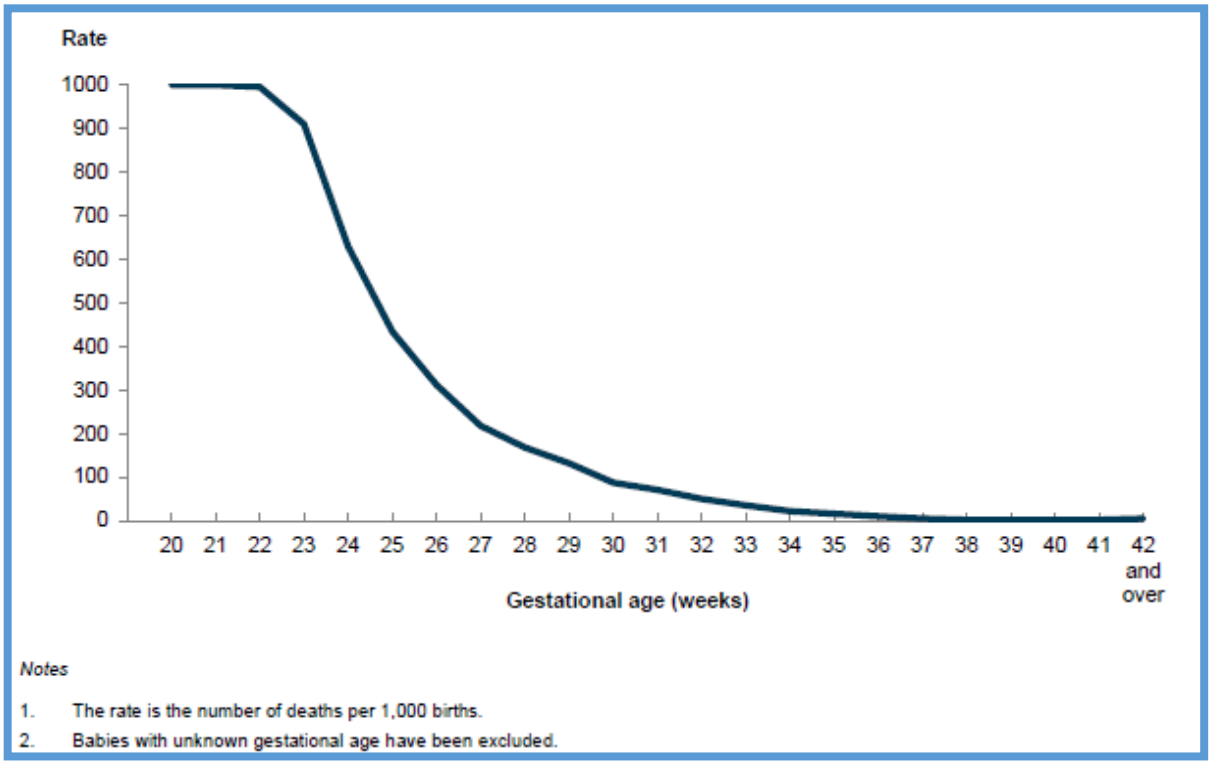


Figure 1-3: Neonatal mortality based on gestational age, from 2011 to 2015 (Scheil et al., 2017).

As can be seen in Figure 1-3, the gestational age of the neonate has a significant effect on neonatal mortality rate. Particularly for extremely premature births, where the gestational age is less than 29 weeks, the mortality rate is upwards of 200 deaths per 1,000 births. This mortality rate increases significantly as the gestational age of the neonate decreases, because the vital organs are predominantly underdeveloped.

Additional factors can also render a neonate more likely to experience difficulties after birth. For example, plurality[4] is a major contributing factor with regards to the neonatal mortality rate. This correlation is highlighted in Figure 1-4.

[4] Plurality is "an indicator of multiple birth, showing the total number of births resulting from a single pregnancy" (*Birth event—birth plurality*, no date).

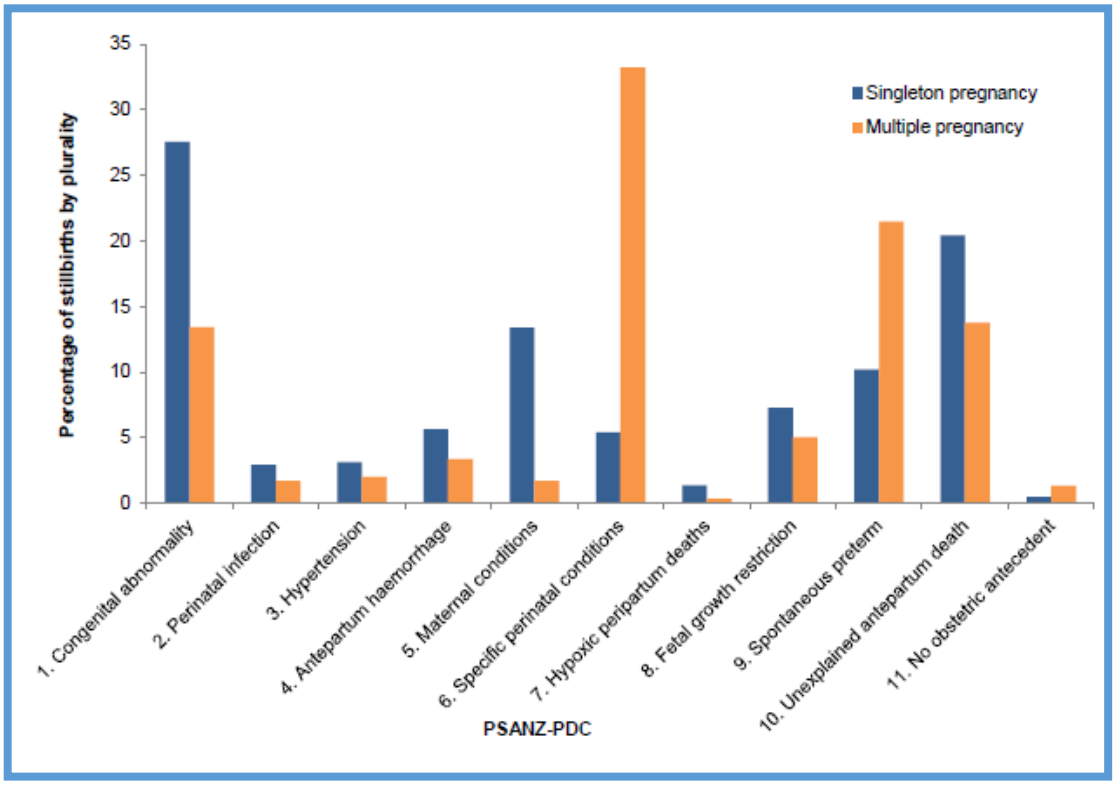


Figure 1-4: Leading causes of death in the neonatal sector, with a direct comparison between singleton and multiple pregnancies ('Perinatal deaths in Australia', 1993).

Figure 1-4 shows a stark difference between the mortality rate of singleton births when compared to the births of twins, triplets or other plural births. This information can lead to better care for these neonates as their care can be planned and anticipated before they are born, based on the information obtained during an ultrasound.

1.2 CURRENT CLIMATE

Prior to the commencement of this project, neonatal warming blankets were discussed with neonatal staff at the Flinders Medical Centre. One warming blanket has been trialed by NICU nurses with little success due to the inflexible nature of the heating element, and the labour intensive method of inserting the neonate into the blanket. It is far easier to wrap a warming blanket around a neonate, rather than trying to slide them into one. This is not only due to increased convenience for neonatal staff, but also minimal physical disruption to the neonate. However, this method has rarely been adopted in the market due to the difficulties associated with designing and utilizing flexible heating elements.

It is important to consider the array of neonatal warming blankets and devices currently used by other organisations in order to determine the key features which are shared between designs, and those which provide additional convenience and safety for the user and neonate respectively. Table 1-1 details some of the commercially available warming blankets and devices used in industry.

Thermal mattresses have also been included as they have been found to increase thermal stability, and maintain neonate temperature within the healthy range, when compared to neonates transported without thermal mattresses (L'Herault, Petroff and Jeffrey, 2001).

Item	Key Features
Medwarm Baby Heated Blanket – Swaddling Style	Reaches temperature of 32 to 39°C Accuracy of 0.1°C Heats to 37°C in 7-10 minutes Heats up to 4 hours Integrated alarm system Low input voltage 600 mm width 350 mm height
Mistral-Air Neonatal Plus	2 air inlets 1160 mm width 1020 mm height

Item	Key Features
Ready Heat Infant Warming Mattress	15-20 minute warm-up time Heats up to 8 hours Air activated

Item	Key Features
Ready Heat Infant Warming Cocoon	Maintains heat for 8 hours Attached hood Adjustable (suitable for neonates of varying sizes) Six warming panels
Neonatal Sleeping Bag Warmer	Portable Reusable Phase change material[5] pouch 440 mm width 290 mm height

[5] Phase Change Materials (PCMs) are "substances that absorb and release thermal energy during the process of melting and freezing" (*Frequently asked questions about phase change materials*, no date).

Table 1-1: Warming blankets and devices currently used in the neonatal care sector.

.

Despite the numerous key features listed in Table 1-1, it is important to note the severe lack of real-time temperature compensation algorithms in these designs. The warming blankets commonly used in industry reach a particular 'desired' value and remain at that temperature regardless of the core body temperature of the neonate. While this enables the customization of the 'desired' temperature based on the age and weight of the neonate, it does not allow for fast correction if the core body temperature of the neonate fluctuates. Therefore, another gap in the market is related to the ability of neonatal warming blankets to adjust temperature based on the real-time body temperature of the neonate.

Furthermore, it is evident that commercially available warming blankets are unlike incubators in terms of sensor capability, data logging and wireless interaction with a mobile phone or computer.

1.3 RESEARCH QUESTION

The aim of the following thesis will be to answer the research question, which can be summarized as follows:

Which neonatal warming blanket design will most effectively combine the sensing and monitoring functionality of an incubator with the flexibility and portability of a commercially available neonatal warming blanket?

1.4 EMBODIMENT DESIGN

Figure 1-5 summarizes the design concept in terms of the main components. Essentially, the warming functionality is enabled through the use of a battery-powered adjustable heating element. The heat produced by this heating element is transferred to the neonate, who is monitored for three main signs of health and wellbeing. The neonate's health is determined by monitoring their vital signs[6]. In order to achieve this, a pulse sensor, temperature sensors, and humidity sensors have been included in the design. The outputs from these sensors are fed directly into an Arduino microcontroller, which simultaneously outputs an alarm signal if sensor values are outside of the healthy range. All data obtained through the Arduino is represented visually on a PC.

[6]Vital signs are defined as being "clinical measurements, specifically pulse rate, temperature, respiration rate, and blood pressure that indicate the state of a patient's essential body functions" (*vital signs | Definition of vital signs in English by Oxford Dictionaries*, no date).

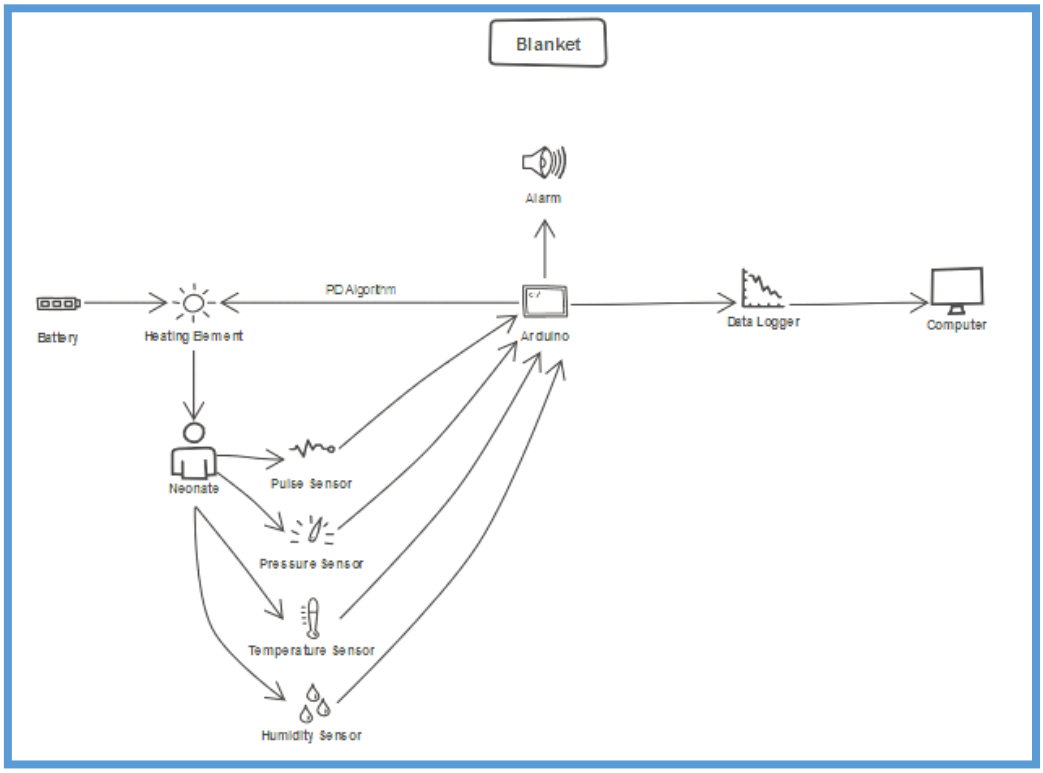


Figure 1-5: Generalized schematic of the neonatal warming blanket design. Note that the pressure sensor, battery and data logger were later excluded from this project due to scope creep.

1.4.1 WARMING BLANKET FUNCTIONALITY AND DESIGN

Warming blankets are essentially made up of heating elements which heat the surrounding material (known as the 'shell') (*How electric blanket is made - material, making, history, used, parts, components, steps, product, industry*, no date). The heating element wire is woven through the channels within the shell in a zigzag pattern as shown in Figure 1-6.

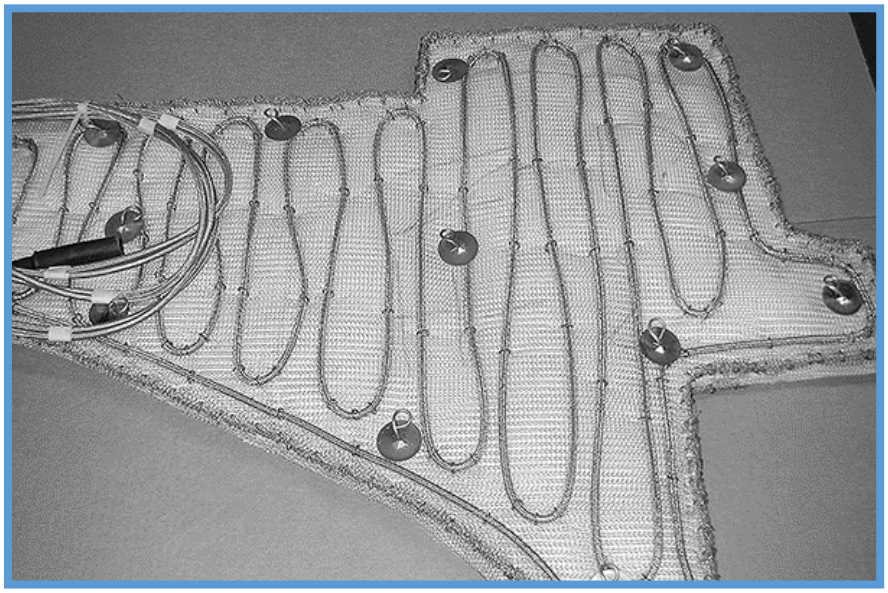


Figure 1-6: Typical zigzag construction of the heating element in a warming blanket (POWER Magazine :: Power generation news and jobs in coal, gas, nuclear, renewables, no date).

Heating elements are able to warm the material due to the conversion of an electrical current to heat. As a result of this energy conversion, the patient is warmed by heat conduction and convection from the blanket material to the patient's skin.

1.5 THESIS STRUCTURE

The following thesis will detail the research, design and development of a neonatal warming blanket for use in the NICU. As seen already, Chapter 1 has seen the introduction of the research topic, with Sections 1.1 and 1.2 providing a justification for its inclusion in the current market. This section has formed the basis of future chapters, such that there is a general understanding of the topic before the design and development of the warming blanket is discussed.

Chapter 2 of this document will outline the literature which has relevance in the neonatal field, as well as the connections between this literature and the future design of the warming blanket. Chapter 3 summarizes the management of this project while Chapter 0, on the other hand, details the design and development of the blanket. This section will form a large portion of the thesis, as various design options are considered and investigated in terms of efficacy.

Chapter 5 continues with an analysis and discussion, therefore summarizing the entirety of the document, and suggesting future improvements and work in the field of neonatal warming and temperature regulation. This section is also significant in terms of its ability to provide information for further development and improvement of the warming blanket for commercial applications.

2 LITERATURE REVIEW

2.1 SENSITIVITY TO TEMPERATURE

Heat loss/gain is a significant consideration when designing and using a warming device. The severity of heat fluctuations depend on a variety of factors including environmental temperature, age and body weight of the neonate. An environmental temperature range exists which is ideal for the neonate to grow and develop. This is called the Neutral Thermal Environment (NTE), and while the temperature is within this range, "oxygen and energy consumption is minimised" (*Clinical Guidelines (Nursing): Temperature Management*, no date). However, not all neonates are able to thermoregulate in the NTE range (T Carlisle 2018, personal communication, August 17) and, outside of this ideal temperature range, there can be critical complications.

Table 2-1 shows the NTE as a function of the age and body weight of the baby.

Age	1000-1200g (±0.5°C)	1201-1500g (±0.5°C)	1501-2500g (±1.0°C)
0-12 Hours	35.0	34.0	33.3
12-24 Hours	34.5	33.8	32.8
24-96 Hours	34.5	33.5	32.3
5-14 Days	33.5	33.5	32.1
2-3 Weeks	33.1	33.1	31.7
3-4 Weeks	32.6	32.6	31.4

Table 2-1: The ideal temperature within an incubator for a neonate less than one month old (NCCU Clinical Guidelines. Section 4: Thermoregulation, 2017).

It is vital to ensure that heat losses to the environment are minimised, or at least compensated for, in the temperature regulation system associated with the blanket. As shown in Table 2-1, the

required temperature of the environment is dependent on the neonate themselves (in terms of age and weight), so a real-time adaptive control system is required.

2.1.1 CAUSE OF TEMPERATURE VARIATIONS

Heat loss from the body of a neonate can occur through four main methods: conduction, convection, evaporation, and radiation. Conductive heat loss occurs when a neonate is placed naked on a cold surface such as a bed, table, or scale. The two surfaces (the skin of the neonate, and the surface of the bed, table or scale) are subject to heat transfer from higher temperature regions to lower temperature regions. This is directly affected by the surface area of either surface coming into contact with one another, and the temperature difference between the two surfaces ('Newborn Thermoregulation Self – Learning Module', 2013). Conductive heat loss is shown in the image labelled 'A' in Figure 2-1.

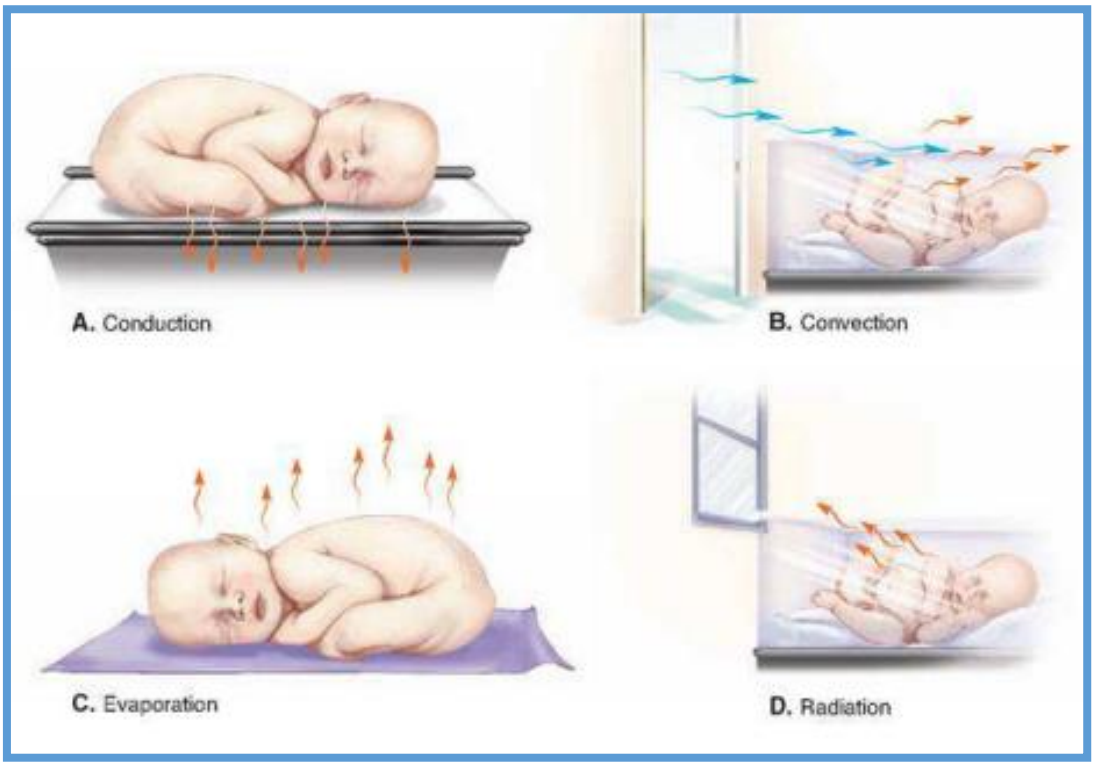


Figure 2-1: Examples of heat loss situations in the neonatal sector ('Newborn Thermoregulation Self – Learning Module', 2013). Note that in each of the images (A-D), the orange arrows represent heat loss, and the blue arrows (only in image B) represent the movement of cold air.

Convective heat loss, on the other hand, occurs due to the exposure of the neonate to surrounding colder, moving air. This might be caused by an open door or window, or a fan in the vicinity of the neonate. Unlike when heat is transferred to a *hard surface* in the case of conductive heat loss, convection results in the transfer of body heat to the *air*. The rate of heat loss due to convection is once again dependent on the exposed surface area of the neonate's body, as well as the air flow and

temperature gradient (i.e. difference in temperature between the air and the neonate's body temperature) ('Newborn Thermoregulation Self – Learning Module', 2013). This method of heat loss is depicted by the image labelled 'B' in Figure 2-1 above.

Evaporative heat loss involves evaporation of moisture from the skin during processes such as sweating, whereby heat transfers moisture into its gaseous form. This heat is therefore removed from the neonate, where the amount of heat lost is once again dependent on the surface area of exposed skin, the amount of moisture, as well as the velocity of the air in the room. This is the most common cause of heat loss in neonates at birth ('Newborn Thermoregulation Self – Learning Module', 2013), and is depicted in the image labelled 'C' in Figure 2-1 above.

Lastly, radiative heat loss occurs when the neonate is near (but not touching) colder surfaces. Heat transfer occurs between these two surfaces regardless of the lack of physical contact, and this heat transfer is dependent on the temperature gradient, the exposed surface area, and the distance between the two surfaces. While evaporative heat loss was the most common cause of heat loss *at birth,* radiative heat loss is the most common cause of heat loss in neonates *after birth* ('Newborn Thermoregulation Self – Learning Module', 2013). This type of heat loss is shown in the image labelled 'D' in Figure 2-1 above.

Neonatal warming blankets are able to reduce the heat loss occurring through these four methods as they provide protection against moving cold air, can help to reduce evaporation of fluids into the surrounding atmosphere, and provide thermal insulation against further heat loss.

2.1.2 CONSEQUENCES OF TEMPERATURE VARIATIONS

Outside of the ideal core body temperature range, there can be critical complications, as outlined in Table 2-2.

Core Temperature Range (°C)	Condition	Symptoms	Recommended Response
<28.0	Severe Hypothermia	Unconsciousness Dilated pupils Atypical pulse Difficulty breathing	Incubator with variable temperature as neonate warms up
28.0-32.0	Moderate Hypothermia	Drowsiness Difficulty breathing	Use isolette Heated mattress
32.0-36.0	Mild Hypothermia	Shivering Difficulty breathing Irritability Changes in mood Pale skin Difficulty feeding	Kangaroo care Hat on newborn Cover with blankets

Core Temperature Range (°C)	Condition	Symptoms	Recommended Response
36.5-37.5	Normal	N/A	N/A
>37.5	Mild Hyperthermia	Flushed skin Difficulty feeding Irritability	Move away from heat sources Undress baby as much as possible Breastfeed as often as possible Monitor temperature every 15 minutes until stable
>38.8	Severe Hyperthermia	Dehydration Irregular heart beat Weak cry Difficulty feeding Seizures Difficulty breathing	Warm bath around 2°C lower than the body temperature of the neonate Breastfeed where possible, or use intravenous fluids to rehydrate neonate Examine the neonate for signs of infection

Table 2-2: Temperature ranges that are used as a guide when determining health of neonate. Note that these temperatures vary depending on the body weight of the neonate (Takayama et al., 2000) ('Newborn Thermoregulation Self – Learning Module', 2013) (*Clinical Guidelines (Nursing) : Temperature Management*, no date).

2.1.3 THERMOREGULATION PRACTICE

Studies have shown that care provided by neonatal staff during the crucial period after birth, or from transition from the delivery room to the NICU, can lead to decreased body temperature (Knobel and Holditch-Davis, 2007). This is due to the fact that procedures such as intubation and suctioning typically require the removal of the neonate from a warm environment, and therefore expose the neonate to excessive temperature irregularity. Such studies have highlighted the importance of pre-warming the NICU, as well as the insertion of the neonate into a plastic bag from the neck down, in order to prevent as much heat loss as possible (Knobel and Holditch-Davis, 2007). The use of plastic bags for the purpose of heat retention is supported by numerous other studies (Leadford *et al.*, 2013) and continues to be used clinically as a means of preventing hypothermia in neonates.

2.1.3.1 MEASURING NEONATAL TEMPERATURE

Studies have shown that the site and method of temperature measurement have distinguishable effects on the temperature recorded (Placidi, Merusi and Gagliardi, 2014). The ideal temperature range for neonates, as measured at each of these locations, is shown in Table 2-3.

Site	Lower Limit (°C)	Upper Limit(°C)
Rectal	36.5	37.5
Skin	36.2	37.2
Axillary (Armpit)	36.5	37.3

Table 2-3: Upper and lower limits of neonatal temperature measurements according to three sites (rectal, skin and axillary) (Thermal Stability of the Premature Infant in NICU, no date).

The differing temperatures depicted in Table 2-3 for each of the different sites indicates that temperature sensors used to measure neonate temperature need to have built-in recognition of the sensor position. This is a significant consideration in the thesis as each site will require a different threshold at which the alarm stimulus is sounded or displayed.

Furthermore, the differing temperatures in Table 2-3 support the recommendation that neonatal body temperature be measured at two separate locations in order to obtain temperature

measurements as close to the true value as possible (*Thermal Stability of the Premature Infant in NICU*, no date). It is also important to note that constant rectal thermometry is not feasible, and this form of temperature measurement is considered high risk in the neonatal sector due to the invasive and often unreliable measurement (T Carlisle 2018, personal communication, August 17). While rectal temperature measurements may reflect the core body temperature of a neonate more accurately than axillary temperature, the risk is often deemed unreasonable for this fragile class of patients.

The most clinically accepted form of temperature measurement in neonates is non-invasive zero heat gradient (zero heat flux[7]) in the mid-thoracic region. This measurement occurs as the neonate lies on a thermally insulating mattress, meaning that minimal heat is transferred from the neonate's body to the mattress. A mid thoracic probe can be placed between the mattress and neonate's back and, after a short time, an approximation of the neonate's core body temperature can be obtained.

It is important to note that, while axillary temperature is less dangerous for neonates when compared to more invasive methods, "the influence of the environmental temperature and incorrect placement of the thermometer can lead to erroneous body temperature measurement" (Marui *et al.*, 2017). Even so, with regards to the thesis design, the neonates using the device are likely to be in temperature regulated hospital room. Furthermore, the neonates would be in an insulated blanket protected from any slight variations in environmental temperature or air flow. However, it is the temperature of the heating elements in the warming blanket that might interfere with the axillary temperature of the neonate, and so this remains an important consideration with regards to axillary temperature measurement.

[7] Heat flux is a term used to describe "the rate of heat energy transfer through a given surface" (*Heat flux - an overview*, no date).

2.1.4 HEART RATE VARIABILITY

Studies have shown that variability in heart rate can provide useful information regarding the balance between the sympathetic nervous system[8] (SNS) and the parasympathetic nervous system[9] (PNS) (Smith, Doig and Dudley, 2004). This has been proven as the high frequency regions of heart rate have been shown to correlate to PNS activity, with the low frequency regions correlating to SNS activity. Due to the decreased physical development of premature neonates, their SNS is typically further developed than the PNS (Smith, Doig and Dudley, 2004). As the PNS is typically dominant in terms of heart rate control, irregular heart rate is commonly experienced by premature neonates in the NICU (Rosen *et al.*, 2000).

In addition to this, there is evidence to support that Sudden Infant Death Syndrome (SIDS) is directly related to cardiovascular variability caused by the PNS and SNS (Pincus, Cummins and Haddad, 1993), resulting in the unforeseen death of a neonate with no prior indication of health complications ('Sudden Infant Death Syndrome (SIDS)', 2018). While the exact cause of SIDS remains unknown, the use of an effective heart rate monitoring system could potentially reduce the ocurrence of sudden infant deaths. Even so, other studies have disputed this and found no correlation between heart rate variability in infants suffering SIDS, and those that were not (Antila *et al.*, 1990). Further research is required in this area before the impact of heart rate monitoring systems on the number of infants suffering SIDS can be determined.

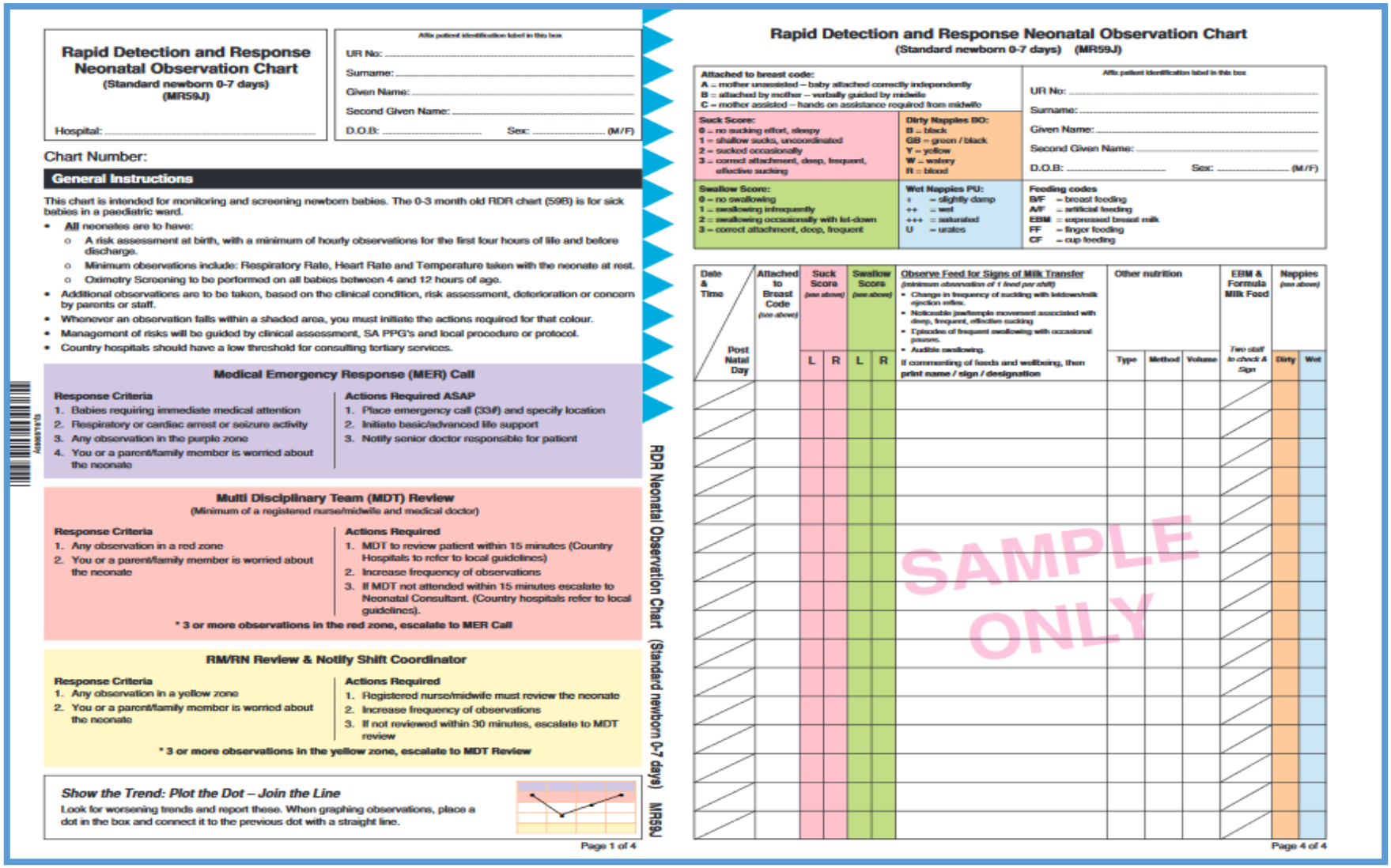
2.2 NEONATE MONITORING AND RESPONSE

Alarms and warnings of vital signs are important in the NICU in order to ensure that neonatal nurses are aware of any changes to the health of the neonates in the department. In South Australia, the most commonly used observation method is in the form of a chart, based on Rapid Detection and Response. This chart is shown in Figure 2-2 and Figure 2-3.

---

[8] The sympathetic nervous system is the "part of the nervous system that serves to accelerate the heart rate, constrict blood vessels, and raise blood pressure" (*Definition of Sympathetic nervous system*, no date).

[9] The parasympathetic nervous system "conserves energy as it slows the heart rate, increases intestinal and gland activity, and relaxes sphincter muscles in the gastrointestinal tract" (*Parasympathetic nervous system*, no date).



Rapid Detection and Response Neonatal Observation Chart
(Standard newborn 0-7 days)
(MR59J)

Hospital: .....

Affix patient identification label in this box

UR No: .....
Surname: .....
Given Name: .....
Second Given Name: .....
D.O.B: ..... Sex: ..... (M/F)

Chart Number:

General Instructions

This chart is intended for monitoring and screening newborn babies. The 0-3 month old RDR chart (59B) is for sick babies in a paediatric ward.

- **All** neonates are to have:
  - A risk assessment at birth, with a minimum of hourly observations for the first four hours of life and before discharge.
  - Minimum observations include: Respiratory Rate, Heart Rate and Temperature taken with the neonate at rest.
  - Oximetry Screening to be performed on all babies between 4 and 12 hours of age.
- Additional observations are to be taken, based on the clinical condition, risk assessment, deterioration or concern by parents or staff.
- Whenever an observation falls within a shaded area, you must initiate the actions required for that colour.
- Management of risks will be guided by clinical assessment, SA PPG's and local procedure or protocol.
- Country hospitals should have a low threshold for consulting tertiary services.

Medical Emergency Response (MER) Call

Response Criteria	Actions Required ASAP
1. Babies requiring immediate medical attention	1. Place emergency call (33#) and specify location
2. Respiratory or cardiac arrest or seizure activity	2. Initiate basic/advanced life support
3. Any observation in the purple zone	3. Notify senior doctor responsible for patient
4. You or a parent/family member is worried about the neonate	

Multi Disciplinary Team (MDT) Review
(Minimum of a registered nurse/midwife and medical doctor)

Response Criteria	Actions Required
1. Any observation in a red zone	1. MDT to review patient within 15 minutes (Country Hospitals to refer to local guidelines)
2. You or a parent/family member is worried about the neonate	2. Increase frequency of observations
	3. If MDT not attended within 15 minutes escalate to Neonatal Consultant. (Country hospitals refer to local guidelines).

*** 3 or more observations in the red zone, escalate to MER Call**

RM/RN Review & Notify Shift Coordinator

Response Criteria	Actions Required
1. Any observation in a yellow zone	1. Registered nurse/midwife must review the neonate
2. You or a parent/family member is worried about the neonate	2. Increase frequency of observations
	3. If not reviewed within 30 minutes, escalate to MDT review

*** 3 or more observations in the yellow zone, escalate to MDT Review**

Show the Trend: Plot the Dot – Join the Line

Look for worsening trends and report these. When graphing observations, place a dot in the box and connect it to the previous dot with a straight line.

RDR Neonatal Observation Chart (Standard newborn 0-7 days) MR59J

Page 1 of 4

Rapid Detection and Response Neonatal Observation Chart
(Standard newborn 0-7 days) (MR59J)

Attached to breast code:
A – mother unassisted – baby attached correctly independently
B – attached by mother – verbally guided by midwife
C – mother assisted – hands on assistance required from midwife

Suck Score:
0 – no sucking effort, sleepy
1 – shallow sucks, uncoordinated
2 – sucked occasionally
3 – correct attachment, deep, frequent, effective sucking

Dirty Nappies BO:
B – black
GB – green / black
Y – yellow
W – watery
R – blood

Swallow Score:
0 – no swallowing
1 – swallowing infrequently
2 – swallowing occasionally with let-down
3 – correct attachment, deep, frequent

Wet Nappies PU:
+ – slightly damp
++ – wet
+++ – saturated
U – urates

Affix patient identification label in this box

UR No: .....
Surname: .....
Given Name: .....
Second Given Name: .....
D.O.B: ..... Sex: ..... (M/F)

Feeding codes
B/F – breast feeding
A/F – artificial feeding
EBM – expressed breast milk
FF – finger feeding
CF – cup feeding

Date & Time / Post Natal Day	Attached to Breast Code (see above)	Suck Score (see above) L	Suck Score (see above) R	Swallow Score (see above) L	Swallow Score (see above) R	Observe Feed for Signs of Milk Transfer (minimum observation of 1 feed per shift) • Change in frequency of suckling with letdown/milk ejection reflex. • Noticeable jaw/temple movement associated with deep, frequent, effective sucking • Episodes of frequent swallowing with occasional pauses. • Audible swallowing. If commenting of feeds and wellbeing, then print name / sign / designation	Other nutrition Type	Other nutrition Method	Other nutrition Volume	EBM & Formula Milk Feed Two staff to check & Sign	Nappies (see above) Dirty	Nappies (see above) Wet

SAMPLE ONLY

Page 4 of 4

Figure 2-2: First pages of a neonatal nurse observation chart commonly used in clinical situations.

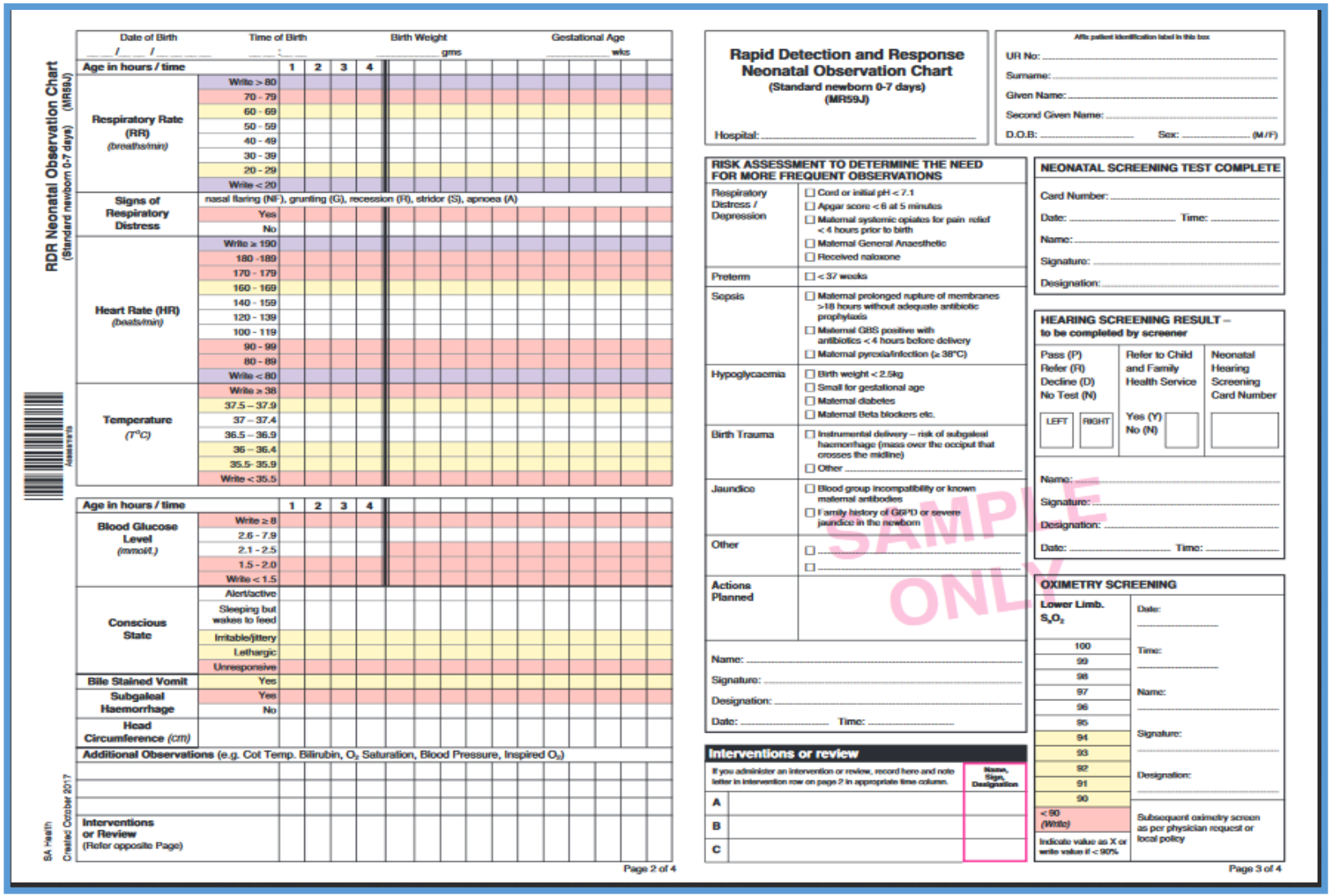


Figure 2-3: Remaining pages of a neonatal nurse observation chart commonly used in clinical situations.

As can be seen from Figure 2-2 and Figure 2-3, the axillary temperature of neonates must be monitored alongside other variables such as respiratory rate, heart rate and blood glucose level. While the thesis involves the design of a neonatal warming blanket responsible for measuring temperature, humidity and heart rate, the user interface could be developed to integrate data from other sensors, such as those measuring respiratory rate and blood glucose level. This would result in a more comprehensive system that could potentially reduce the complexity of neonatal systems in the Flinders Medical Centre.

Each section of the chart shown in Figure 2-2 and Figure 2-3 is colour coded to indicate the necessary course of action. In terms of the axillary temperature of the neonate, 36.9 to 37°C is not coloured, meaning that this range is safe and no action is required to change the body temperature of the neonate. If the axillary temperature is either between 37.5 and 37.9°C or between 35.9 and 36°C, the temperature falls within the yellow group, meaning that a Registered Midwife (RM) or Registered Nurse (RN) must review and the shift coordinator must be notified. The frequency of temperature observations must also be increased from this point onwards to ensure that the temperature does not fall or rise further from the expected body temperature of the neonate. Lastly, if the axillary temperature of the neonate is less than 35.5°C or greater than 38°C, this falls in the red category, meaning that a multi-disciplinary team (MDT) review is required. This review is required within 15 minutes of the initial observation, and the frequency of future observations must be increased. If the review does not occur within the allocated 15 minutes, or if the temperature of the neonate falls in the red category three or more times, it is escalated to a medical emergency response (MER) call, and life support is initiated.

In future revisions, the design detailed in this thesis will complement this system, as the temperature data will be displayed, logged and categorised without the need for nurses to manually take the neonate's temperature and write details on the chart. This will also increase the frequency of observations as they can occur automatically throughout the neonate's stay in the NICU, and only neonates falling in the yellow or red categories will be flagged for the neonatal staff. In addition to this, transcription errors will be greatly reduced in the NICU.

2.2.1 ALARM SYSTEMS

Literature has investigated ideal alarm settings in a hospital environment (Block, Nuutinen and Ballast, 1999), as well as speculating that "the design and implementation of alarms has not always taken the cognitive capacity and processing mechanisms of the user into account" (Edworthy and Hellier, 2006).

One study by Bitan et al. (2004) investigated the reaction of neonatal nurses to the sounds of alarms in the NICU. The study found that "nurses often do not respond directly to alarms, but, rather, use them as additional sources of information in their ongoing flow of actions" (Bitan *et al.*, 2004). The factors which were found to affect the likelihood of a nurse reacting to an alarm were the cause of the alarm, the duration and the characteristics of the neonate (Bitan *et al.*, 2004). This is an important consideration in the thesis because there may be other factors which will need to be incorporated into a warning system for the temperature controlling blanket (such as physical and audio indications of neonatal wellbeing).

Upon reviewing the literature, it was also realized that alarm fatigue is a significant clinical issue. This fatigue is essentially the result of desensitization to the alarm stimulus, and can result from "a high false alarm rate, poor positive predictive value[10], lack of alarm standardization, and the number of alarming medical devices in hospitals today" (Cvach, 2012). Konkani et al. (2012) have also raised the idea that a reduction in alarm tones can increase "patient safety" and promote healing (Konkani, Oakley and Bauld, 2012).

In order to reduce the number of alarms in the hospital which are not indicating crucial information, the alarm system used by the designed neonatal warming blanket will use audio alarms only when crucial information must be immediately conveyed to the staff or parents. Any other information will be displayed visually in order to reduce the desensitization of neonatal staff to alarms in the NICU, and potentially reduce the time taken to respond to critical situations.

2.2.2 GRAPHICAL USER INTERFACES FOR TEMPERATURE MONITORING

An important part of any temperature monitoring and control device is the feedback that is provided for the user or, in this case, the neonatal staff and parents in the NICU. This can be in the form of visual and/or audio feedback, and it is important to consider pre-existing user interface designs in order to improve on these where possible.

There are aspects of user interface design which remain consistent in literature. These include input controls, which might be in the form of drop down lists, buttons or text fields. Navigational

[10] Positive predictive value describes the response of neonatal staff to an alarm when action is required (when the neonate does require actual assistance. Therefore, *poor* positive predictive value describes the response of neonatal staff to alarms when the neonate does not require assistance.

components are also important in a medical GUI, as are informational components such as notifications or message boxes. Of course, the number of these elements included in the design is dependent on the complexity and function required of the user interface (Kaur and Mathew, 2016; Qin, Bek and Utley, 2000; Zosimadis, 2000). These elements make up a usable user interface, and should be considered in the thesis. A sample of the user interfaces used in pre-existing technology have been outlined in Table 2-4.

Image	Features	Potential Improvements
 Figure **2-4**: User interface used to display both **graphical and numerical data (Q3Logics, no date)**.	Clear numerical temperature values Waveform showing temperature history	Increase label size Include upper and lower alarm limits on graph Change thickness of waveform for greater contrast between the waveform and the background
 Figure **2-5**: User interface displaying humidity, temperature and pressure **(Clean Room Pharma HVAC, no date)**.	Clear numerical temperature values Colour coded data Time and date information	Increase label size

Image	Features	Potential Improvements
 Figure **2-6**: Medical application displaying blood pressure, heart rate, temperature and calories b**urned (Medical App Design by Ramotion, no date)**.	Numerical data values Aesthetically pleasing	Increase label size Increase contrast between background and data (such as black text on white background)

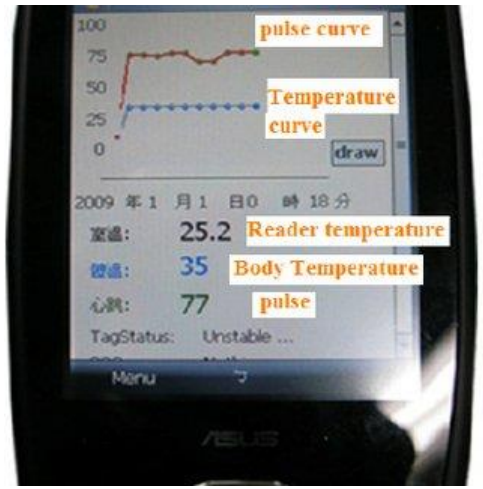
Image	Features	Potential Improvements
 Figure **2-7**: User interface presenting pulse and temperature values **(Wu et al., 2011)**.	**Numerical data values** **Single graph depicting pulse and temperature data** **Date information** **Colour coded graph correlating to numerical values**	**Units and labels on the graph** **Include units for the numerical data**

Table 2-4: Examples of user interfaces used to monitor health data.

2.3 RELEVANT STANDARDS

The following portion of the thesis will detail the standards that were considered during the design of the warming blanket.

2.3.1 ALARM SYSTEM REQUIREMENTS (AS/NZS 3200.1.8:2005)

This document outlines the standards associated with the "general requirements, tests and guidance for alarm systems in medical electrical equipment and medical electrical systems" ('Medical electrical equipment Part 1.8: General requirements for safety—Collateral Standard: General requirements, tests and guidance for alarm systems in medical electrical equipment and medical electrical systems', 2005). This is of vital importance with regards to the thesis, as alarm systems are commonly integrated into the design of neonatal thermoregulation devices to determine when there has been a change that may be detrimental to the health of the neonate, and to alert neonatal staff and parents.

Specifications are required for any visual alarm indicators (such as flashing lights). These are outlined in Table 2-5.

Alarm Category	Indicator Colour	Flashing Frequency	Duty Cycle
High Priority	Red	1.4Hz to 2.8 Hz	20% to 60% ON
Medium Priority	Yellow	0.4Hz to 0.8Hz	20% to 60% ON
Low Priority	Cyan or Yellow	Constant (ON)	100% ON

Table 2-5: Requirements of a visual warning system ('Medical electrical equipment Part 1.8: General requirements for safety—Collateral Standard: General requirements, tests and guidance for alarm systems in medical electrical equipment and medical electrical systems', 2005).

It is a requirement in the medical field that there is at least one visual warning indicator, and this must be visible and distinguishable at a distance 1 metre away from the medical equipment. Similarly, an audio signal is also required, with a pulse frequency[11] between 150Hz, and 1000Hz.

When the alarm is signalling a high priority situation (emergency), the pulse duration[12] must be within 75 and 200 milliseconds. Alternatively, when the alarm is signalling a medium or low priority, the pulse duration must be between 125 and 250 milliseconds.

2.3.2 HEATING DEVICE REQUIREMENTS (IEC 80601-2-35:2009)

This document describes the safety requirements associated with "blankets, pads and mattresses intended for heating in medical use" (*IEC 80601-2-35:2009(en) Medical electrical equipment — Part 2-35: Particular requirements for the basic safety and essential performance of heating devices using blankets, pads or mattresses and intended for heating in medical use*, 2009). It begins by outlining the mechanical strength requirements of such a material. The mattress, pad or blanket is required to undergo strength and rigidity testing to the standards outlined in Sections 21.101 and 21.102 of the document.

It is also required that the device only heat when all components are correctly installed by the operator or person performing regular maintenance. This will prevent overheating and potential fire risks. Furthermore, the "range of the temperature control setting shall be at least from 35°C to 38°C, but not exceeding 41°C" (*IEC 80601-2-35:2009(en) Medical electrical equipment — Part 2-35: Particular requirements for the basic safety and essential performance of heating devices using blankets, pads or mattresses and intended for heating in medical use*, 2009). The tests that must be carried out to ensure the fulfilment of these requirements are outlined in each respective section of the document.

2.3.3 QUALITY REQUIREMENTS (AS ISO 13485:2003)

This document refers to the quality management of medical systems. Essentially, it highlights the requirement of adequate documentation relating to the medical system including a quality manual, documented objectives and policies, documented procedures, and documented control processes. Furthermore, the document outlines the responsibility of management to ensure that the medical

[11] The pulse frequency is defined as the number of pulses per second.

[12] Pulse duration is defined as the amount of time taken for one full pulse to occur.

system remains of a high quality, and that the "importance of meeting customer as well as statutory and regulatory requirements" (*IS/ISO 13485: Medical Devices-Quality Management Systems-Requirements for Regulatory Purposes*, 2003) is not overlooked. Additional factors that can support or detract from quality assurance are also discussed (namely training, infrastructure and the work environment).

2.3.4 ELECTRICAL SAFETY REQUIREMENTS (AS/NZS 3200.1.1:1995)

This standard refers to specifications for the approval and testing of medical electrical equipment. Specifically, the document outlines the safety requirements associated with this equipment in an Australian hospital setting. In terms of the design phase of the thesis, this document is arguably irrelevant. However, were the designed neonatal warming blanket to be used in a clinical setting, this document outlines the requirements that would need to be considered.

For example, all medical electrical equipment requires ('Australian/New Zealand Standard Approval and test specification— Medical electrical equipment Part 1.1: General requirements for safety—Collateral Standard: Safety requirements for medical electrical systems [IEC title: Medical electrical equipment Part 1', 1995):

- "Instructions for cleaning, where applicable, sterilizing and disinfecting each item of equipment forming part of the system"
- "Additional safety measures which should be applied, during installation of the system"
- "Which parts of the system are suitable for use within the patient environment"
- "Additional measures which should be applied during preventative maintenance"
- "A warning that multiple portable socket-outlets shall not be placed on the floor"
- "The maximum permitted load for any multiple portable socket-outlet(s)"
- "An instruction that multiple portable socket-outlets provided with the system shall only be used for powering equipment which forms part of the system"
- "An explanation of the risks of connecting a non-medical electrical equipment, which has been supplied as part of the system, directly to the wall outlet when the system is supplied via a multiple portable socket-outlet with a separating transformer"

- "An explanation of the risks of connecting electrical equipment which has not been supplied as part of the system, to the multiple portable socket-outlet"

The remainder of the document outlines the various hazards associated with electrical equipment (such as fire hazards and leakage currents) as well as how these must be prevented and mitigated.

2.3.5 PROGRAMMABLE ELECTRICAL MEDICAL SYSTEM STANDARDS (AS/NZS 3200.1.4:1997)

This document outlines the safety requirements for "programmable electrical medical systems" ('Approval and test specification— Medical electrical equipment Part 1.4: General requirements for safety—Collateral Standard: Programmable electrical medical systems [ISO title: Medical electrical equipment, Part 1: General requirements for safety — 4. Co', 1997). A detailed summary of an appropriate risk assessment and risk management system is outlined.

This document also mentions that hazards can be to patients, operators, service personnel, bystanders, and the environment. They may be caused by human factors, hardware issues, software issues, integration errors or environmental conditions ('Approval and test specification— Medical electrical equipment Part 1.4: General requirements for safety—Collateral Standard: Programmable electrical medical systems [ISO title: Medical electrical equipment, Part 1: General requirements for safety — 4. Co', 1997). The risk is identified in terms of the probability of it occurring, and the severity of its implications. A risk is also categorised into one of three groups (intolerable, as low as reasonably practicable (ALARP) or broadly acceptable). These categories are shown in Figure 2-8.

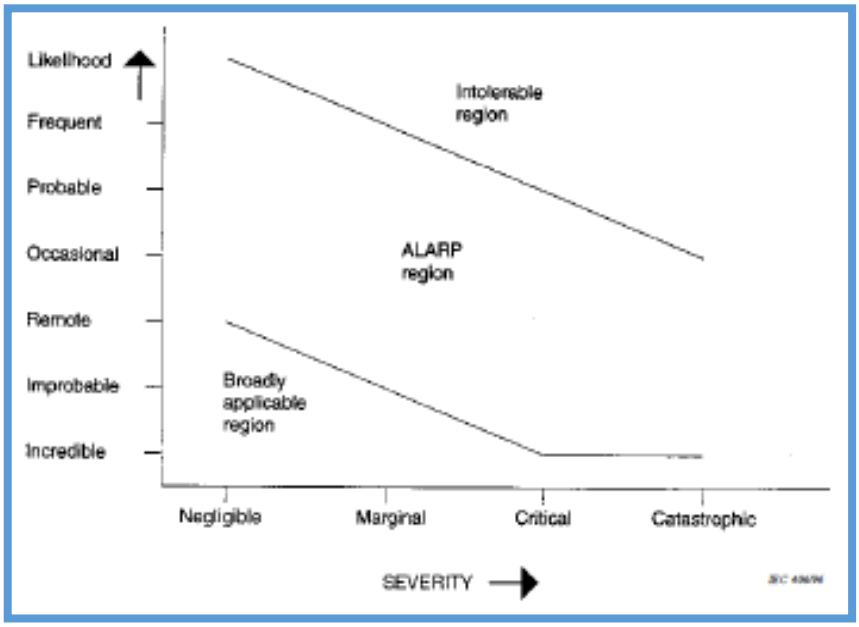


Figure 2-8: Tool used to determine the category of a particular risk, based on severity and likelihood ('Approval and test specification— Medical electrical equipment Part 1.4: General requirements for safety—Collateral Standard: Programmable electrical medical systems [ISO title: Medical electrical equipment, Part 1: General requirements for safety — 4. Co', 1997).

As can be seen from Figure 2-8, risks become intolerable when a catastrophic event is extremely likely to occur. In the case of medical equipment specifically used with neonates, the tolerance of risks becomes arguably smaller, due to the vulnerability of this target market.

2.4 SIMILAR STUDIES

Upon review of the literature, there are few neonatal warming blankets available in the current market which adequately cater to the complex and sensitive needs of premature or ill neonates. One outstanding study resulted in the production of the neonatal sleeping bag design. Kongsayreepong, et al. (2002) carried out a study in the hopes of designing "a reusable, custom-made warming blanket" that "prevents core hypothermia during major neonatal surgery" (Kongsayreepong *et al.*, 2002). A major point of investigation in this study was to determine whether this reusable neonatal warming blanket proved to have a greater efficacy than the disposable blankets that were on the market at the time of publication. To determine which option produced the most desired warming result, the heat transfer between the blanket and the air was analysed. While the study verified that the reusable neonatal warming blanket did prove beneficial when preventing core hypothermia in neonates, it did not prove to have significantly different

results than the disposable blankets that were used as a point of comparison (Bair Hugger 530[13] and Bair Hugger 555[14]).

Another study by Hubert et al. (2010) incorporated a hybrid design. This meant that, instead of investigating the efficacy of a neonatal warming blanket or incubator design independently, this study analysed a combination of the two. A key aspect of this methodology therefore meant that the efficacy of two incubator warming devices were compared with a neonatal warming blanket, with either of the incubators first being tested with a neonatal warming blanket, then without. Interestingly, it was found that one incubator resulted in a higher efficacy when used in conjunction with a neonatal warming blanket while the other incubator design did not result in better temperature regulation when used in conjunction with a neonatal warming blanket (Hubert *et al.*, 2010). This result was determined through the measurement of a time constant, variation in body temperature, and changes in air temperature. Overall, while the results of this study are arguably inconclusive, this proves that there is room for further investigation as the combination of a neonatal warming blanket with an incubator could potentially lead to more responsive control of the environment. A point of interest in future studies will be to determine the efficacy of the neonatal warming blanket designed in this thesis when used in conjunction with a neonatal incubator.

A significant gap in either of the above mentioned studies was evident in terms the thermoregulation capabilities of the neonatal warming blanket. Horey, Alvite and Kohn (2004) developed a warming blanket that consisted of a heating element, and sensors to detect both ambient air temperature and the desired temperature of the patient. A control system was used in order to adjust the output of the heating element in order to compensate for any difference between the ambient air temperature, and the desired temperature (Horey, Alvite and Kohn, 2002). The purpose of this design was to cater for changes in air temperature that occur in a home environment. For example, a person may set a household electric blanket to a particular

---

[13] The Bair Hugger 530 is a neonatal warming blanket that has been developed for emergency situations where there is risk of bodily harm by decreased body temperature (*3M BAIR HUGGER Model 530 Pediatric Warming Blanket*, no date). For example, this blanket could be used while the neonate or child is transported from an emergency site to a hospital.

[14] The Bair Hugger 550 is similar to the Bair Hugger 530 in the way that it too is designed for immediate but temporary warming of the neonate or child (*3M™ Bair Hugger™ Pediatric Underbody Blanket, Model 555 | 3M United States*, no date).

temperature that warms them early in the night but, as the air temperature drops, the heat output of the warming blanket may become insufficient. A design such as this could be adapted to measure the desired and actual core temperature of a neonate in the NICU.

Similar patents have been approved which detail temperature compensation processes for warming blankets. Sullivan et al. (1992) invented a flexible heating pad capable of self-regulation. In order to regulate temperature, a "solid state[15] timed interval control circuit" (Sullivan *et al.*, 1992) was used in conjunction with a positive temperature coefficient (PTC)[16] material. The most important characteristic of this material is that its resistance varies with temperature – a characteristic crucial to the success of this thermoregulation system. This is an important consideration for the thesis as material selection forms a part of product design. Another point of similarity between this design and that proposed in the thesis is the heating element flexibility. As discussed in the patent, conventional heating pads can be damaged if "the pad is flexed, or rolled up" as "it may damage the structural integrity of the heating element" (Sullivan *et al.*, 1992).

The patent most similar to the thesis details a design by Mills et al. (2012) which regulates temperature based on the body temperature of small animals and patients. While this reflects the same purpose as that of the design proposed in the thesis, this design requires manual calibration during operation. In other words, the operator must adjust the temperature themselves, and the system responds by altering the voltage supplied to the heating element (Mills *et al.*, 2012). The design proposed in the thesis will be automatically controlled so that there is no requirement for operator adjustments.

Grosholz and Andreasen (1966) investigated the use of automatic temperature regulation algorithms in incubator design. However, due to the nature of enclosed neonatal incubators, temperature regulation of the neonate is achieved by altering the temperature of the air within the incubator, rather than the skin temperature of the neonate directly using heating elements (Grosholz and Andreasen, 1966).

---

[15] Solid state is a term used to describe a specific loading method. "The invention of the transistor allowed devices to be controlled without any moving parts, thus the term 'solid state'" ('Solid State Timers and Controllers', no date). In this way, solid state systems utilize the properties inherent with solid semiconductors, rather than using moving parts.

[16] Positive temperature coefficient materials have a resistance that increases as the temperature increases (*Temperature Coefficient of Resistance*, no date).

Lauck (1967) also designed a heating element temperature control system, whereby a pre-determined temperature was set as the 'desired' temperature. This is similar to the design proposed in the thesis whereby the heating elements are turned on or off in order to achieve the desired temperature that was set by the operator. However, one point of difference is that this design uses a thyristor, which "provides heating element use of both positive and negative halves of a conventional alternating current power supply" (Lauck, 1967). In other words, the thyristor enables more efficient heating. This could be a future consideration for the design outlined in this thesis.

2.5 GAPS IN CURRENT KNOWLEDGE AND DESIGN

Upon discussion with Dr. Tony Carlisle (2018) in the Flinders Medical Centre NICU, the severe lack of versatile neonatal warming blankets in the current market was evident. A Parker Healthcare neonatal warming blanket identified by staff at the Flinders Medical Centre was trialled with poor results. The blanket was largely inflexible due to the material properties of the heating element selected, and the nurse was required to slide the neonate into the blanket, which proved to be labour-intensive in a fast-paced and time-sensitive environment.

When reviewing the literature, the lack of flexibility of warming blanket heating elements was confirmed, and there was a distinct lack of research into the use of automated control systems for neonatal thermoregulation. Therefore, the value of this project lies in the fact that it will incorporate these areas for development in the current market, in order to produce a neonatal warming blanket that is approved, and potentially adopted, by neonatal staff.

3 PROJECT MANAGEMENT

3.1 WORK BREAKDOWN STRUCTURE

The Work Breakdown Structure (WBS) outlined in Figure 3-1 has served as a project management tool which ensured the effective breakdown of project tasks. This was of crucial importance in the thesis as many components were acquired, tested and assembled, and so it was necessary to plan each step carefully to ensure there was little room for error.

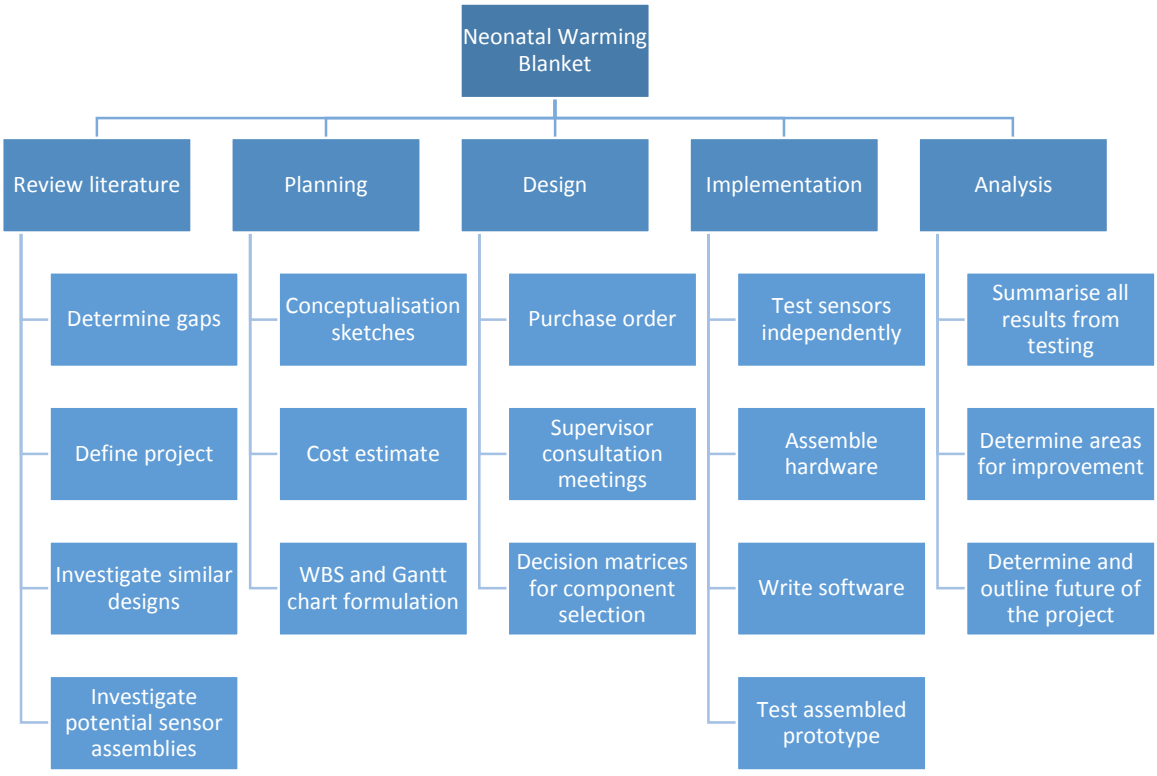


Figure 3-1: Work Breakdown Structure (WBS) for neonatal warming blanket design and implementation.

3.2 PROJECT PROCESS

The project process was straightforward, where the initial planning and design led to the ordering of components and testing of those components to ensure suitability for the application. From there, the

components were assembled, and the prototype was tested in its entirety. This process is visually depicted in Figure 3-2.

Figure 3-2: Brief overview of the process to be undertaken during the project.

In terms of dependencies, Figure 3-2 accurately depicts the need for each step to be completed before the next can begin. For example, the design was required to reach completion before the purchase list could be submitted, the purchase must have been completed before the testing of individual components could be carried out, and so on. In this way, Figure 3-2 visually represents the tasks included in the Critical Path[17], such that these tasks depicted those with no leniency.

3.3 PROJECT DELIVERABLES

The deliverables of the project were also identified prior to the commencement of the design phase. These were identified as those listed in Table 3-1.

Deliverable	Estimated Completion Date	Actual Completion Date
Proposal Seminar	April 10, 2018	April 10, 2018
Literature Review	July 23, 2018	July 23, 2018

[17] A Critical Path is the "longest sequence of activities in a project plan which must be completed on time for the project to complete on due date" (*What is a critical path?*, no date)

Deliverable	Estimated Completion Date	Actual Completion Date
Test Results for Independent Sensors	July 8, 2018	June 17, 2018
PCB	July 23, 2018	July 5, 2018
Completed Prototype with PID Algorithm Implemented	October 14, 2018	October 14, 2018
Results Seminar Presentation	September 19, 2018	September 18, 2018
Expo Poster	October 17, 2018	N/A
Expo Presentation	October 31, 2018	N/A
Thesis	October 15, 2018	October 15, 2018

Table 3-1: Summary of key project deliverables and predicted completion dates. Note that the items highlighted in green are assessment due dates set by the University, and are therefore immovable. Items with an actual completion data shown as N/A are due after the submission of this thesis.

The identification of these deliverables, and the adherence to planned completion dates was crucial to the success of the project. As this project consisted of numerous test regimes and design phases, it was necessary to set these estimated completion dates as a method of ensuring project completion.

3.4 OBJECTIVE TREE

This portion of the thesis depicts the objective tree that was created in order to further refine the features of the blanket, and determine which of these were most important, and which features were desired but not required for the product to be successful. Figure 3-3 shows this objective tree, alongside the weighting factor for each element.

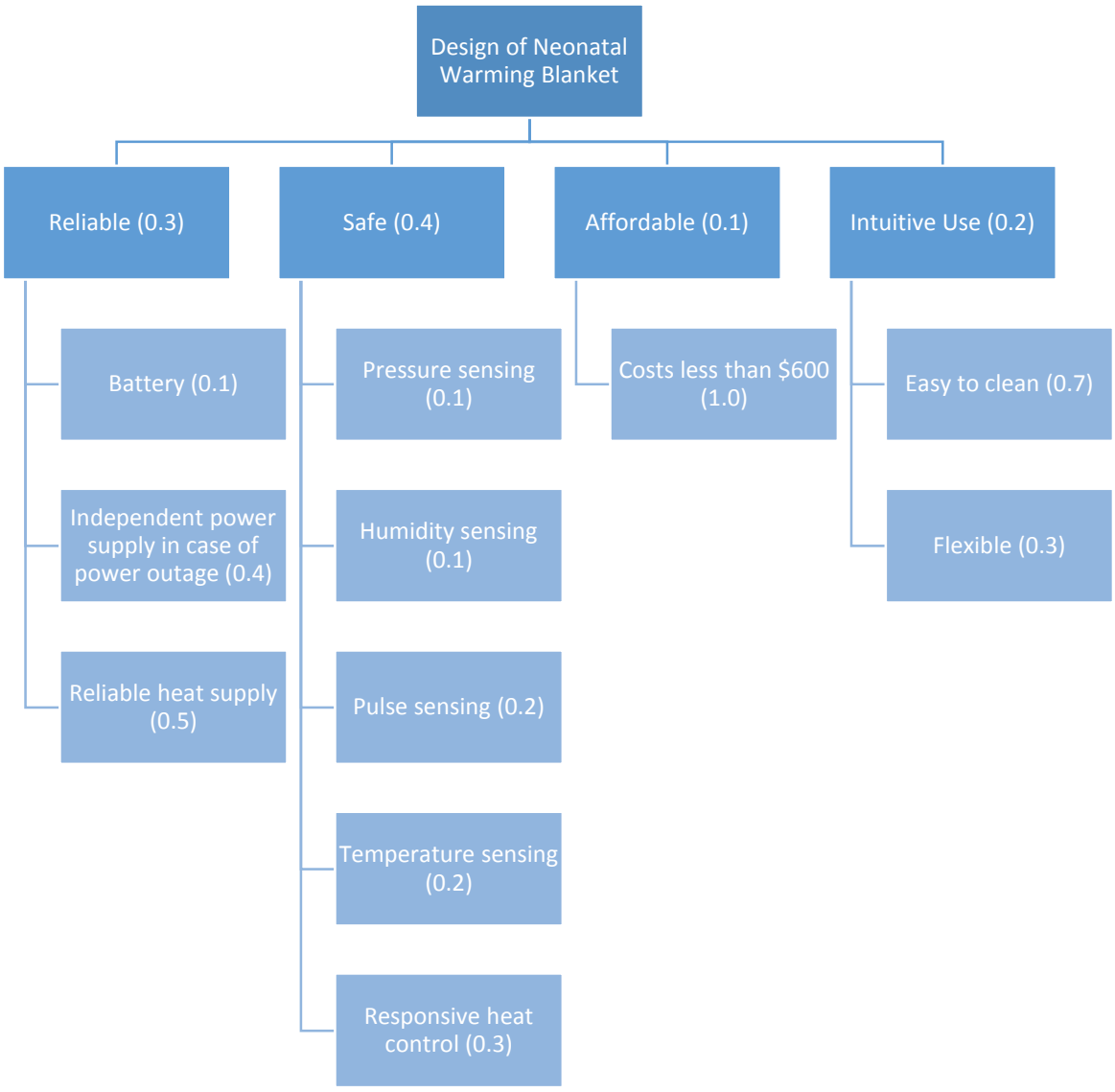


Figure 3-3: Objective tree outlining various components or features of the neonatal warming blanket, accompanied by the associated importance weighting.

Table 3-2 shows the resultant weighted scores, as well as the calculations used to determine these scores. The rows highlighted in green have been identified as the most important elements of the

design and the rows not highlighted are those which are neither critical to the success of the project, nor are they unimportant.

Criteria	Calculation of Final Weighted Score	Final Weighted Score
Battery	0.1*0.3	0.03
Independent power supply in case of power outage	0.4*0.3	0.12
Reliable heat supply	0.5*0.3	0.15
Pressure sensing	0.1*0.4	0.04
Humidity sensing	0.1*0.4	0.04
Pulse sensing	0.2*0.4	0.08
Temperature sensing	0.2*0.4	0.08
Responsive heat control	0.3*0.4	0.12
Costs less than $300	1.0*0.1	0.10
Easy to clean	0.7*0.2	0.14

Criteria	Calculation of Final Weighted Score	Final Weighted Score
Flexible	0.3*0.2	0.06

Table 3-2: Calculation of weighted scores for each element of the objective tree, with the most important elements highlighted in green, according to Decision Theory (Parmigiani, 2001)[18].

[18] Decision Theory utilises the principles adopted in statistics and probability in order to deduce the most appropriate option out of a variety of options (Parmigiani, 2001). These practices are employed when there is an element of uncertainty in the decision being made. For example, the student was unsure which design elements would be most important to the success of the prototype and so, by completing the objective tree, these decisions were made clear. Essentially, Decision Theory is based on the idea that uncertainty can be reduced or even eliminated when additional information is acquired, such as the weighting scores shown above.

While not all of these elements can be implemented in this thesis due to the prototype nature of the project, these remain important considerations for future work.

3.5 DESIGN REQUIREMENTS (ESSENTIAL AND DESIRED)

To justify the integration of a new design in the neonatal field, the proposed solution needed to meet current system requirements, and introduce an element of novelty such that the design offered benefits that no other commercially available warming device or blanket did.

The design needed to offer fast warm-up times, as well as being able to accurately monitor the body temperature of the neonate. The blanket connected with a user interface which monitored the temperature of the neonate. The design also incorporated additional sensor functionality in terms of humidity and pulse.

In terms of material requirements, the blanket needed to envelop the neonate and remain flexible enough to do so. This needed to remain the case despite the inclusion of a heating element. Each of these requirements specified by the Flinders Medical Centre were entered into a Priority List, so that the essential items in the list were easily distinguishable from the desires which were not crucial to the success of the project.

This Priority List is depicted by Table 3-3, where the essential requirements were those that cannot be compromised, as these were deemed crucial to the success of the project. The desired features, on the other hand, were those that would enhance the form or function of the product, but the lack of these features did not result in an unsuccessful project.

Essential Requirements	Desired Features
Temperature sensing	Pressure sensing
Heating element	Data logging
Alarm system	Aesthetically pleasing

Essential Requirements	Desired Features
Flexible	Battery powered
Affordable (within project budget)	Fast cool-down time
Low power consumption	Pulse sensing
Fast warm-up time	Humidity sensing
	Washable

Table 3-3: List of essential and desired requirements, as specified by the student and supervisors of the project.

3.6 DESIGN SPECIFICATIONS

The design requirements and customer desires outlined above were then associated with set values and metrics, in the form of the list of specifications shown in Table 3-4. This enabled the further refinement of the design, as well as allowing specific components to be selected, based on their ability to meet these specifications.

Metric	Value	Justification
Length of Blanket	40-55 cm	Typical newborn crown-heel length is around 50 cm (*Newborn Measurements*, no date), so this would likely be at the 'longer end' of premature neonate body length. "A length of less than 47 cm is a sign of prematurity" (*Important measurements of a newborn*, 2014).
Width of Blanket	30 cm	Once again considering the dimensions of premature neonates, this dimension was determined as providing adequate closeness to the neonate's skin to provide warmth, while also providing flexibility to fit larger neonates if required.
Weight	< 5 kg	This value was small enough that the blanket remained easily portable, while remaining realistic and leaving room for the material of the blanket to be decided and tested.
Steps required to operate	1	Only turn-on required. This ensured the simplicity of the design, and ensured that expert knowledge was not required to use the product.
Minimum number of temperature sensors	3	Sufficient temperature sensors needed to be included in the blanket design such that the temperature at various points of the neonate's body could be measured. Three of these sensors ensured that an average body temperature could be obtained. .

Metric	Value	Justification
Minimum number of pulse sensors	1	A single pulse sensor was required to obtain the pulse rate of the neonate. Unlike temperature, the pulse of the neonate does not vary with the location of the sensor on the body, so one pulse sensor was deemed appropriate.
Minimum number of humidity sensors	2	At least two humidity sensors were required, as one was placed inside the blanket, and one was placed outside of the blanket. This enabled the comparison of humidity in the immediate vicinity of the neonate (inside the blanket) and the humidity of the surrounding environment.
Temperature sampling rate	≤ 1 second	Temperature variations can happen extremely quickly depending on the mode of heat loss that the neonate is exposed to. For this reason, a high sampling rate was required to capture any changes in body temperature.
Pulse sampling rate	≤ 1 second	Similarly to temperature changes, pulse changes can happen quickly and with little warning. Additionally, premature neonates have an average heart rate of around 200 beats per minute (T Carlisle 2018, personal communication, August 17). For these reasons, the sampling rate of the pulse sensor was high.

Metric	Value	Justification
Humidity sampling rate	≤ 10 seconds	Humidity changes occur more slowly than temperature and pulse changes. Furthermore, humidity changes are far less critical, and so a general trend in humidity should suffice in order to ensure the safety of the neonate.
Temperature sensor resolution	≤ 0.1 °C	The temperature of a neonate can change from healthy to critical within a single degree, and so it was important to ensure that any changes were recognised by the sensor within this range. For this reason, a high resolution was required.
Pulse sensor resolution	≤ 2 bpm	A healthy heart rate, and a critical heart rate, are typically around 150 to 200 beats per minute apart. For this reason, a resolution of around 2 beats per minute was sufficient in order to capture any changes that could affect the wellbeing of the neonate.
Humidity sensor resolution	≤ 5%	Changes in humidity are not fatal, and are merely important to consider as a means of preventing pressure sores. Therefore, the resolution of this humidity sensor could be quite large, so long as it captured the general trend of changing humidity.

Metric	Value	Justification
Temperature sensor accuracy	±0.1 °C (maximum)	Temperature variations in neonates can disrupt growth and development, and can also prove fatal if these temperature variations lead to hypo- or hyperthermia. Additionally, the changes in temperature resulting in either condition can occur within a change of a single degree, so the accuracy of this sensor needed to be as high as possible.
Pulse sensor accuracy	±5 bpm (maximum)	If a neonate has an unhealthy or irregular pulse rate, the trend can be deduced before the change proves fatal.
Humidity sensor accuracy	±10% RH (maximum)	Humidity is far less important when caring for more stable neonates, such as those no longer requiring intensive care. Therefore, the humidity sensor could be less accurate as it was required to show generalised trends and changes, and exact measurements were not crucial.
Production cost	≤ $600	This was the project budget set by the University, and this was adhered to in order to ensure that all components could be purchased and obtained.
Warm-up time	≤10 minutes	This is the average time taken for commercial warming blankets to heat up.

Table 3-4: Product specifications established using the requirements and desires specified previously.

3.7 FUNCTION STRUCTURE AND ANALYSIS

Figure 3-4 details the conversion of the requirements discussed into solution neutral functions.

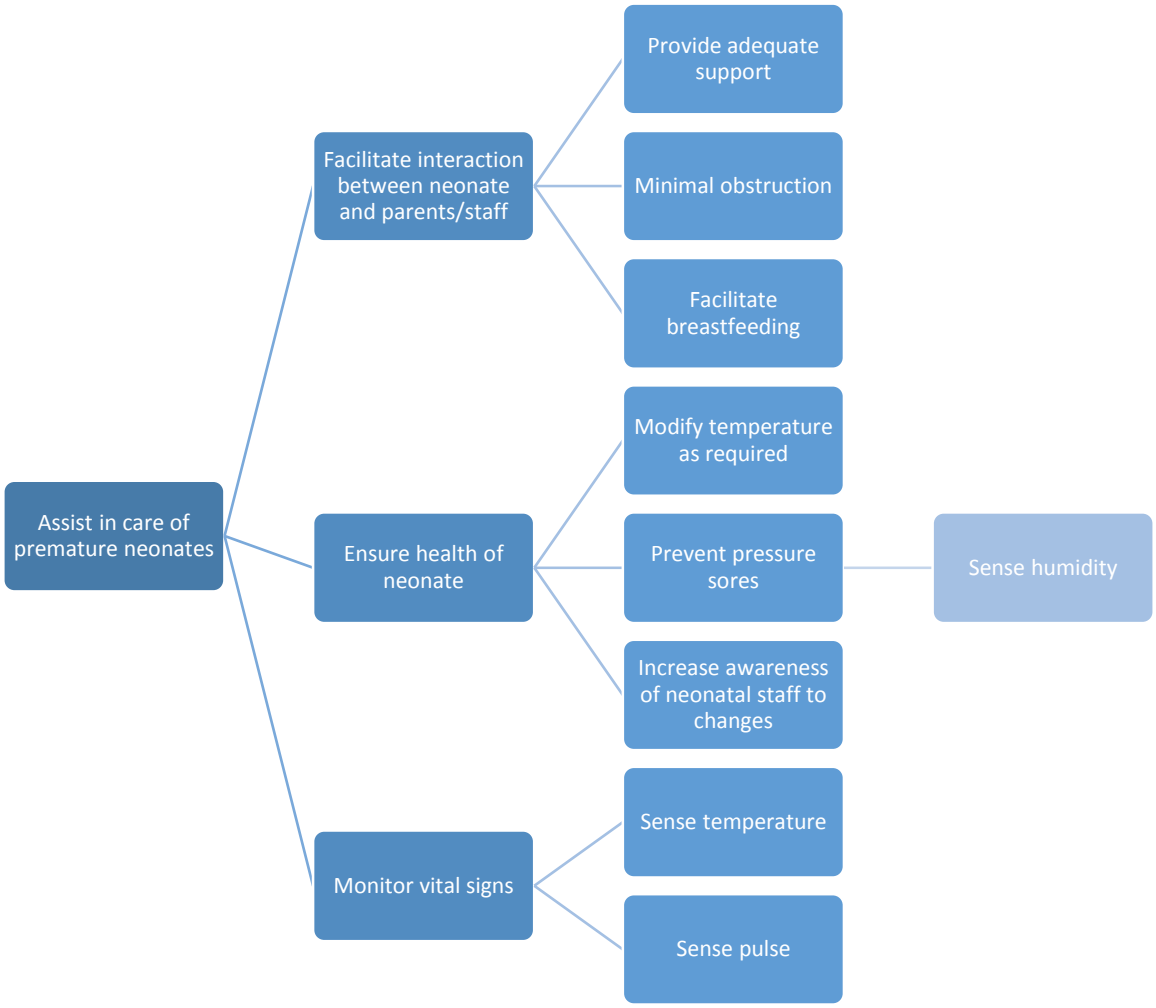


Figure 3-4: Function structure of neonatal warming blanket, in terms of key functionality.

3.8 RISK ANALYSIS

The following portion of the thesis will detail the risks associated with the design and development of the neonatal warming blanket. Figure 3-6 depicts these risks, as well as their associated categorization and compensation methodology. This categorization was assigned based on the risk assessment matrix commonly used in industry, and shown in Figure 3-5.

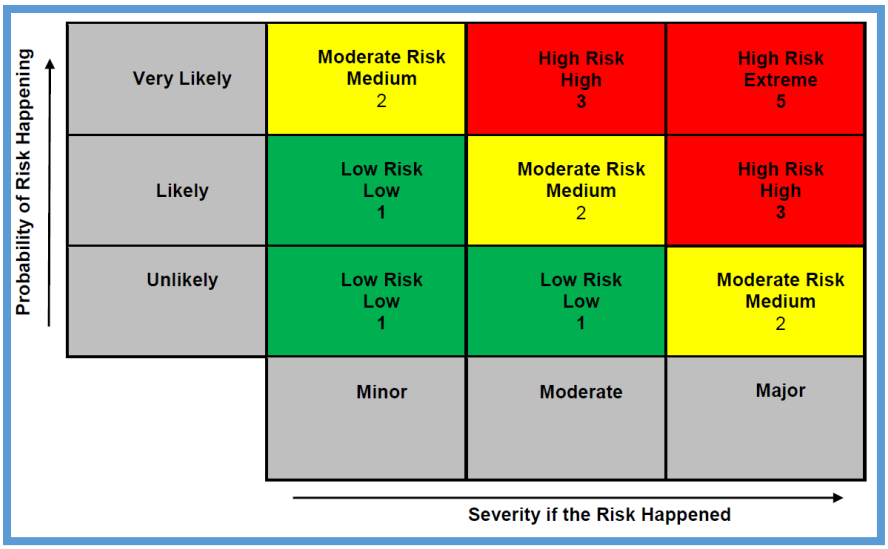
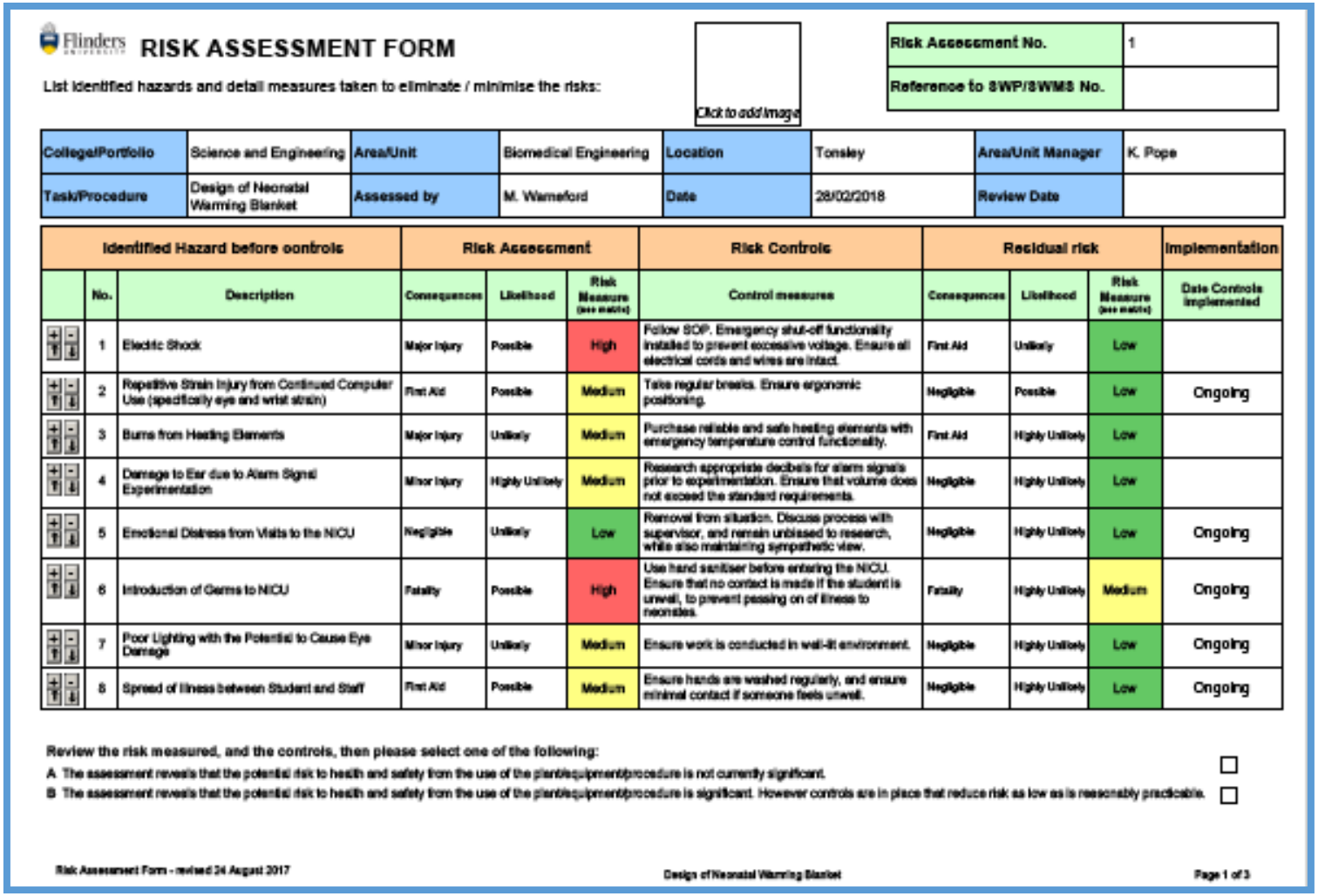


Figure 3-5: Tool used to categorize identified risks based on the probability of their occurrence, and severity for any consequences that may occur (*Risk Assessment Guidelines for Assessing and Addressing Working Alone Risk*, 2017).

Risks identified as very likely or likely with moderate to major consequences have been highlighted in red. These were deemed irrational for the thesis, and so compensation methods were employed in order to reduce the severity or likelihood of these risks, thus removing them from the high risk category. Furthermore, a risk very likely to occur, with minor consequences, was deemed a moderate risk. This was also the case for a likely occurrence and medium severity, as well as an unlikely occurrence with major consequences. These moderate risks were highlighted in yellow in both Figure 3-5 and Figure 3-6, and were deemed appropriate only where there was no chance of demoting the risk to the low risk category, and where there was no alternative method of completing the task.



Flinders UNIVERSITY

RISK ASSESSMENT FORM

List identified hazards and detail measures taken to eliminate / minimise the risks:

Click to add image

Risk Assessment No.	1
Reference to SWP/SWMS No.	

College/Portfolio	Science and Engineering	Area/Unit	Biomedical Engineering	Location	Tonsley	Area/Unit Manager	K. Pope
Task/Procedure	Design of Neonatal Warming Blanket	Assessed by	M. Warneford	Date	28/02/2018	Review Date	

Identified Hazard before controls			Risk Assessment			Risk Controls	Residual risk			Implementation
	No.	Description	Consequences	Likelihood	Risk Measure (see matrix)	Control measures	Consequences	Likelihood	Risk Measure (see matrix)	Date Controls Implemented
+ -	1	Electric Shock	Major Injury	Possible	High	Follow SOP. Emergency shut-off functionality installed to prevent excessive voltage. Ensure all electrical cords and wires are intact.	First Aid	Unlikely	Low	
+ -	2	Repetitive Strain Injury from Continued Computer Use (specifically eye and wrist strain)	First Aid	Possible	Medium	Take regular breaks. Ensure ergonomic positioning.	Negligible	Possible	Low	Ongoing
+ -	3	Burns from Heating Elements	Major Injury	Unlikely	Medium	Purchase reliable and safe heating elements with emergency temperature control functionality.	First Aid	Highly Unlikely	Low	
+ -	4	Damage to Ear due to Alarm Signal Experimentation	Minor Injury	Highly Unlikely	Medium	Research appropriate decibels for alarm signals prior to experimentation. Ensure that volume does not exceed the standard requirements.	Negligible	Highly Unlikely	Low	
+ -	5	Emotional Distress from Visits to the NICU	Negligible	Unlikely	Low	Removal from situation. Discuss process with supervisor, and remain unbiased to research, while also maintaining sympathetic view.	Negligible	Highly Unlikely	Low	Ongoing
+ -	6	Introduction of Germs to NICU	Fatality	Possible	High	Use hand sanitiser before entering the NICU. Ensure that no contact is made if the student is unwell, to prevent passing on of illness to neonates.	Fatality	Highly Unlikely	Medium	Ongoing
+ -	7	Poor Lighting with the Potential to Cause Eye Damage	Minor Injury	Unlikely	Medium	Ensure work is conducted in well-lit environment.	Negligible	Highly Unlikely	Low	Ongoing
+ -	8	Spread of Illness between Student and Staff	First Aid	Possible	Medium	Ensure hands are washed regularly, and ensure minimal contact if someone feels unwell.	Negligible	Highly Unlikely	Low	Ongoing

Review the risk measured, and the controls, then please select one of the following:

A The assessment reveals that the potential risk to health and safety from the use of the plant/equipment/procedure is not currently significant. ☐

B The assessment reveals that the potential risk to health and safety from the use of the plant/equipment/procedure is significant. However controls are in place that reduce risk as low as is reasonably practicable. ☐

Risk Assessment Form - revised 24 August 2017 | Design of Neonatal Warming Blanket | Page 1 of 3

Figure 3-6: Risk assessment for neonatal warming blanket design and prototyping.

3.9 COSTS AND BUDGETING

The costs incurred throughout the project have been outlined in Table 3-5. Careful budgeting and consulting throughout the project was required to ensure that the project fell within the budget constraints set by the University.

Item	Quantity	Total Cost ($AUD)
Arduino Bluno Mega 2560 (*Bluno Mega 2560 - A Bluetooth 4.0 Micro-controller Compatible with Arduino Mega Australia*, no date)	1	72.41
DHT22 Temperature and Relative Humidity Sensor Module (*DHT22 Temperature and Relative Humidity Sensor Module Australia*, no date)	2	26.36
MCP9808 High Accuracy I2C Temperature Sensor Breakout Board (MCP9808 High Accuracy I2C Temperature Sensor Breakout Board Australia, no date)	5	42.50
Pressure-Sensitive Conductive Sheet (Velostat/Linqstat) (*Pressure-Sensitive Conductive Sheet (Velostat/Linqstat) Australia*, no date)	1	6.34
Heating Pad (5x10cm) (*Heating Pad - 5x10cm - SparkFun COM-11288: SparkFun Australia*, no date)	5	36.20
Pulse Sensor SEN11574 (*Pulse Sensor Australia*, no date)	1	42.44
On-Off Power Button/Pushbutton Toggle Switch (*On-Off Power Button / Pushbutton Toggle Switch Australia*, no date)	2	5.70

Item	Quantity	Total Cost ($AUD)
Gravity: Arduino Digital Buzzer (*Gravity: Arduino Digital Buzzer DFR0032: Arduino Australia - In Stock | Little Bird Electronics*, no date)	2	11.42
MOSFET Transistor, N Channel, 15.7 A, 60 V, 0.042 ohm, 10 V, 1 V	5	9.65
Instrument Amplifier, 1 Amplifier, 500 µV, 0.02 V/µs, 150 kHz, ± 1.35V to ± 18V, DIP	1	11.40
Customised PCB	5	30
Wired DHT22 Sensors (AM2302)	2	50
Heat Shrink	N/A	N/A*
Solder Wick	N/A	N/A**
100K Ohm resistors	10	N/A*

Item	Quantity	Total Cost ($AUD)
100K Ohm resistors	10	N/A*
Breadboard wires	10+	N/A**
Breadboard	1	N/A*
Neonatal Blanket Material	1	N/A**
Total	N/A	344.42
Budget	N/A	600
Remaining	N/A	255.58

Table 3-5: Outline of the costs associated with the project, including a justification for each cost. Items with cost listed as N/A* or N/A** are supplied by the University or student respectively, and therefore incur no additional cost.

4 DESIGN

4.1 SOFTWARE DEVELOPMENT APPROACH

The software development portion of this thesis was carried out by adhering to the software development planning actions utilized in industry. Namely, these steps are outlined in Figure 3-7.

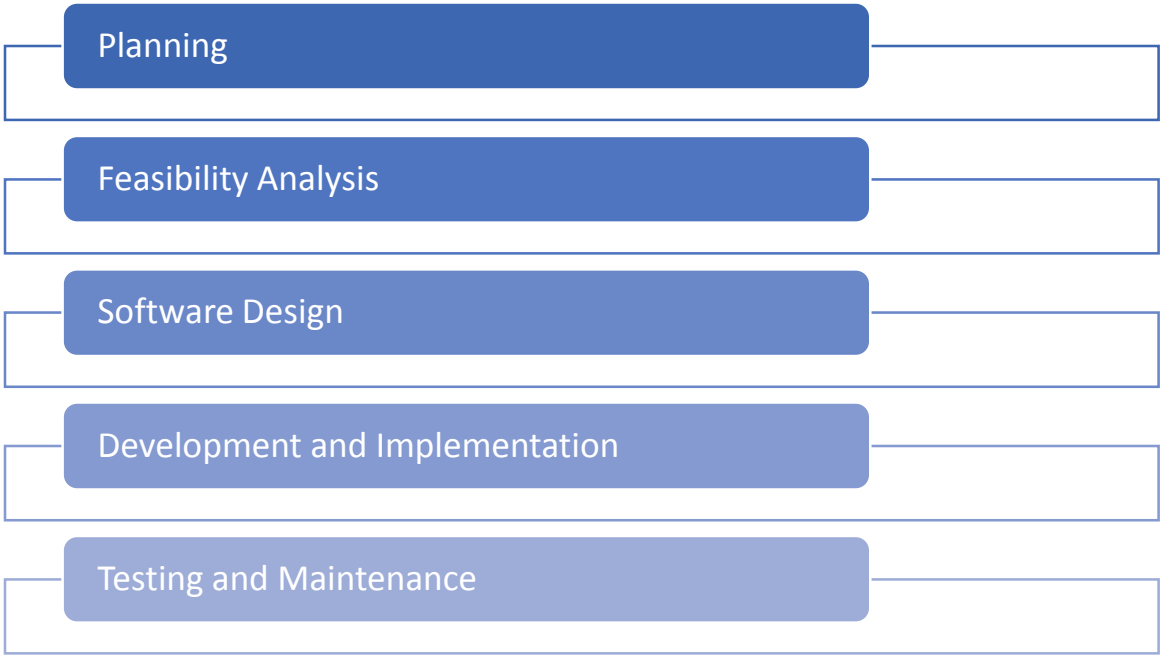


Figure 3-7: Software development methodology.

"Without the perfect plan, calculating the strengths and weaknesses of the project, development of software is meaningless" (*6 Stages of Software Development Process*, no date). Therefore, the initial phase of software development involved brainstorming the requirements of the software, and any issues that could have arisen at any time during the planning or implementation of the software.

4.1.1 SELECTION OF INTEGRATED DEVELOPMENT ENVIRONMENT

An Integrated Development Environment (IDE) is the program used to write the software controlling the Arduino. Although many features are shared between IDEs, there are some distinguishing features that may make a particular IDE most applicable to the current design. The various IDE options were

therefore analysed and compared, and the advantages and disadvantages of each have been listed in Table 3-6.

IDE	Advantages	Disadvantages
NetBeans	Student has past experience Functions well with multiple screens Supports multiple languages Free and open source	Slow debug (Garancsi, Robinson and Maldonado, no date)
Processing 3.0	Free and open source Can draw two and three dimensional graphics Extensive libraries and tools (*Environment (IDE)\Processing*, no date) Intuitive text editor Extensive documentation	"Low level" (*What are the advantages of using Processing? - Processing 2.x and 3.x Forum*, no date) Extensive code can cause the IDE to slow down

IDE	Advantages	Disadvantages
Eclipse	Supports multiple languages Free and open source Fast debug Configurable user interface Ideal for Java (*Battle Of The IDEs: Xcode Vs. Eclipse*, 2013)	Uses large amount of computer power (*Why Eclipse instead of word processors and command prompt?*, 2012) All plugins need to be installed separately (Garancsi, Robinson and Maldonado, no date) "Steep learning curve" (*Battle Of The IDEs: Xcode Vs. Eclipse*, 2013)
Microsoft Visual Studio	Free and open source Supports multiple platforms (Simmons, 2017) Student has past experience	Long set-up time Limited automatic formatting (Volkov, Dovgodko and Super, 2017)

IDE	Advantages	Disadvantages
IntelliJ IDEA	Code completion functionality Analysis of code as it is written Support multiple languages Version control (*What's New in IntelliJ IDEA*, 2018) Indexes the project for the user (Solntsev, 2012) Automatically builds syntax tree Automatic code completion	Uses many resources, and can cause slow processing time (Solntsev, 2012)

IDE	Advantages	Disadvantages
Xcode	Does not require excessive computer power Significant and easily accessible documentation Code step functionality in debug mode (*Battle Of The IDEs: Xcode Vs. Eclipse*, 2013) Includes iPhone simulator Code completion functionality Ideal for Objective-C (*Battle Of The IDEs: Xcode Vs. Eclipse*, 2013)	Free only on Apple computers Only compatible with iOS[19] (*Battle Of The IDEs: Xcode Vs. Eclipse*, 2013) Non-configurable display

[19] iOS is "an operating system used for mobile devices manufactured by Apple Inc." (*What is ios? | Infosys Technologies Ltd Interview Question*, no date).

IDE	Advantages	Disadvantages
Arduino	Simple and easy to use Cross-platform Open source (*Arduino Development Kits: Advantages and Features*, 2014) Student experience in the coding language (C/C++) (*What are the disadvantages of Arduino?*, no date)	Simplicity can be detrimental for more complex projects Limited debugging functionality "Arduino libraries are not very efficient in certain parts and waste RAM and CPU cycles" (*What are the disadvantages of Arduino?*, no date)
Komodo	Ideal for web languages (*NetBeans IDE vs Komodo IDE 2018 Comparison*, 2018) Code completion functionality Cross platform (*NetBeans vs Komodo IDE detailed comparison*, 2018) Supports multiple languages Integrated debugger (*NetBeans vs Komodo IDE detailed comparison*, 2018)	Unable to select default directory upon opening of the IDE (Ross, 2017) Poor menu layout (non-intuitive)

IDE	Advantages	Disadvantages
Qt Creator	Significant amount of help online (Wezorek and Connors, 2013) Thorough and easily accessible documentation Supports multiple platforms	Aesthetics disliked by many users (Wezorek and Connors, 2013) Complicated to pair with C++ ("requires separate compilation step") (*Why aren't more desktop apps written with Qt?*, no date) Qt syntax is not always recognized by C++ IDEs (*Why aren't more desktop apps written with Qt?*, no date)

Table 3-6: Brief overview of the advantages and disadvantages of accessible Integrated Development Environments (IDEs).

As highlighted in Table 3-6, there were two particular IDEs that were selected for use during this project. Namely, the Processing 3.0 IDE and Arduino IDE were used in conjunction, and for two different portions of software development. The Arduino IDE was used to store all sensor data, and initialise the temperature control algorithm. This IDE was selected due to past experience, which made the initial software development phase straight forward and efficient. C++ was used to write code in this IDE.

The Processing 3.0 IDE, on the other hand, was used to display sensor data to the user using Java code. This IDE was selected after it was realised that its functionality was much more suited to the creation of user interfaces than the Arduino IDE.

The decision to select these two IDEs was more comprehensive than the advantage and disadvantages shown in Table 3-6, however. The intricacies of this decision are detailed further in Table 3-7.

IDE		Cost ($AUD)	Student Experience	Supported Languages	Documentation	Average Weighting
	Weighting	**0.10**	**0.40**	**0.10**	**0.40**	**N/A**
NetBeans	Value	0.00	Yes	"Java desktop, mobile, and web applications, as well as HTML5 applications with HTML, JavaScript, and CSS. The IDE also provides a great set of tools for PHP and C/C++ developers" (*NetBeans IDE - Overview*, no date)	Excellent	N/A
	Rating	1.00	1.00	1.00	1.00	N/A

IDE		Cost ($AUD)	Student Experience	Supported Languages	Documentation	Average Weighting
	Weighting	**0.10**	**0.40**	**0.10**	**0.40**	**N/A**
	Weighted Rating	0.10	0.40	0.10	0.40	0.25
Processing 3.0	Value	0.00	No	Processing Code	Excellent	N/A
	Rating	1.00	0.00	0.50	1.00	N/A
	Weighted Rating	0.10	0.00	0.05	0.40	0.11

IDE		Cost ($AUD)	Student Experience	Supported Languages	Documentation	Average Weighting
	Weighting	**0.10**	**0.40**	**0.10**	**0.40**	**N/A**
Eclipse	Value	0.00	No	Java, Ada, ABAP, C, C++, C#, COBOL, D, Fortran, Haskell, JavaScript, Julia, Lasso, Lua, NATURAL, Perl, PHP, Prolog, Python, R, Ruby, Rust, Scala, Clojure, Groovy, 1Scheme, and Erlang	Excellent	N/A
	Rating	1.00	0.00	1.00	1.00	N/A
	Weighted Rating	0.10	0.00	0.10	0.40	0.15

IDE		Cost ($AUD)	Student Experience	Supported Languages	Documentation	Average Weighting
	Weighting	**0.10**	**0.40**	**0.10**	**0.40**	**N/A**
Microsoft Visual Studio	Value	0.00	Yes	C, C++, C++/CLI, Visual Basic .NET, C#, F#, JavaScript, TypeScript, XML, XSLT, HTML, CSS, Python, Ruby, Node.js, and M	Excellent	N/A
	Rating	1.00	1.00	1.00	1.00	N/A
	Weighted Rating	0.10	0.40	0.10	0.40	0.25

IDE		Cost ($AUD)	Student Experience	Supported Languages	Documentation	Average Weighting
	Weighting	**0.10**	**0.40**	**0.10**	**0.40**	**N/A**
IntelliJ IDEA	Value	Free (Demmer, 2014)	No	Java, CloudSlang, Clojure, Dart, Erlang, Go, Groovy, Haxe, Perl, Scala, XML/XSL, Kotlin, Haskell, Lua and Python	Excellent	N/A
	Rating	1.00	0.00	0.60	1.00	N/A
	Weighted Rating	0.10	0.00	0.06	0.40	0.14

IDE		Cost ($AUD)	Student Experience	Supported Languages	Documentation	Average Weighting
	Weighting	**0.10**	**0.40**	**0.10**	**0.40**	**N/A**
Xcode	Value	Free open source version or $130 AUD for added functionality (*Choosing a Membership - Apple Developer*, no date)	No	C, C++, Objective-C, Objective-C++, Java, AppleScript, Python, Ruby, Rez, Swift, GNU Pascal, Ada, C#, Perl and D	Moderate	N/A
	Rating	1.00	0.00	1.00	0.50	N/A
	Weighted Rating	0.10	0.00	0.10	0.20	0.10

IDE		Cost ($AUD)	Student Experience	Supported Languages	Documentation	Average Weighting
	Weighting	**0.10**	**0.40**	**0.10**	**0.40**	**N/A**
Komodo	Value	Free open source version or $130.00 AUD for advanced features (*Komodo IDE Pricing | G2 Crowd*, no date)	No	Python, Perl, PHP, Ruby, Tcl, SQL, Smarty, CSS, HTML, XML	Excellent	N/A
	Rating	1.00	0.00	0.60	1.00	N/A
	Weighted Rating	0.10	0.00	0.06	0.40	0.14

IDE		Cost ($AUD)	Student Experience	Supported Languages	Documentation	Average Weighting
	Weighting	**0.10**	**0.40**	**0.10**	**0.40**	**N/A**
Qt Creator	Value	LGPLV3 and GPLV2/GPLV3 are free, Commercial licensing starts from $459.00/month	No	C++, QML, ADA, BASIC (Beginner's All-Purpose Symbolic Instruction Code), C#, D, Go, Haskell, Java, JavaScript, Lisp and Lua	Excellent	N/A
	Rating	1.00	0.00	1.00	1.00	N/A
	Weighted Rating	0.10	0.00	0.10	0.40	0.15
Arduino IDE	Value	0.00	Yes	C/C++	Moderate	N/A

IDE		Cost ($AUD)	Student Experience	Supported Languages	Documentation	Average Weighting
	Weighting	**0.10**	**0.40**	**0.10**	**0.40**	**N/A**
	Rating	1.00	1.00	1.00	0.70	N/A
	Weighted Rating	0.10	0.40	0.10	0.28	0.22

Table 3-7: Decision matrix comparing common Integrated Development Environments.

While the highlighted IDE (Processing 3.0) is not the highest scoring IDE of those investigated, it was chosen due to its ability to create GUIs quickly and effectively. This was ideal as well as the fact that there were libraries in the Processing IDE than enabled simple communication with the Arduino IDE.

4.2 FEASIBILITY ANALYSIS

A feasibility analysis was carried out to ensure that the software functionality and form devised in the brainstorming phase was possible. "The aims of a feasibility study are to find out whether the system is worth implementing and if it can be implemented, given the existing budget and schedule" (*Feasibility study*, no date). Rather than detailing the feasibility of the project in a general sense, this chapter deals only with the feasibility of the software component, as detailed in Table 3-8.

Element for Consideration	Description
Technical Feasibility	Two Integrated Development Environments (IDEs) were selected (Arduino and Processing, respectively). Prior experience with the Arduino IDE meant that this was found to be a straightforward option for programming the Arduino microcontroller. The Processing IDE was selected after extensive research on GUI creation. Prior experience and research has also shown that the technical experience developed during a degree in Biomedical Engineering has proved sufficient when writing software using these IDEs. Furthermore, the opportunity of gaining technical staff assistance increased the technical feasibility of this project.
Social Feasibility	The social feasibility of a project is important as it relates to the impact of the software and hardware on the potential users of the system (in this case, neonatal staff and parents as well as the neonates themselves). In the case of the proposed software, the interface was very simple and intuitive to use. Colors enabled simple translation of numerical values, and all data was displayed on one screen, so minimal retraining of staff would be necessary if the device were to be adopted. In terms of the safety of neonates, this will be ensured through testing in future studies of the system.
Economic Feasibility	This portion of the feasibility study relates to the financial requirements of the proposed software, and whether or not these requirements are realistic given the project budget. Firstly, the IDEs selected are available in community editions, and so it was possible to access a free version for the duration of the project. Additionally, the student was responsible for the software writing portion of the project, so no additional labour costs were involved where they might be in industry.

Table 3-8: Feasibility study broken down into technical, social and economic feasibility.

4.2.1 SOFTWARE DESIGN

In order to approach any software design task, a number of subsections are typically devised in order to separate the code into more manageable and realistic divisions. These divisions are detailed in Table 3-9.

Software Subsection	Details
Sensor setup	This phase involved the initial declarations required to run the various sensors. In all cases, this involved the creation of objects in the code, which would be responsible for the storage of sensor data.
Data transmission	The serial library was used in the Arduino and Processing IDEs to transmit and receive data respectively. This library enabled communication between the IDE and a computer, or other device. The UART ports on the Arduino microcontroller were responsible for this serial data communication, and these ports communicated with the transmitter and receiver pins on the microcontroller, and with the computer via a micro USB port (as was the case in this project). "The Arduino Mega has three additional serial ports: Serial1 on pins 19 (RX) and 18 (TX), Serial2 on pins 17 (RX) and 16 (TX), Serial3 on pins 15 (RX) and 14 (TX)" (*Serial*, no date).
Data reception	Serial.serialEvent() is a function used to detect when there is data available for receiving. Serial.read() could then be called to receive this data. The Serial.print() function was used to print this data to the serial monitor in the Arduino and Processing IDEs, and text() can be used to write this data to the GUI for display to the user.
GUI	The Processing IDE was used to create the initial versions of the user interface.

Table 3-9: Division of software objectives into smaller, more manageable tasks.

4.3 MICROCONTROLLER SELECTION

There are a plethora of microcontrollers that are commercially available, and readily accessible. It was important to compare these based on their ability to abide by the project budget, the required ports, and the required memory for the project. An effective method of determining which microcontroller is best suited to the application was the use of a decision matrix. The decision matrix depicted in Table 3-10 was used in order to determine the ideal microcontroller for this thesis.

There were five main criteria decided to be most significant when determining ideality: cost, operating voltage, number of analog ports, number of digital ports, and memory capacity. These criteria were then given a weighting between 0 and 1 corresponding to their importance in the final selection. In this case, cost was given a weighting of 0.20, operating voltage a rating of 0.10, analog ports 0.20, digital ports 0.20 and, lastly, memory capacity was given a rating of 0.30.

Each of the potential microcontrollers was then given a rating between 0 and 1 for each of these criteria. In order to determine the final weighted score of any given microcontroller, the weighted scores specific to that microcontroller were first multiplied by the weighting factor of any given criterion. The weighted scores for each criterion for a given microcontroller were then averaged in order to deduce a final weighted score for each option.

Microcontroller		Cost ($AUD)	Operating Voltage (V)	Number of Analog Input/Output Ports	Number of Digital Input/Output Ports	Flash Memory Capacity (KB)	Average Weighted Rating
	Weighting	**0.20**	**0.10**	**0.20**	**0.20**	**0.30**	**N/A**
Bluno Mega 2560 (*Bluno Mega 2560 - A Bluetooth 4.0 Micro-controller Compatible with Arduino Mega Australia*, no date)	Value	72.41	5.00	16.00	54.00	256.00	N/A
	Rating	0.70	1.00	1.00	0.80	0.50	N/A
	Weighted Rating	0.14	0.10	0.20	0.16	0.15	0.15
Arduino Due (*Arduino Due - DEV-11589 - SparkFun Electronics*, no date)	Value	72.95	3.30	12.00	54.00	512.00	N/A
	Rating	0.70	0.20	0.80	0.80	0.70	N/A
	Weighted Rating	0.14	0.02	0.16	0.16	0.21	0.14

Microcontroller		Cost ($AUD)	Operating Voltage (V)	Number of Analog Input/Output Ports	Number of Digital Input/Output Ports	Flash Memory Capacity (KB)	Average Weighted Rating
	Weighting	**0.20**	**0.10**	**0.20**	**0.20**	**0.30**	**N/A**
BeagleBone Black - Rev C (*BeagleBone Black - Rev C - DEV-12857 - SparkFun Electronics*, no date)	Value	112.55	3.30	7.00	65.00	$4.00*10^6$	N/A
	Rating	0.20	0.20	0.10	1.00	1.00	N/A
	Weighted Rating	0.04	0.02	0.02	0.20	0.30	0.12
MicroPython pyboard v1.1 (with Headers) (*MicroPython pyboard v1.1 (with Headers) - DEV-14413 - SparkFun Electronics*, no date)	Value	91.11	N/A	N/A	N/A	1024.00	N/A
	Rating	0.50	0.00	0.00	0.00	0.90	N/A
	Weighted Rating	0.10	0.00	0.00	0.00	0.27	0.07

Microcontroller		Cost ($AUD)	Operating Voltage (V)	Number of Analog Input/Output Ports	Number of Digital Input/Output Ports	Flash Memory Capacity (KB)	Average Weighted Rating
	Weighting	**0.20**	**0.10**	**0.20**	**0.20**	**0.30**	**N/A**
Raspberry Pi 3 Model B+ (*Raspberry Pi 3 Model B+ - Raspberry Pi*, no date)	Value	54.96	5.00	N/A	N/A	N/A	N/A
	Rating	0.80	1.00	0.00	0.00	0.00	N/A
	Weighted Rating	0.16	0.10	0.00	0.00	0.00	0.05
Raspberry Pi Zero W (*Raspberry Pi Zero W - Raspberry Pi*, no date)	Value	14.96	5.00	N/A	N/A	N/A	N/A
	Rating	1.00	1.00	0.00	0.00	0.00	N/A
	Weighted Rating	0.20	0.10	0.00	0.00	0.00	0.06

Microcontroller		Cost ($AUD)	Operating Voltage (V)	Number of Analog Input/Output Ports	Number of Digital Input/Output Ports	Flash Memory Capacity (KB)	Average Weighted Rating
	Weighting	**0.20**	**0.10**	**0.20**	**0.20**	**0.30**	**N/A**
Arduino MKR WAN 1300 (*Arduino MKR WAN 1300 (LoRa connectivity)*, no date)	Value	53.47	5.00	8.00	8.00	256.00	N/A
	Rating	0.80	1.00	0.10	0.10	0.50	N/A
	Weighted Rating	0.16	0.10	0.02	0.02	0.15	0.09
Arduino MKR GSM 1400 (*Arduino MKR GSM 1400*, no date)	Value	93.67	5.00	8.00	8.00	256.00	N/A
	Rating	0.50	1.00	0.10	0.10	0.50	N/A
	Weighted Rating	0.10	0.10	0.02	0.02	0.15	0.08

Table 3-10: Decision matrix comparing commercially available microcontrollers.

4.4 TEMPERATURE SENSING

4.4.1 TEMPERATURE SENSOR SELECTION

In industry, most medical temperature sensors are NTC thermistors (YSI400 series) as they are pre-calibrated, pre-sterilised, small, single-use and affordable. However, the programming capabilities with these sensors is limited, and so other options were analysed in order to determine the most suitable sensor for this application. The options investigated are shown in Table 3-11.

Sensor Name		Cost ($AUD)	Operating Voltage (V)	Operating Range (°C)	Accuracy (°C)	Average Weighted Rating
	Weighting	**0.20**	**0.20**	**0.10**	**0.50**	**N/A**
Arduino Temperature Sensor Module TA0061 (*Arduino Temperature Sensor Module*, no date)	Value	4.35	5.00	-55.00 to 125.00	+/- 0.50	N/A
	Rating	0.40	0.50	0.50	0.80	N/A
	Weighted Rating	0.08	0.10	0.05	0.40	0.16
Arduino Temperature Sensor Module (*Arduino Temperature Sensor Module | Catch.com.au*, no date)	Value	6.55	N/A*	-55.00 to 125.00	+/- 0.50	N/A
	Rating	0.20	0.00	0.50	0.80	N/A
	Weighted Rating	0.04	0.00	0.05	0.40	0.12
Keyes Temperature Humidity	Value	2.68	5.00	0.00 to 60.00	+/- 2.00	N/A

Sensor Name		Cost ($AUD)	Operating Voltage (V)	Operating Range (°C)	Accuracy (°C)	Average Weighted Rating
	Weighting	**0.20**	**0.20**	**0.10**	**0.50**	**N/A**
Sensor DHT11 For Arduino (*Keyes Temperature Humidity Sensor DHT11 For Arduino*, no date)	Rating	0.90	0.50	1.00	0.10	N/A
	Weighted Rating	0.18	0.10	0.10	0.05	0.11
Arduino Temperature Humidity and Pressure Sensor Module BME280 (*Arduino Temperature Humidity and Pressure Sensor Module BME280 | Catch.com.au*, no date)	Value	14.95	1.71 to 3.60	-40.00 to 85.00	N/A*	N/A
	Rating	0.10	0.70	0.80	0.00	N/A
	Weighted Rating	0.02	0.14	0.08	0.00	0.06
Arduino K Type Thermocouple Temperature Sensor (*Arduino K Type Thermocouple Temperature Sensor*,	Value	10.95	N/A*	N/A*	N/A*	N/A
	Rating	0.10	0.00	0.00	0.00	N/A

Sensor Name		Cost ($AUD)	Operating Voltage (V)	Operating Range (°C)	Accuracy (°C)	Average Weighted Rating
	Weighting	**0.20**	**0.20**	**0.10**	**0.50**	**N/A**
no date)	Weighted Rating	0.02	0.00	0.00	0.00	0.01
Waterproof DS18B20 Temperature Sensor + DS18B20 Adapter Module for Arduino (*DS18B20 Temperature Sensor + DS18B20 Adapter Module for Arduino*, no date)	Value	4.66	3.20 to 5.25	-55.00 to 110.00	+/- 0.50	N/A
	Rating	0.30	1.00	0.70	0.80	N/A
	Weighted Rating	0.06	0.20	0.07	0.40	0.18
Analogue Temperature Sensor for Arduino ARD2-2213 (*Analogue Temperature Sensor For Arduino ARD2-2213 | eBay*, no date)	Value	3.45	N/A*	-55.00 to 125.00	+/- 5.00	N/A
	Rating	0.60	0.00	0.50	0.00	N/A
	Weighted Rating	0.12	0.00	0.05	0.00	0.04

Sensor Name		Cost ($AUD)	Operating Voltage (V)	Operating Range (°C)	Accuracy (°C)	Average Weighted Rating
	Weighting	**0.20**	**0.20**	**0.10**	**0.50**	**N/A**
DS18B20 Temperature Sensor for Arduino (*DS18B20 Temperature Sensor for Arduino Lovers*, no date)	Value	7.06	3.00 to 5.00	-40.00 to 125.00	N/A*	N/A
	Rating	0.20	1.00	0.60	0.00	N/A
	Weighted Rating	0.04	0.20	0.06	0.00	0.08
DS18B20 DC 5V Digital Temperature Sensor Module For Arduino (*DS18B20 DC 5V Digital Temperature Sensor Module For Arduino*, no date)	Value	3.14	5.00	-55.00 to 125.00	+/- 0.50	N/A
	Rating	0.80	0.50	0.50	0.80	N/A
	Weighted Rating	0.16	0.10	0.05	0.40	0.18
Adafruit MCP9808 High Accuracy	Value	8.50	2.70 to 5.50	-40.00 to 125.00	+/- 0.25	N/A

Sensor Name		Cost ($AUD)	Operating Voltage (V)	Operating Range (°C)	Accuracy (°C)	Average Weighted Rating
	Weighting	**0.20**	**0.20**	**0.10**	**0.50**	**N/A**
I2C Temperature Sensor Breakout Board (*MCP9808 High Accuracy I2C Temperature Sensor Breakout Board Australia*, no date)	Rating	0.20	1.00	0.60	1.00	N/A
	Weighted Rating	0.04	0.20	0.06	0.50	0.20
High Accuracy LM75A I2C Temperature Sensor Module for Arduino (*High Accuracy LM75A I2C Temperature Sensor Module for Arduino*, no date)	Value	3.30	2.80 to 5.50	-55.00 to 125.00	+/- 2.00	N/A
	Rating	0.70	1.00	0.50	0.30	N/A
	Weighted Rating	0.14	0.20	0.05	0.15	0.14
Keyestudio Analog Temperature Sensor Compatible with Arduino (*Keyestudio Analog Temperature Sensor*	Value	3.62	5.00	N/A*	N/A*	N/A
	Rating	0.50	0.50	0.00	0.00	N/A

Sensor Name		Cost ($AUD)	Operating Voltage (V)	Operating Range (°C)	Accuracy (°C)	Average Weighted Rating
	Weighting	**0.20**	**0.20**	**0.10**	**0.50**	**N/A**
Compatible with Arduino, no date)	Weighted Rating	0.10	0.10	0.00	0.00	0.05
Temperature Sensor TMP36 (*Temperature Sensor - TMP36 Australia*, no date)	Value	1.50	2.70 to 5.50	-40.00 to 125.00	+/- 2.00	N/A
	Rating	1.00	1.00	0.60	0.30	N/A
	Weighted Rating	0.20	0.20	0.06	0.15	0.60
AM2320 Digital Temperature and Humidity Sensor (*AM2320 Digital Temperature and Humidity Sensor Australia*, no date)	Value	7.52	5.00	-40.00 to 80.00	+/- 0.50	N/A
	Rating	0.20	0.50	0.80	0.80	N/A
	Weighted Rating	0.04	0.10	0.08	0.40	0.16

Sensor Name		Cost ($AUD)	Operating Voltage (V)	Operating Range (°C)	Accuracy (°C)	Average Weighted Rating
	Weighting	**0.20**	**0.20**	**0.10**	**0.50**	**N/A**
DHT22 Temperature and Relative Humidity Sensor Module (*DHT22 Temperature and Relative Humidity Sensor Module Australia*, no date)	Value	13.18	3.30 to 6.00	-40.00 to 80.00	+/- 0.50	N/A
	Rating	0.10	1.00	0.80	0.80	N/A
	Weighted Rating	0.02	0.20	0.08	0.40	0.18

Table 3-11: Decision matrix comparing commercially available temperature sensors.

As can be seen in Table 3-11 above, the Adafruit MCP9808 High Accuracy I2C Temperature Sensor Breakout Board was selected as it was the ideal compromise between cost, operating voltage, operating range, and accuracy.

4.4.2 INTER-INTEGRATED CIRCUIT (I2C) PROTOCOL

MCP9808 Temperature Sensors use the I2C protocol. Each I2C bus consists of a SCL, or clock, signal and a SDA, or data, signal as shown in Figure 3-8.

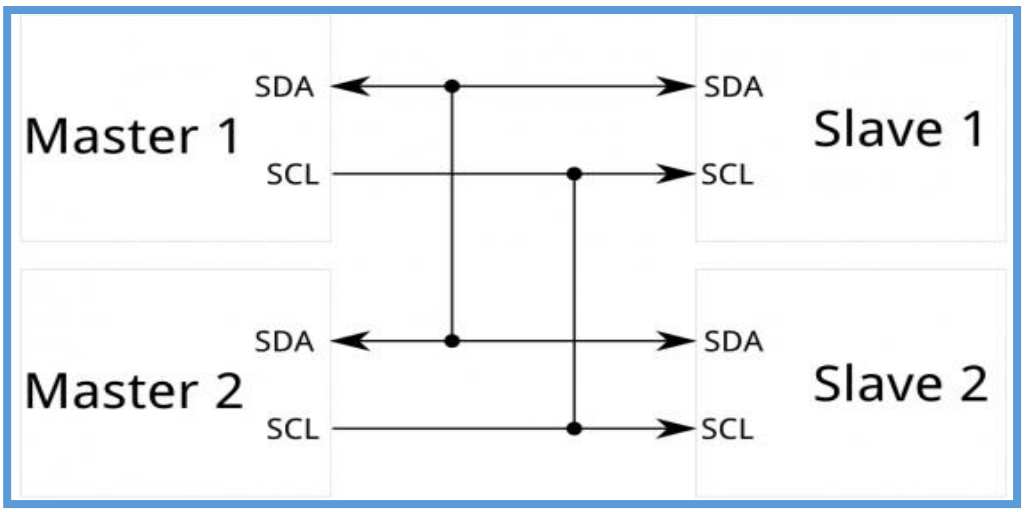


Figure 3-8: Data (SDA) and clock (SCL) pins for a number of slaves and master devices (*Basics of the I2C Communication Protocol*, no date).

If the master (microcontroller) is to request data from one particular temperature sensor, the master sends the address of the desired sensor to ALL temperature sensors. Each temperature sensor then compares this address with their own address and data is transmitted only for the one temperature sensor whose address matches.

This is depicted by Figure 3-9, where the Master sends an address of 0101101 to all of the slaves. Slave 1 has an assigned address of 0110000, so it checks this against the address sent by the master, and there is not a match. Slave 2 has an address of 1001000 and, when compared to the requested address of 0101101, there is no match. Lastly, Slave 3 has the same address as that requested by the master, so transmission occurs and only this slave is able to send data on the SDA line. This sensor then sends a low voltage ACK bit, indicating that the sensor value should be acknowledged by the microcontroller.

(Master) Alternately, the slaves that do not match (temperature sensors with different addresses), do nothing.

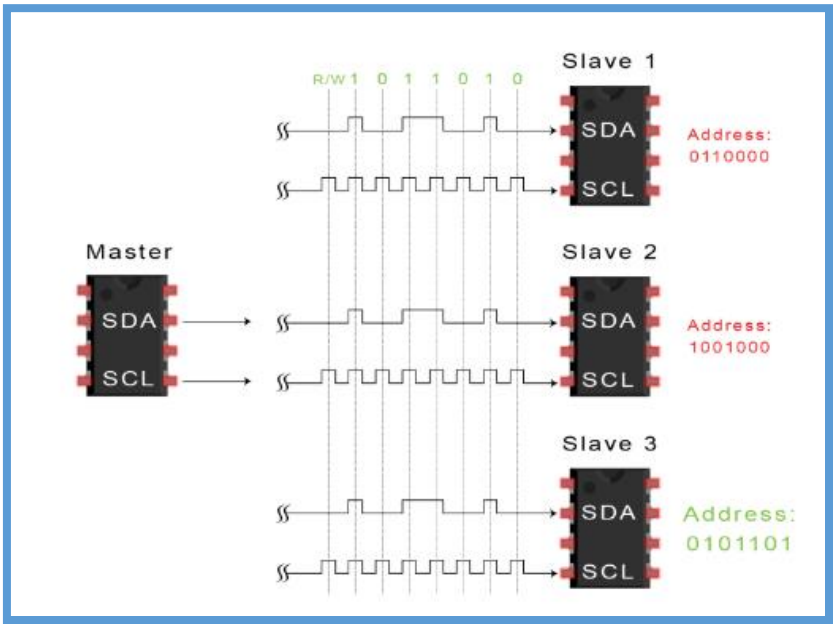


Figure 3-9: Assigning addresses for one master device, and multiple slave devices (*Basics of the I2C Communication Protocol*, no date).

In terms of the specific temperature sensor used in the thesis, the MCP9808 board has two I2C Data pins. The SCL pin is the clock pin, and the SDA pin is the data pin. There is also an optional alert pin, which can be programmed to alert the user when the temperature exceeds a set value, or falls below a set value. A pull-up resistor is required to use this pin. Lastly, there are three address pins (A0, A1 and A2) which can be used to adjust the address in order to ensure that the five temperature sensors could use the same SDA and SCL pins. The addresses used to identify each of the temperature sensors were designated using a binary system where a pin was 'ON' if it was supplied with 5V, and 'OFF' if it was connected to GND. The addresses of each temperature sensor have been shown in Table 3-12.

Temperature Sensor	Binary[20] Address	Hexadecimal[21] Address	Corresponding Pin Connections
TS0	000	0x18	A0 to GND A1 to GND A2 to GND
TS1	001	0x19	A0 to GND A1 to GND A2 to 5V
TS2	010	0x1A	A0 to GND A1 to 5V A2 to GND

[20] The binary system is a method of representing numbers using only two digits (1 and 0) (*Binary Definition*, no date). This is known as a base-2 system.

[21] The hexadecimal system is a method of representing numbers using the digits 0 to 9 followed by the letters A to F (*What is hexadecimal?*, no date). This is known as a base-16 system.

Temperature Sensor	Binary[20] Address	Hexadecimal[21] Address	Corresponding Pin Connections
TS3	011	0x1B	A0 to GND A1 to 5V A2 to 5V
TS4	100	0x1C	A0 to 5V A1 to GND A2 to GND

Table 3-12: Address designation for each temperature sensor.

It is important to note that the binary addresses shown in Table 3-12 do not correspond directly to the subsequent hexadecimal addresses listed. This is due to the fact that the default address for a MCP9808 sensor is 0x18. As a result, the value of the binary address is then added to this base value. For example, temperature sensor 4 has a binary address of 100, which is equal to 4. This is added to the base of 18 which would become 22. As the addresses are defined in the hexadecimal system, this is equal to 0x1C.

The assignment of these addresses to each temperature sensor is depicted visually in Figure 3-10.

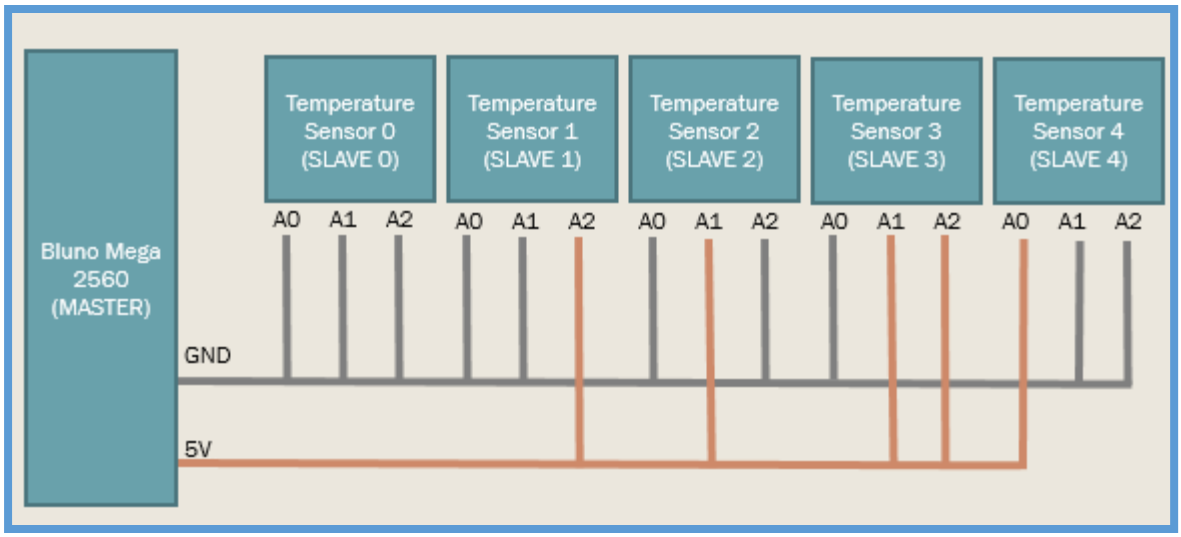


Figure 3-10: Visual depiction of assigned addresses to each of the five temperature sensors.

In terms of temperature sensor assembly, the header strip was prepared first, as it needed to be cut to the desired length. This header strip was inserted into a breadboard with the long pins facing downwards in order to simplify the soldering process. The breakout board was then placed on top of the header strip, so that the short ends of the header strip poked through the board. The board could then be soldered onto this header strip, and the soldering joints were inspected visually to ensure adequate contact. The sensor could then connect to the microcontroller, as shown in Figure 3-11.

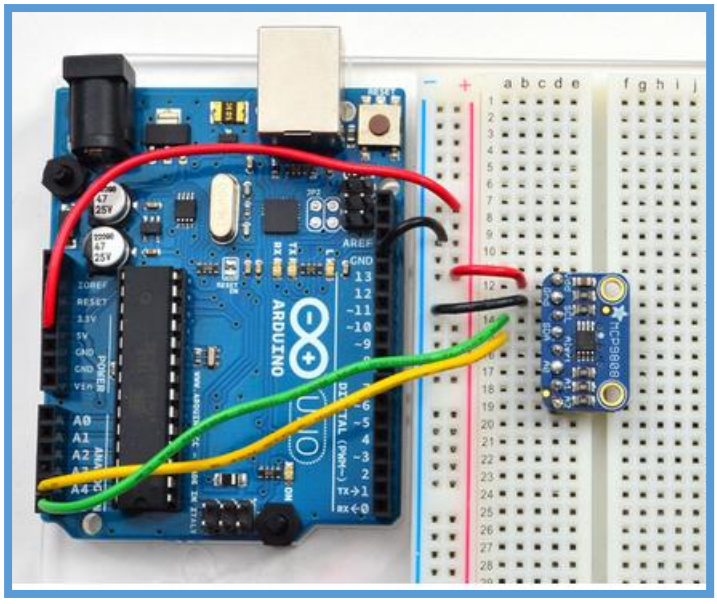


Figure 3-11: Wires connecting temperature sensor to an Arduino UNO (*Pinouts | Adafruit MCP9808 Precision I2C Temperature Sensor Guide | Adafruit Learning System*, no date).

The connections made in Figure 3-11 between the temperature sensor pins and the microcontroller ports have been outlined in Table 3-13.

Microcontroller Port	Temperature Sensor Pin
5V Power Supply	VDD
GND	GND
I2C Clock SCL (Digital 21)	SCL
I2C Data SDA (Digital 20)	SDA

Table 3-13: Hardware connections between the microcontroller and temperature sensor.

4.4.3 TEMPERATURE SENSOR RESPONSE

The output of the temperature sensor is a sampled data signal. That is, a continuous time signal is sampled at discrete time intervals. An example of this type of signal is shown in Figure 3-12.

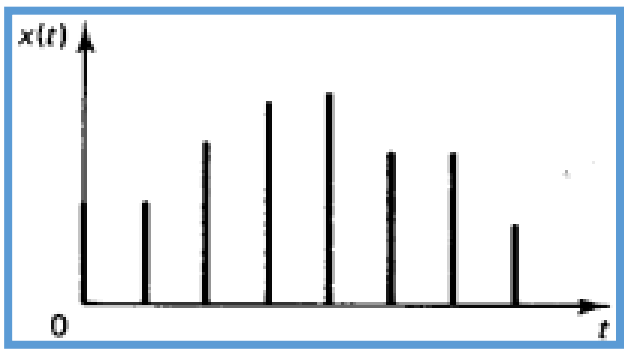


Figure 3-12: Example of a sampled data signal, such as that which is the output of the selected temperature sensor (He, 2017a).

These time intervals are specified by the sampling rate of the particular sensor and, in the case of the MCP9808 High Accuracy I2C Temperature Sensor Breakout Board, this sampling rate is outlined in Table 3-14 for a range of user-defined resolution values.

Resolution (°C)	Samples/sec (Typical)
0.5	33
0.25	15
0.125	7
0.0625	4

Table 3-14: Number of samples per second as a function of sensor resolution ('Maximum Accuracy Digital Temperature Sensor', no date).

4.4.4 TEMPERATURE SENSOR CODE

In order to control the MCP9808 Temperature Sensor, the Adafruit_MCP9808 library was downloaded in the Arduino IDE. This library enabled access to the code required to interface with the sensor, and this code has been included in Appendix A (Individual Temperature Sensor Code) for reference.

4.4.5 TESTING TEMPERATURE SENSOR INDEPENDENTLY

In order to ensure that the selected temperature sensor was suitable for the neonatal warming blanket design, it was necessary to first test this sensor as an isolated component in order to analyse its functionality, accuracy and precision.

The accuracy of this sensor was verified by comparing the results with a reference sensor (Digital DT-625 Thermometer and Humidity Meter). The values from these sensors were also compared with the sensor originally selected for humidity monitoring (DHT22 sensor). The results from this initial testing have been summarised in Table 3-15.

Conditions	Data Source	Trial 1 (°C)	Trial 2 (°C)	Trial 3 (°C)	Average (°C)	Range (°C)
Room temperature	MCP9808 Temperature Sensor	19.19	19.31	19.37	19.29	0.18
	DHT22 Temperature and Humidity Sensor	25.20	25.30	26.70	25.73	1.50
	DT-625 Thermometer and Humidity Meter	18.80	18.90	18.90	18.87	0.10
Axillary temperature	MCP9808 Temperature Sensor	35.13	35.19	35.19	35.17	0.06
	DHT22 Temperature and Humidity Sensor	35.90	35.80	35.90	35.87	0.10
	DT-625 Thermometer and Humidity Meter	35.40	35.40	35.50	35.43	0.10

Table 3-15: Results obtained from independent testing of the MCP9808 temperature sensor, when compared to a digital DT-625 Thermometer and Humidity Meter. The outputs of the Arduino temperature sensor are depicted by the top row of each condition, and digital thermometer on the bottom row.

Figure 3-13 depicts the results from testing with these three sensors at room temperature, with the corresponding range in values depicted by the bar.

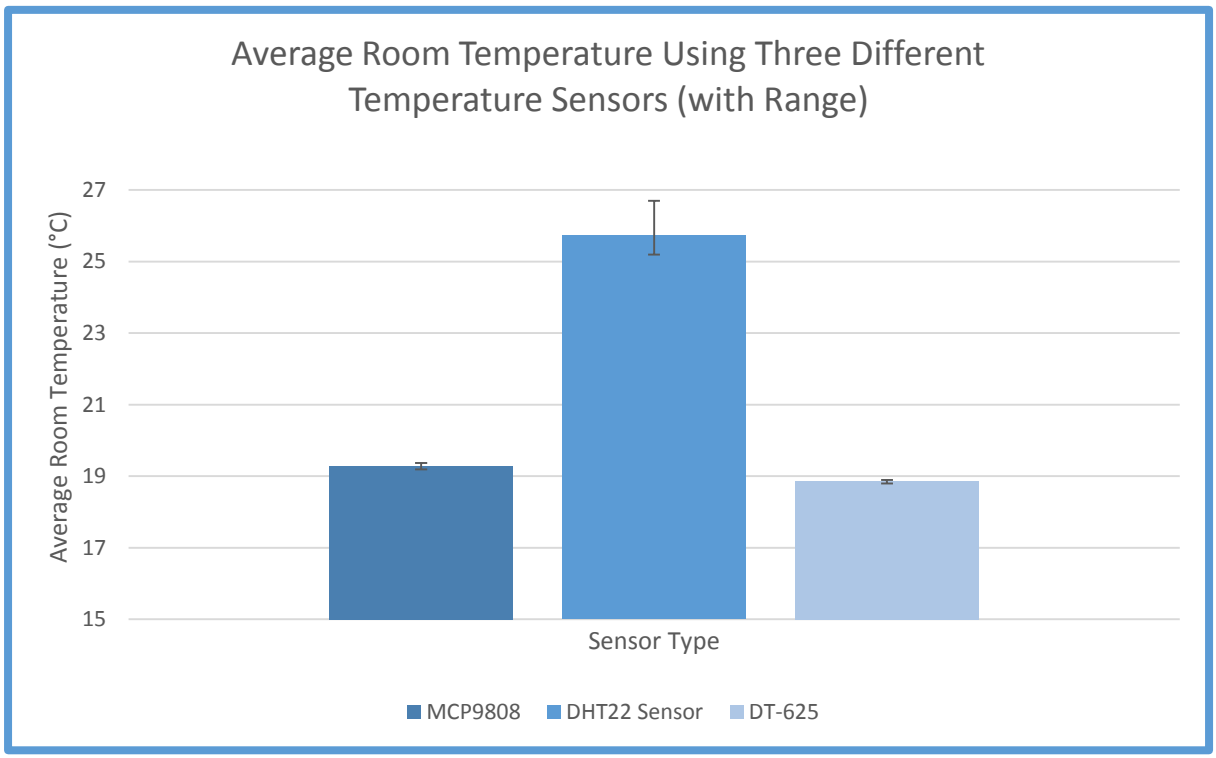


Figure 3-13: Comparison of temperature measurements using three different temperature sensors, in room temperature conditions. The bar on each set of data represents the range of the three values obtained during testing.

From Figure 3-13, it can be seen that the DHT22 sensor does not produce temperature values of a high correlation with the other two sensors in room temperature conditions. For this reason, this sensor will not be analysed further under these conditions.

However, there is only a 0.42°C difference between the values produced by the MCP9808 sensor, and the hospital-grade DT-625 sensor. In order to determine whether this is a statistical significance, a paired sample t-test[22] was conducted in Excel, as shown in Table 3-16.

[22] A t-test is a statistical test used to "determine whether the mean difference between two sets of observations is zero" (*Paired Sample T-Test*, no date). A paired sample t-test involves measuring each subject twice.

	MCP9808 Sensor	DT-625 Sensor
Mean	19.29	18.86667
Variance	0.0084	0.003333
Observations	3	3
Pearson Correlation	0.944911	
Hypothesized Mean Difference	0	
Df	2	
t Stat	17.61173	
P(T<=t) one-tail	0.001604	
t Critical one-tail	2.919986	
P(T<=t) two-tail	0.003208	
t Critical two-tail	4.302653	

Table 3-16: Paired sample t-test results using the temperature sensors at room temperature. The significance of these results for a two-tailed test has been highlighted in yellow.

The most significant piece of data from Table 3-16 is the two-tailed test data (*FAQ: What are the differences between one-tailed and two-tailed tests?*, no date), highlighted in yellow above. A two-tailed test considers the significance of values above and below the mean. Therefore, when considering a significance level of 0.05, the test will assume a level of 0.025 above and 0.025 below.

From Table 3-16, the P value[23] for the two tailed test is 0.003. As this is less than 0.05, this indicates a statistical significance between the results using the MCP9808 sensor and the results using the DT-625 sensor at room temperature.

In order to determine the percentage error, the following equation is used:

$$Percentage\ Error\ (\%) = \frac{Observed\ Value - Theoretical\ Value}{Theoretical\ Value} * 100$$

[23] A P value is "the level of marginal significance within a statistical hypothesis test representing the probability of the occurrence of a given event" (*P-Value*, no date).

Therefore, at room temperature, the percentage error of the DHT22 sensor can be determined.

$$Percentage\ Error\ (\%) = \frac{25.73 - 18.87}{18.87} * 100 = 36.35\%$$

As determined previously, this percentage error is too high to justify its inclusion in the design. Similarly, the percentage error of the MCP9808 sensor at room temperature can be deduced.

$$Percentage\ Error\ (\%) = \frac{19.29 - 18.87}{18.87} * 100 = 2.23\%$$

This percentage error is much lower than the DHT22 sensor, thus confirming its suitability for the design. The MCP9808 sensor also had a high level of correlation with the DT-625 sensor when skin temperature was measured under the arm, as shown in Figure 3-14.

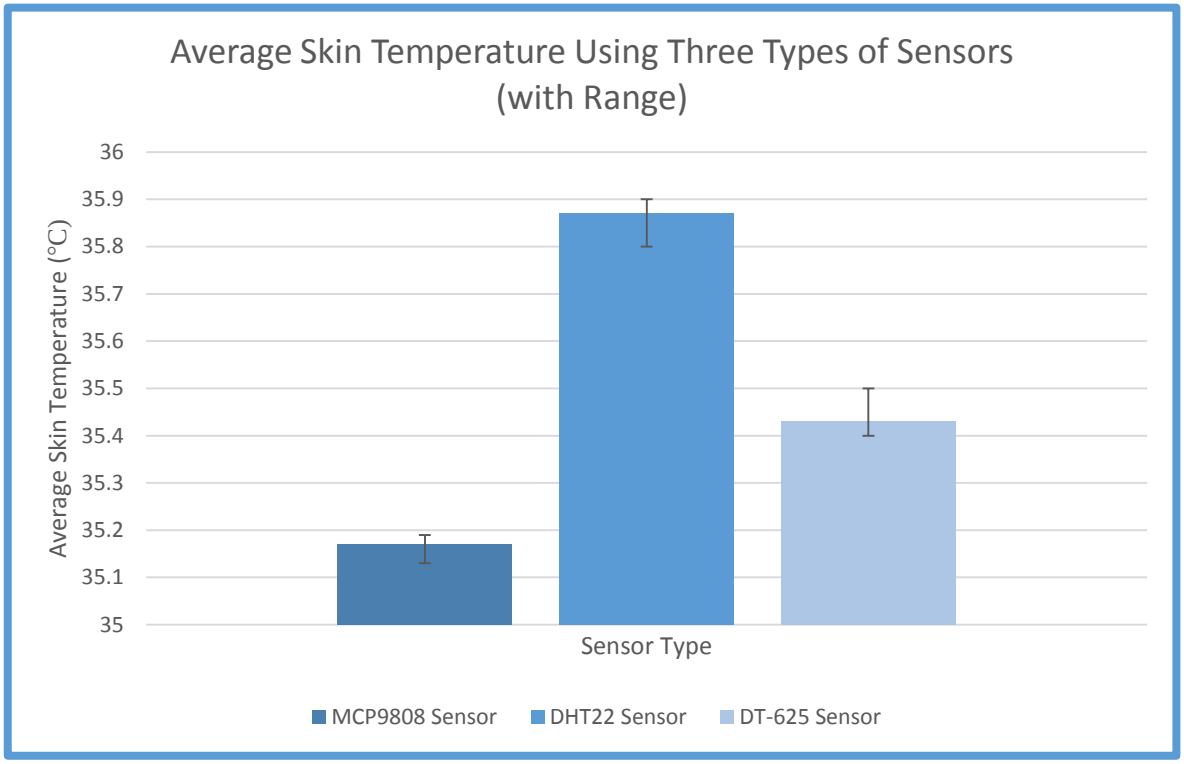


Figure 3-14: Average skin temperature for the three temperature sensors. The bars shown on each set of data represent the range of data obtained during the three tests.

The MCP9808 sensor differed from the reference values provided by the DT-625 sensor by only 0.06°C, and the DHT22 sensor differed from the DT-625 sensor by 0.44°C. Therefore, the statistical significance was determined based on the MCP9808 and DT-625 sensors only. The results of the paired sample t-test are highlighted in Table 3-17.

	MCP9808 Sensor	DT-625 Sensor

	MCP9808 Sensor	DT-625 Sensor
Mean	35.17	35.43333
Variance	0.0012	0.003333
Observations	3	3
Pearson Correlation	0.5	
Hypothesized Mean Difference	0	
df	2	
t Stat	-9.06192	
P(T<=t) one-tail	0.00598	
t Critical one-tail	2.919986	
P(T<=t) two-tail	0.01196	
t Critical two-tail	4.302653	

Table 3-17: Results from the standard t-test using the temperature sensors at skin temperature. The significance value for a two-tailed test has been highlighted in yellow.

From Table 3-17, the P value of the paired sample t-test for a two tailed test was approximately 0.012. As this is less than 0.05, there is statistical significance between these two sensors at skin temperature. Furthermore, at skin temperature, the percentage error of the DHT22 sensor can be calculated.

$$Percentage\ Error\ (\%) = \frac{35.87 - 35.43}{35.43} * 100 = 1.24\%$$

Similarly, the percentage error of the MCP9808 sensor can be calculated at skin temperature.

$$Percentage\ Error\ (\%) = \frac{35.37 - 35.43}{35.43} * 100 = 0.17\%$$

It is the accuracy of the sensor when measuring body temperature that is crucial to the success of the product, rather than the accuracy of the sensor when measuring room temperature. Once again, the DHT22 sensor had a much higher percentage error than the MCP9808 sensor, and so it was not to be used for temperature measurement under these conditions. The MCP9808 sensor, on the other hand, has been found to be appropriate for the current design due to the prototype nature of the project but a more accurate sensor should be selected if the budget were to be

increased in the future. This would ensure that there was no significant difference between the values output by the chosen and reference temperature sensors in these conditions.

4.5 TEMPERATURE CONTROL

The following section of this thesis details the temperature control system incorporated into the designed neonatal warming blanket. This temperature control system consists of heating elements and an algorithm responsible for altering the output of these heating elements based on the temperature of the neonate.

4.5.1 HEATING ELEMENT SELECTION

Five separate heating elements were used to warm the neonate, with dimensions 5 x 15 cm. One of these heating elements is depicted in Figure 3-15. These were chosen due to their flexibility, which was a known issue with elements used in previous warming blanket designs. This flexibility enables the blanket to wrap around the neonate without damaging the heating elements.

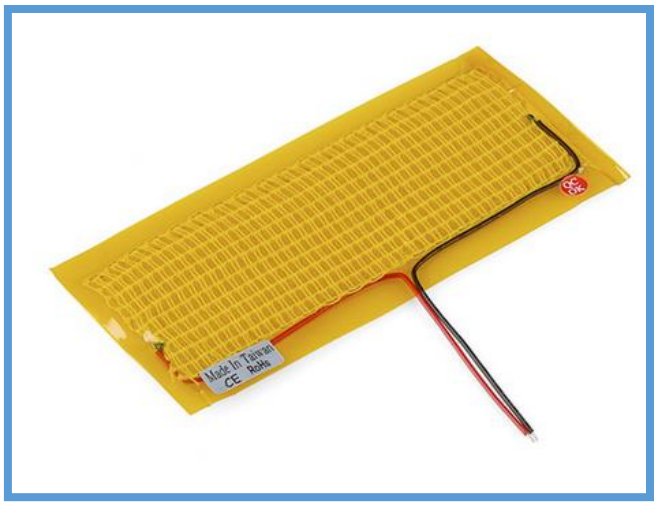


Figure 3-15: Heating pad with dimensions 5 x 15 cm (*Heating Pad - 5x10cm - SparkFun COM-11288: SparkFun Australia*, no date).

The warming profile of these heating pads is shown in Figure 3-16.

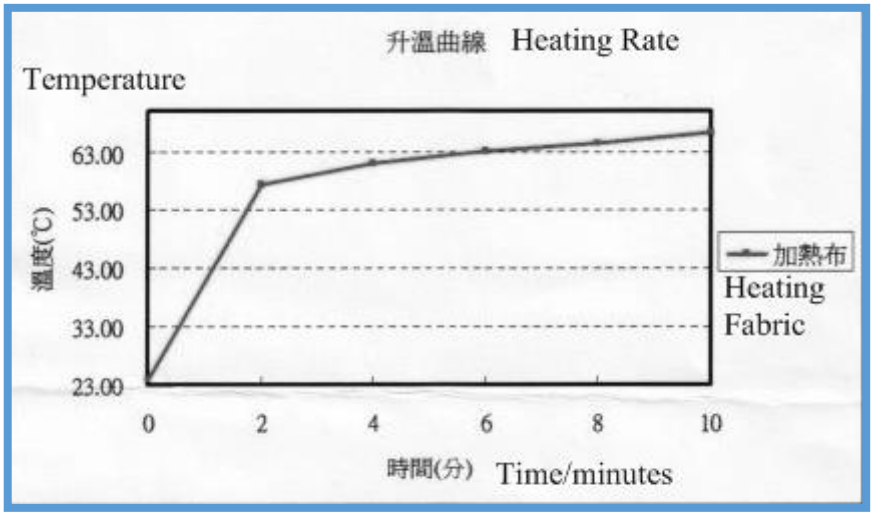


Figure 3-16: Warming profile of the selected heating pads (*Heating Pad - 5x10cm - SparkFun COM-11288: SparkFun Australia*, no date).

As can be seen from Figure 3-16, the time taken for these heating pads to read a maximum temperature is 10 minutes. This time is comparable to the warm-up time of other commercially available warming blankets, and therefore increases the likelihood of adoption by the Flinders Medical Centre.

4.5.2 HEATING ELEMENT ASSEMBLY

A voltage of 5V was applied to the heating pads in order to increase their temperature, and the operating current equated to around 600mA, with 8.3Ω nominal resistance.

Power Metal Oxide Semiconductor Field Effect Transistors (MOSFETs) were also included in the assembly. A power MOSFET is essentially a low power switch, which are typically characterized by a vertical assembly with the Source and Drain on opposite sides ('Power MOSFET Basics', no date). There are numerous types of MOSFETs as outlined in Table 3-18.

MOSFET Type	V_{GS} (+ve)	V_{GS} (0V)	V_{GS} (-ve)
N-channel Enhancement	ON	OFF	OFF
N-channel Depletion	ON	ON	OFF
P-channel Enhancement	OFF	OFF	ON
P-channel Depletion	OFF	ON	ON

Table 3-18: Effect of positive, negative, or zero voltage on MOSFET channels ('Power MOSFET Basics', no date).

For this project, an N-Channel Enhancement MOSFET was selected. This type of MOSFET has pins shown in Figure 3-17.

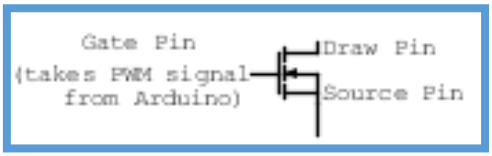


Figure 3-17: Depiction of 3-pin N-Channel Enhancement MOSFET using the FidoCadJ software.

As shown in Table 3-18, a positive voltage (V_{GS}) would turn this type of MOSFET 'ON". This means that a current would pass through, and, as a result, the temperature of the heating elements would increase. If the voltage (V_{GS}) was equal to 0V, there would be no difference between the source voltage and the gate voltage, and so the MOSFET would not allow a current to pass through. Lastly, if the voltage (V_{GS}) was negative, the source voltage would become greater than the gate voltage, meaning that the gate voltage was not high enough to cross the threshold. Therefore, no current would pass through under these circumstances, either. In other words, current would only pass through the MOSFET when a positive voltage was applied. This current would turn the heating elements on.

It is important to consider the cause of this voltage, or lack thereof. As shown in Figure 3-17, the signal sent to the gate pin is from a Pulse Width Modulation, or PWM, pin on the main microcontroller. This PWM pin outputs a signal with a particular duty cycle depending on the output of an algorithm that will be discussed in subsequent sections of this thesis. This relationship is highlighted in Figure 3-18.

Figure 3-18: Process required to turn the heating elements on and off.

The specifications of the selected MOSFET are outlined in Table 3-19, with an image of the selected surface-mount MOSFET shown in Figure 3-19.

Metric	Value
Transistor Polarity	N Channel
Transistor Case Style	SOT-223
Continuous Drain Current	4A
Drain Source Voltage	60V
Threshold Voltage	1.6V
Number of Pins	4 Pins
Maximum Operating Temperature	150°C

Table 3-19: Specifications for the selected surface mount MOSFET (NDT3055L) (*NDT3055L - MOSFET Transistor, N Channel, 4 A, 60 V, 70 mohm, 10 V, 1.6 V*, no date)

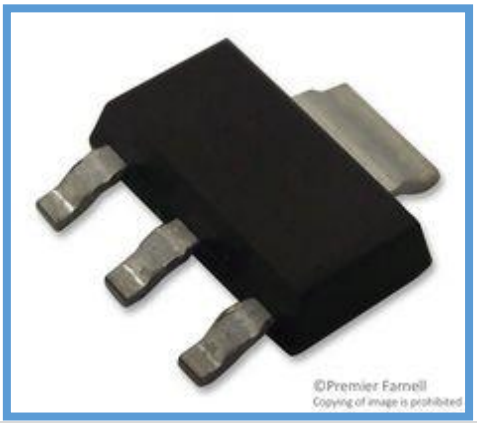


Figure 3-19: The selected 4-pin N-Channel MOSFET (*NDT3055L - MOSFET Transistor, N Channel, 4 A, 60 V, 70 mohm, 10 V, 1.6 V*, no date).

The overall structure of the heating element assembly, including the heating elements and MOSFETs, is highlighted in Figure 3-20.

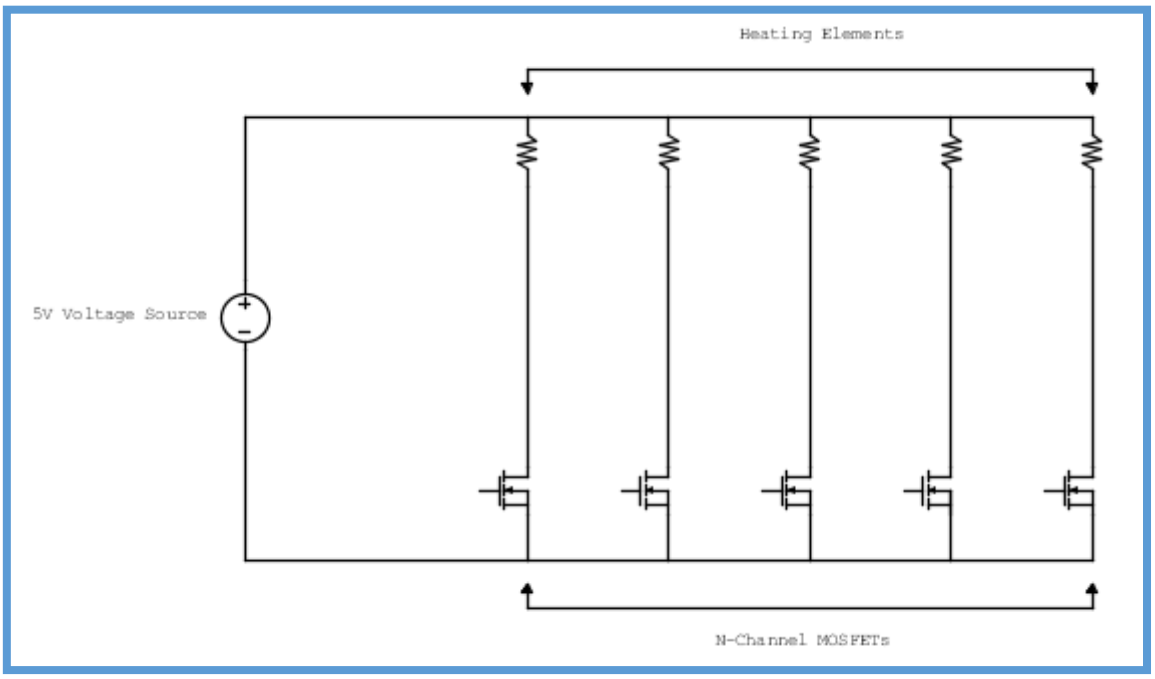


Figure 3-20: Circuit diagram depicting set-out of the 5V adapter, the five heating elements, and the N-Channel MOSFETs. This diagram was drawn using the FidoCadJ software.

The incorporation of five separate MOSFETs in the design was significant as this feature would enable the individual control of temperature through each of the five separate heating elements in future studies. For example, if the temperature of the neonate's head region dropped, the heat could be adjusted in this region alone while maintaining the temperature at the rest of the neonate's body.

4.5.3 HEAT CONTROL ALGORITHM

This section details the myriad of heat control algorithms commonly used in industry. These were compared in order to determine the most appropriate algorithm for the current design. Table 3-20 has been included to provide a brief summary of these algorithms, in terms of the advantages and disadvantages of each.

Algorithm	Description	Advantages	Disadvantages
Proportional Integral Derivative (PID)	Proportional component relates to present data Integral component relates to past data Derivative component relates to future data	Commonly used in industry Improved response as number of temperature sensors increases (*Advanced Temperature Control Strategies for Thermal Testing*, no date) Fast and accurate response	Requires specific positioning of the temperature sensor

Algorithm	Description	Advantages	Disadvantages
Take-Back-Half (TBH)	When actual temperature equals desired temperature, a new value ([H+Ho]/2) replaces two variables in the system (namely H and Ho) (Woodward, 2005) Comparable to "the I portion of a PID controller" but "every time that the error crosses zero, the output is cut" (*Take-Back Half*, no date)	Robust "Convergence is assured if this system period is at least eight times longer than the heater-sensor time delay" (Woodward, 2000) Zero steady state error[24] "Only one gain constant to tune" (*Take-Back Half*, no date) Fast response time "Decent stability except with exceedingly high gain" (*Take-Back Half*, no date)	"Can only be used in velocity control applications" (*Take-Back Half*, no date)

[24] Steady state error is the "difference between the input (command) and the output of a system in the limit as time goes to infinity (i.e. when the response has reached steady state)" (*Control Tutorials for MATLAB and Simulink - Extras: Steady-State Error*, no date).

Algorithm	Description	Advantages	Disadvantages
Straight Integration	Measures actual temperature, and subtracts it from the desired temperature Each time temperature is sampled, the loop gain multiplies the difference and "adds it as a cumulative adjustment to the heater-power setting" (Woodward, 2005)	Simple algorithm, and easy to implement Zero steady-state error	Continuous oscillation Significant overshoot
Valve Motor Drive (VMD)	(*Temperature Controller Basics Handbook | Instrumart*, no date)	Fast response Simple Variable valve position	Irrelevant in the given context

Algorithm	Description	Advantages	Disadvantages
Profiling or Ramp-soak	Ramp is the change in set point Time to sit at each set point is the soak or dwell One ramp, or one soak, is equal to one segment The segments can be altered to provide temperature control (*Temperature Controller Basics Handbook | Instrumart*, no date)	Can be derived from PID algorithm (*Converting PID controller to Ramp Soak controller*, no date)	

Algorithm	Description	Advantages	Disadvantages
Multi-loop	"Once a control structure is fixed, control performance is then determined mainly by tuning each multiple single-loop PID controller" (Lee *et al.*, 2004) Various novel methods of tuning multiple single-loop PID controllers are presented in literature	Accept multiple inputs and outputs Compact Can operate as standalone system Faster scanning rate Absence of buttons, and therefore increased security (*Temperature Controller Basics Handbook | Instrumart*, no date)	Rarely coupled with user interfaces

Table 3-20: Summary of various temperature regulation and control algorithms.

The PID algorithm was selected as this algorithm is commonly used in industry for temperature regulation applications. The student also had experience with this type of algorithm, making the coding and debugging processes more efficient.

This algorithm has three parts: a proportional, integral and derivative coefficient. It involves reading data from a sensor, and responding to this data by actuating a change in the system. The actuated signal is calculated using the three coefficients mentioned earlier. A visual representation of this algorithm is shown in Figure 3-21.

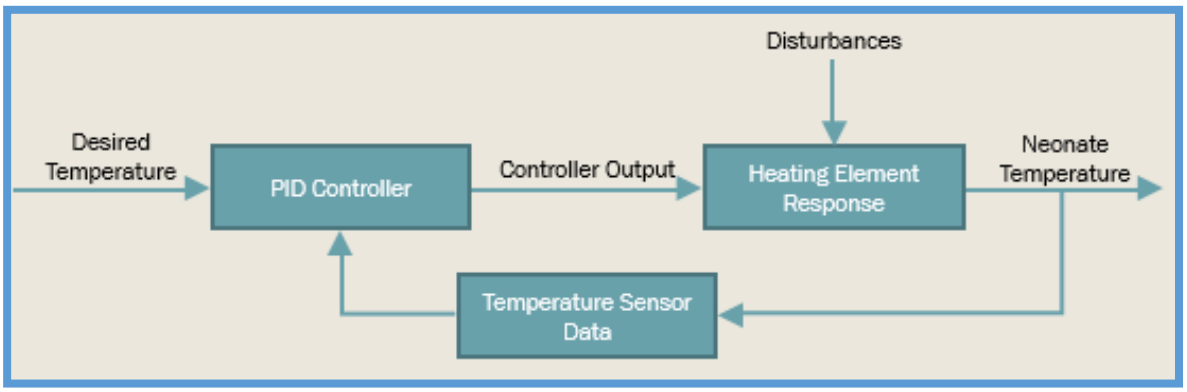


Figure 3-21: Basic overview of PID algorithm. The Setpoint is the desired temperature of the neonate, and the measured process variable is the actual neonate temperature measured by the temperature sensor(s).

As can be deduced from Figure 3-21, the desired temperature was compared to the actual temperature of the neonate (obtained from the temperature sensor data). The difference between these (the error) was calculated and the corresponding controller output set the duty cycle of the PWM signal sent to turn the heating elements 'ON' or 'OFF'.

When the sensor value (actual neonate temperature) was equal to the set point (desired neonate body temperature), the error was zero and the heating element remained at a constant temperature rather than increasing or reducing the heat supplied by the heating elements. This was because the purpose of the heating element in this case was to maintain the temperature as it was already ideal.

However, when there was a difference between the actual temperature and the desired temperature, the error was no longer equal to zero. If the actual temperature was higher than the desired temperature, the duty cycle of the PWM signal was reduced. This meant that the heating element spend more time 'OFF' than 'ON'.

Alternatively, if the actual temperature was lower than the desired temperature, the duty cycle of the PWM signal increased to compensate for this drop in temperature. This meant that the heating elements spent more time 'ON' than 'OFF'.

The PID block in Figure 3-21 requires further analysis, however, in order to gain a comprehensive understanding of the temperature regulation process. Figure 3-22 highlights the calculations performed by this algorithm in order to produce the controller output.

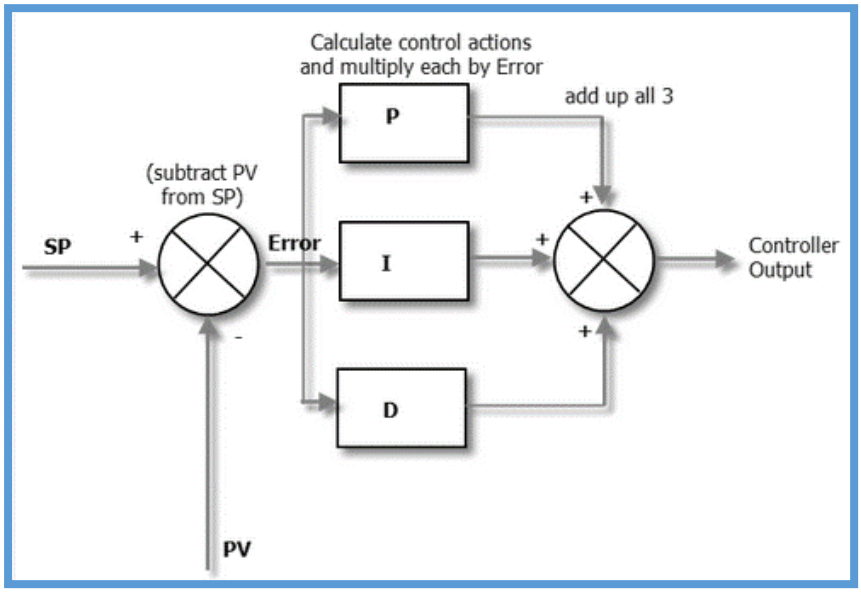


Figure 3-22: Overview of the calculations performed by each portion of the PID algorithm.

The derivative block (abbreviated to 'D' in the diagram above) performs calculations based on the rate of change of temperature. The integral block (abbreviated as 'I' above) performs calculations based on "the sum of all the instantaneous values that the signal has been" and the proportional block (abbreviated as 'P' above) is responsible for the multiplication of the error by the proportional gain.

When refining an algorithm to produce the fastest response time with minimal overshoot, it was important to understand the effect of each of the model parameters. The typical effect of each of the proportional, integral and derivative terms is outlined in Table 3-21, Table 3-22 and Table 3-23, respectively.

Proportional Term (P)	
Rise Time	Decrease
Overshoot	Increase
Settling Time	Small Change
Steady-State Error	Decrease

Table 3-21: Effect of Proportional term on the rise time, overshoot, settling time and steady-state error (*PID Control - National Instruments*, 2012).

Integral Term (I)	
Rise Time	Decrease
Overshoot	Increase
Settling Time	Increase
Steady-State Error	Eliminate

Table 3-22: Effect of Integral term on the rise time, overshoot, settling time and steady-state error (*PID Control - National Instruments*, 2012).

Derivative Term (D)	
Rise Time	Small Change
Overshoot	Decrease
Settling Time	Decrease
Steady-State Error	Small Change

Table 3-23: Effect of Derivative term on the rise time, overshoot, settling time and steady-state error (*PID Control - National Instruments*, 2012).

The trends highlighted in Table 3-21, Table 3-22 and Table 3-23 were used to develop the ideal PID algorithm for the given application.

4.5.4 PID VERIFICATION

An Excel program was used to simulate the PID response to theoretical fluctuations in temperature. The first simulation assumed that the temperature of the neonate was 35°C, and this fell in the 'mild hypothermia' range. As shown in Table 3-24, temperatures in the range corresponded to a PID output of 0.75, or a PWM duty cycle of 75%.

Core Temperature Range (°C)	Condition	PID Output	Duty Cycle (%)
<28.0	Severe Hypothermia	1	100
28.0-32.0	Moderate Hypothermia	0.85	85
32.0-36.5	Mild Hypothermia	0.75	75
36.5-37.5	Normal	0.5	50
37.5-38.8	Mild Hyperthermia	0.25	25
>38.8	Severe Hyperthermia	0	0

Table 3-24: PID output and corresponding duty cycles set by the PWM pin on the Arduino microcontroller.

When the temperature of the neonate was set to 35°C in Excel, various PID tuning parameters were tested to determine the ideal response. The first of these iterations is shown in Figure 3-23, where the

process delay and controller gain were both set to 10, and all other tuning parameters were left at 0. The red function showed the desired temperature of 37°C, while the blue function represented the temperature of the neonate (35°C in Figure 3-23).

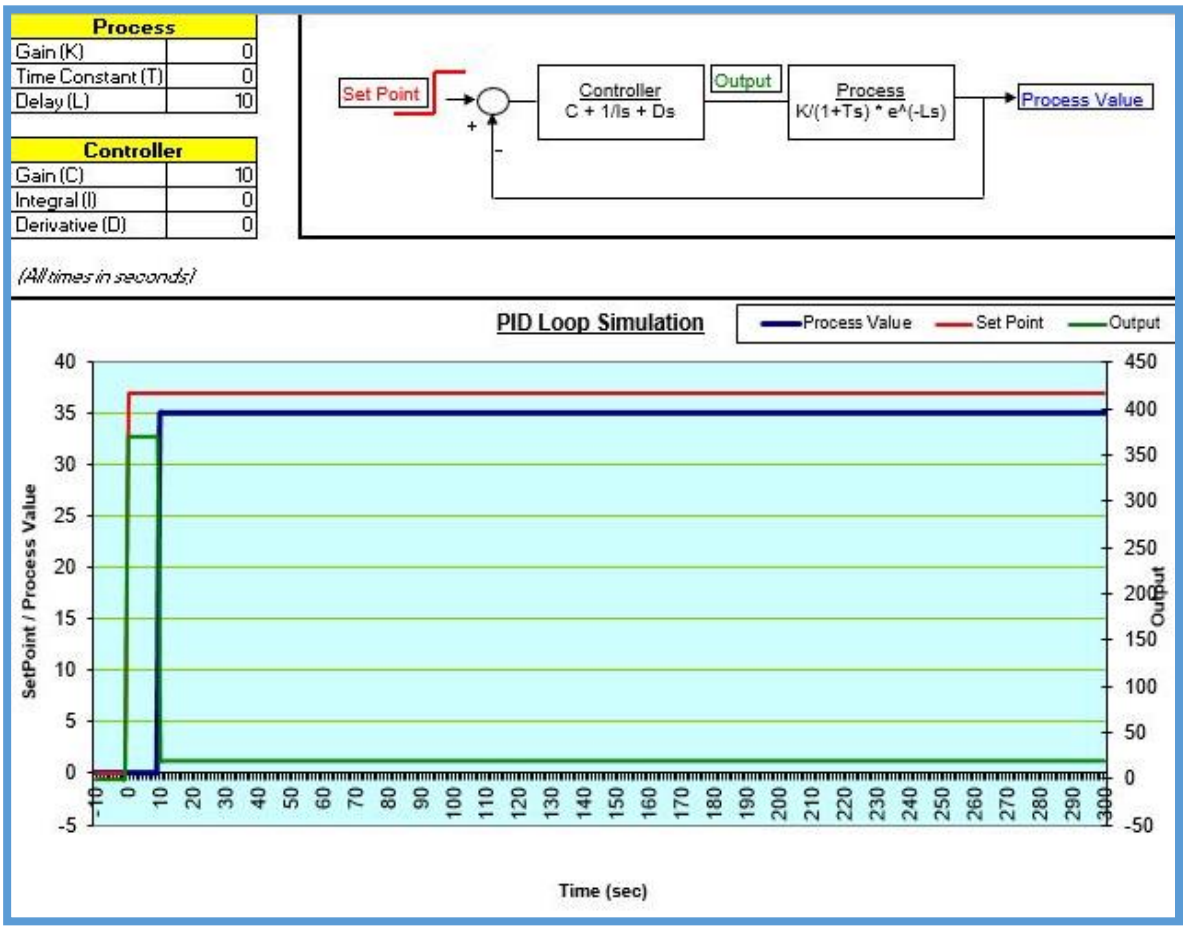


Figure 3-23: Example PID response when the neonate temperature was set to 35°C (*PID Loop Simulator*, no date). Process delay was set to 10, and controller gain was also set to 10.

With the controller gain and process delay both set to 10 in Figure 3-23, the neonate temperature remained at 35°C, so this system did not adequately increase the neonate temperature to the desired 37°C. Therefore, the tuning parameters were altered further to produce the somewhat improved response shown in Figure 3-24.

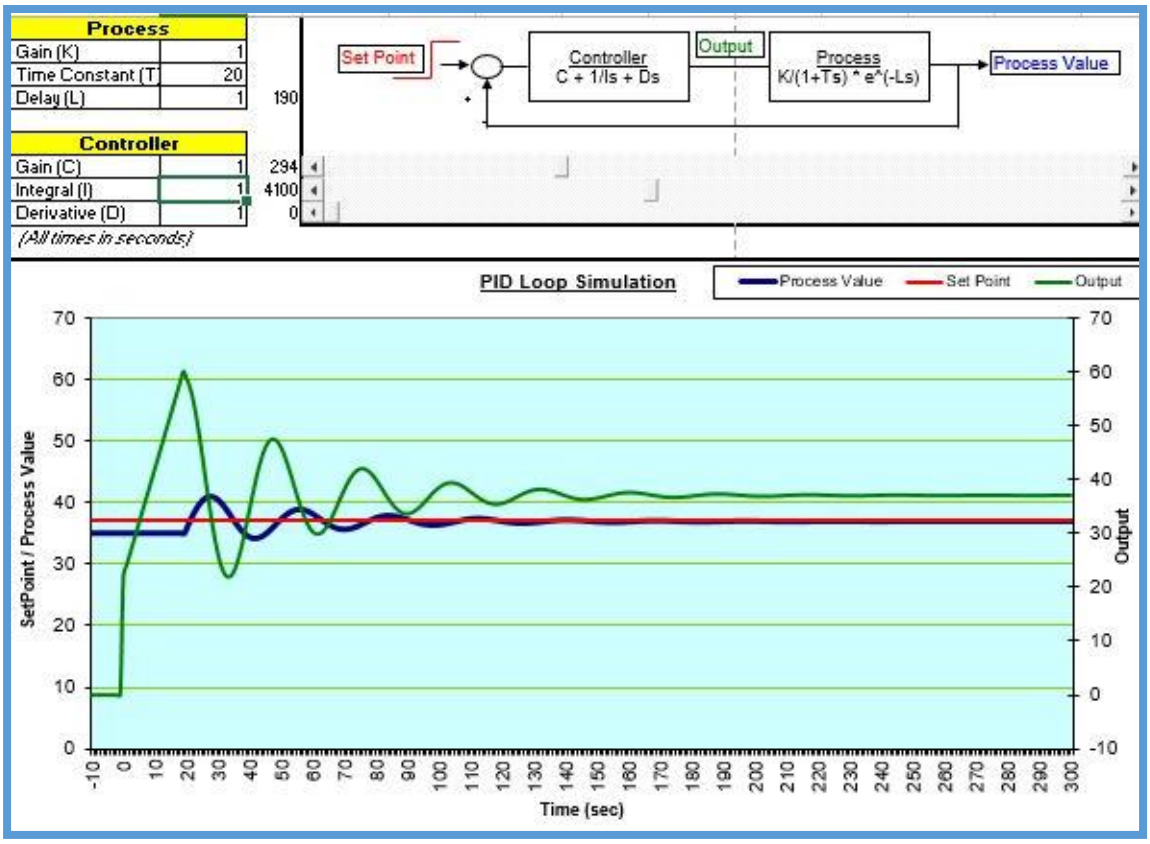


Figure 3-24: Tuned PID response when neonate temperature was set to 35°C (*PID Loop Simulator*, no date). Process gain and delay were set to 1 and the time constant was set to 20. The controller gain, integral and derivative terms were all set to 1.

While the system response shown in Figure 3-24 is an improvement compared to Figure 3-23, oscillations occur before the neonate temperature settles at 37°C. This means that the ideal temperature is not reached until around 160 seconds. During this time, the neonate temperature fluctuates rapidly from 35°C to around 42°C and back again, causing the neonate to become hypo- and hyperthermic repeatedly in a short period of time. These fluctuations could prove fatal and so another PID revision was required. The next system is depicted in Figure 3-25, where the oscillations shown previously were eliminated.

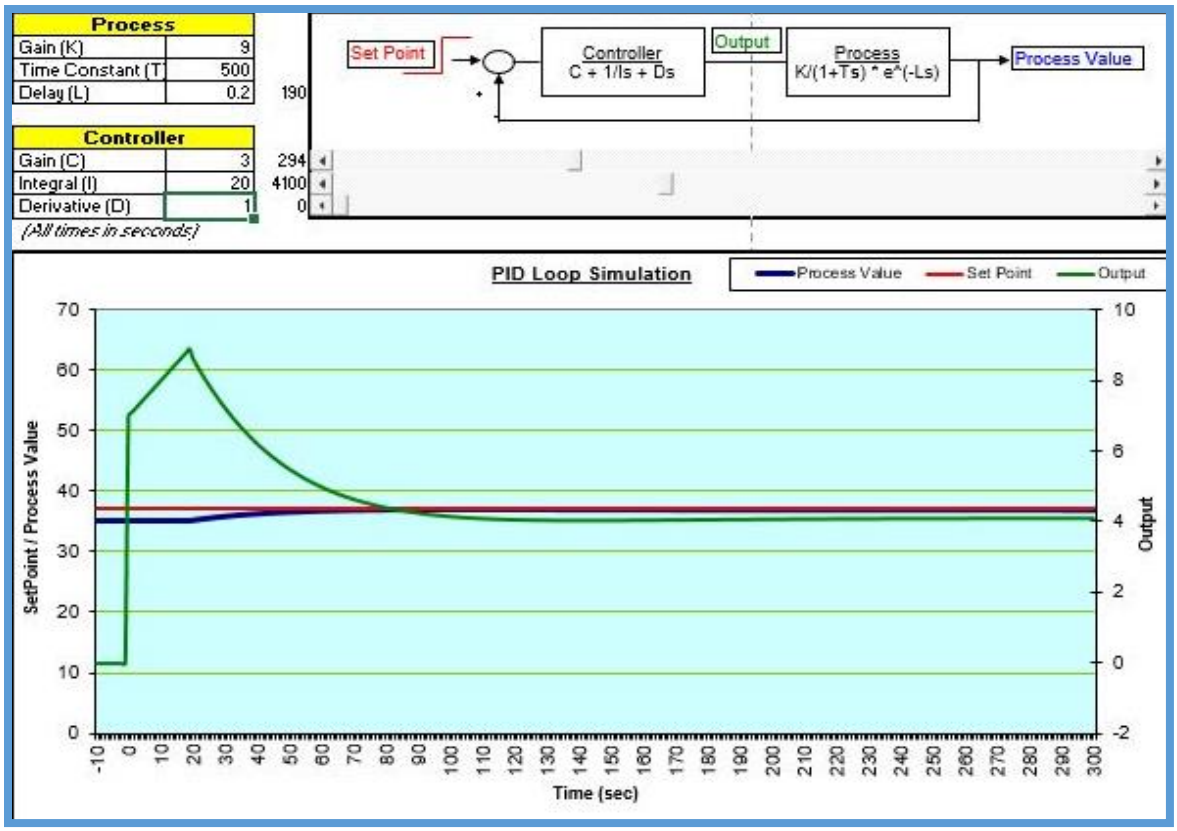


Figure 3-25: PID algorithm response when the neonate temperature was set to 35°C (*PID Loop Simulator*, no date). The process gain, time constant and delay were set to 9, 500 and 0.2 respectively. The controller gain, integral term and derivative terms were set to 3, 20 and 1 respectively.

The changes shown in Figure 3-25 effectively eliminated the oscillations shown in Figure 3-24. The neonate temperature increased from 35°C to 37°C without any overshoot. One improvement to this system would be to decrease the amount of time taken for the neonate temperature to stabilise at 37°C. As can be deduced from Figure 3-25, this occurs at around 50 seconds. Further alternations to the tuning parameters were carried out to decrease this time, as shown in Figure 3-26.

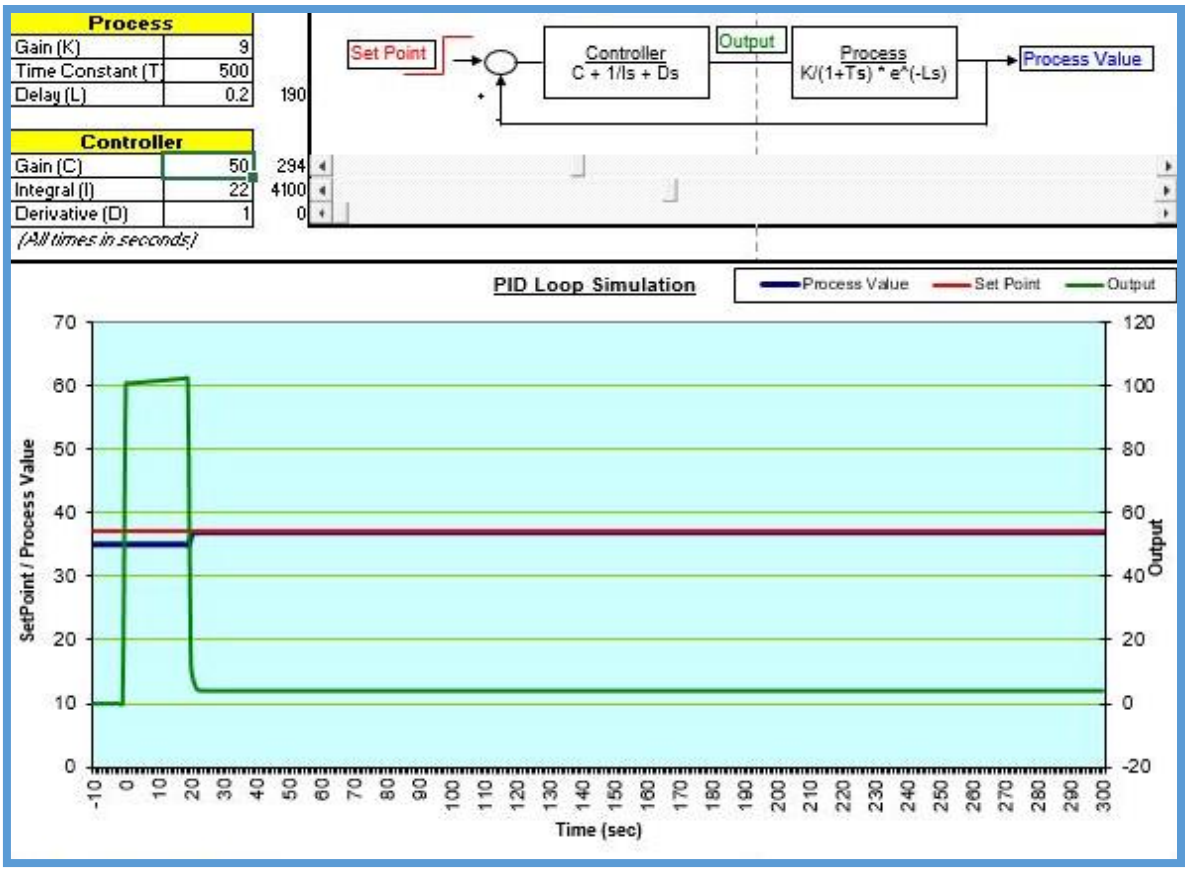


Figure 3-26: Final revision of PID tuning parameters when the neonate temperature was set to 35°C (*PID Loop Simulator*, no date). The process gain, time constant and delay were set to 9, 500 and 0.2 respectively while the controller gain, integral and derivative terms were set to 50, 22 and 1 respectively.

The main difference between Figure 3-25 and Figure 3-26 was the increase in controller gain from 3 to 50. The controller integral term was also increased slightly from 20 to 22, while all other tuning parameters remained constant. This led to a decrease in settling time from around 50 seconds to around 25 seconds. These were deemed to be the ideal tuning parameters for this system, as further changes did not improve the system response.

From Figure 3-26, the controller and process equations were deduced. The generalised controller equation (in the 's' domain) is shown below.

$$T(s) = C + \frac{1}{Is} + Ds$$

In this equation, T(s) represents the neonate temperature in the 's' domain. The variables C, D and I are coefficients that are altered for each unique system to produce the desired transient response. Therefore, for this particular system, the controller equation was deduced.

$$T(s) = 50 + \frac{1}{22s} + s$$

Similarly, the process equation was deduced from the generalised equation shown below. This process equation represents the effect of external factors such as the heating elements, temperature sensors and environmental temperature.

$$Process: \frac{K}{(1 + Ts)^{e^{-Ls}}}$$

In the general equation, T is equal to the time constant, L is equal to delay, and K is equal to the gain of the system. For this particular system, the process equation became as follows.

$$Process: \frac{9}{(1 + 500s)^{e^{-0.2s}}}$$

It was important to determine the system response for other temperatures in order to investigate whether the ideal tuning parameter values changed drastically. This ensured that any temperature fluctuations were adequately compensated for, regardless of whether they were higher or lower than the desired temperature.

Therefore, another system was investigated with a neonate temperature of 36°C. The first iteration using this neonate temperature is shown in Figure 3-27.

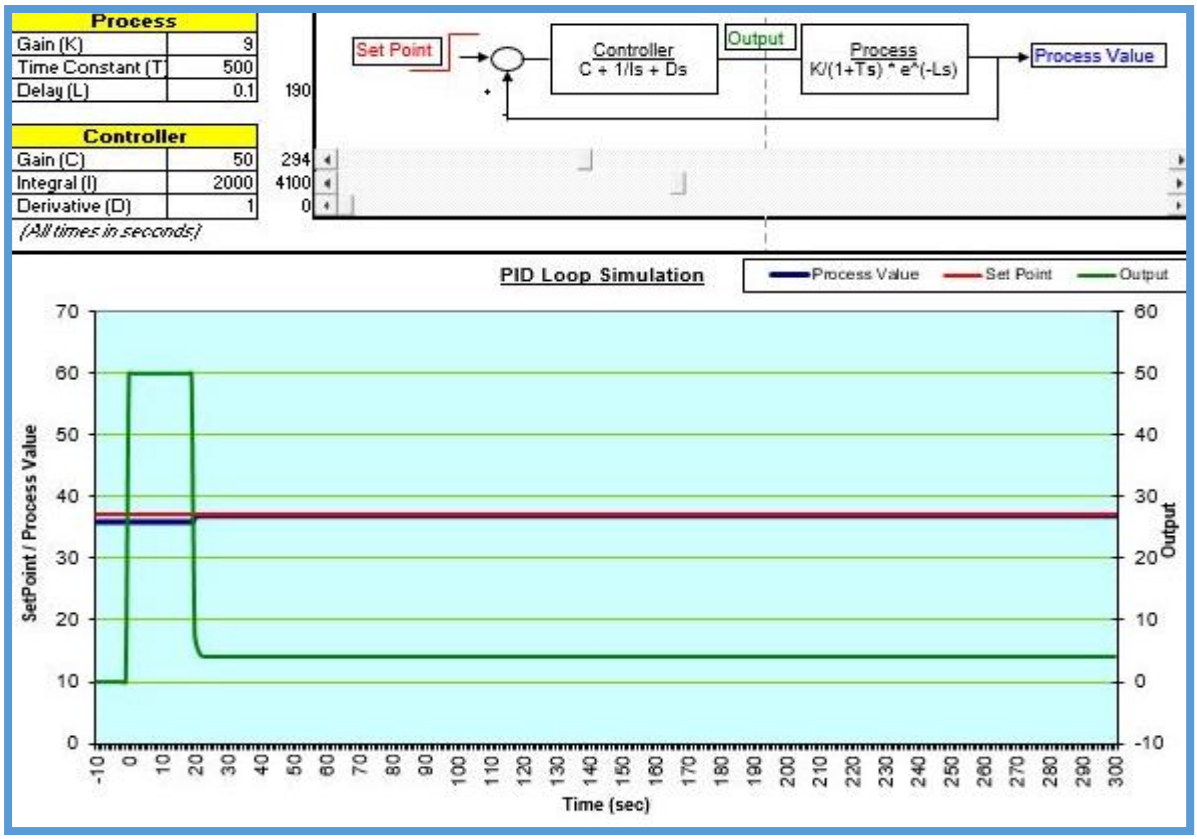


Figure 3-27: PID system response when the initial neonate temperature was 35°C (*PID Loop Simulator*, no date). The process gain, time constant and delay were set to 9, 500 and 0.1 respectively. The controller gain, integral and derivative terms were set to 50, 2000 and 1 respectively.

The tuning parameters shown in Figure 3-27 were obtained after completing a step-by-step analysis similar to that shown when the neonate temperature began at 35°C. However, the integral tuning parameter was significantly different for a system when the initial neonate temperature was 36°C. When the initial neonate temperature increased from 35°C to 36°C, the integral term increased from 22 to 2000. In this case, the controller equation became:

$$T(s) = 50 + \frac{1}{2000s} + s$$

The process equation was also deduced.

$$Process: \frac{9}{(1 + 500s)^{e^{-0.1s}}}$$

In this case, the process equation remained consistent with that obtained when the neonate temperature was 35°C.

The results of a system with an initial neonate temperature of 38°C was also analysed. After numerous independent tests, the ideal parameters were deduced, and are shown in Figure 3-28.

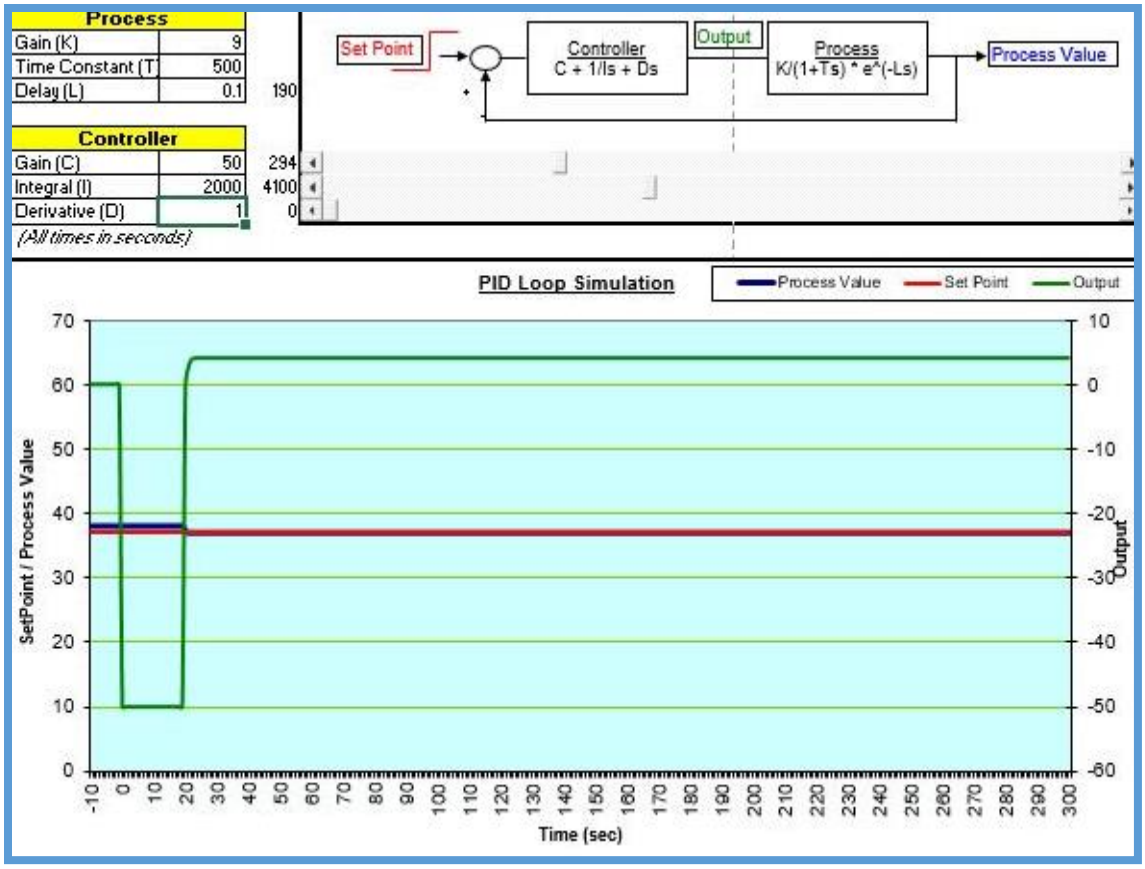


Figure 3-28: Ideal PID response when the initial neonate temperature was set to 38°C (*PID Loop Simulator*, no date). The process gain, time constant and delay were set to 9, 500 and 0.1 respectively. The controller gain, integral and derivative terms were set to 50, 2000 and 1 respectively.

The same parameter values were used when the initial neonate temperature was 38°C as when the neonate temperature was 36°C. This indicates a level of symmetry about the set point, and this was used as a means of simplifying the tuned PID algorithm.

4.6 PULSE SENSING

Although the target neonates are those no longer requiring intensive care, pulse information remains vital to the detection of any underlying issues or complications. This is of additional importance for neonates at home, because there is a chance of SIDS being prevented through early detection of an abnormal pulse.

4.6.1 PULSE SENSOR SELECTION

Pulse monitoring is often a function incorporated into neonatal incubator design and is vital to the successful monitoring of any neonate. For this reason, a pulse sensor was included in the current design. For simplicity, the chosen sensor was a 'plug-and-play' type component, including heart rate amplification and noise-cancellation algorithms in the pre-written software (*Pulse Sensor Australia*, no date). This component was also selected for its low current draw, which is only 4mA at 5V (the ideal operating voltage of the Bluno Mega 2560).

The general principle behind pulse sensors is related to optical measurement. A Light Emitting Diode (LED) and Light Dependent Resistor (LDR) are used, where light is typically shone through the finger. The reduction in intensity after the light has travelled through the finger is measured by the LDR. When the heart beats, more light is absorbed by the blood so the amount of light detected at the LDR decreases. This, in turn, increases the resistance of the LDR, and this alteration in resistance is then converted to a change in voltage by an Operational Amplifier (OP-AMP) (Pujar, 2017).

As with each of the components included in this design, there were multiple pulse sensing options to consider before selecting the one which was most relevant in the current context. These comparisons are summarized in Table 3-25.

Pulse Sensor	Advantages	Disadvantages
Pulse Sensor SEN11574 (*Pulse Sensor - SEN-11574 - SparkFun Electronics*, no date)	Plug-and-play Simple optical sensor Amplification capabilities Noise cancellation capabilities Minimal power draw Convenient clip No soldering required Readily available Arduino code Affordable	

Pulse Sensor	Advantages	Disadvantages
SparkFun Single Lead Heart Rate Monitor – AD8232 (*SparkFun Single Lead Heart Rate Monitor - AD8232 - SEN-12650 - SparkFun Electronics*, no date)	Affordable Acts as OP-AMP Amplifies and filters heart rate signal Pins to attach for arms and legs included LED indicator pulses in synchronization with heart rate	
Biomedical Sensor Pad (*Biomedical Sensor Pad (10 pack) - SEN-12969 - SparkFun Electronics*, no date)	Measure ECG, EEG and EMG Easily adheres to skin	Limited number of pads (i.e. 10 disposable pads)
Arduino Pulse Heart Rate Sensor (*Arduino Pulse Heart Rate Sensor*, no date)	Can be used on earlobe or finger Paired with premade application Amplification and noise elimination functionality Affordable	

Pulse Sensor	Advantages	Disadvantages
AD8232 ECG Heart Monitor Measurement Sensor Module Pulse Signal Board +Cable (*AD8232 ECG Heart Monitor Measurement Sensor Module Pulse Signal Board +Cable | eBay*, no date)	Uses double poles high pass filter Eliminates noise from movement and half-cell potential[25] Incorporates OP-AMP Ultra low power Affordable	
Heartbeat Detection Sensor Module By Finger 5V For Arduino Phototransistor (*5x Heartbeat Detection Sensor Module By Finger 5V For Arduino Phototransistor | eBay*, no date)	Compatible with Arduino	Decreased accuracy due to external light sources (noise) Unconfirmed operating temperature

[25] Half-cell potential is "the potential developed at the electrode of each half cell in an electrochemical cell" (*What is Half-Cell Potential?*, no date).

Pulse Sensor	Advantages	Disadvantages
GY-MAX30100 Low Power Heart Rate Click Sensor Module for Arduino (*GY-MAX30100 Low Power Heart Rate Click Sensor Moudle for Arduino | eBay*, no date)		
DFRobot Gravity: Analog Heart Rate Monitor Sensor ECG for Arduino (*DFRobot Gravity: Analog Heart Rate Monitor Sensor ECG for Arduino | eBay*, no date)	Includes AD8232 chip to remove noise Couples with 'Serial Plotter' code in Arduino IDE Plug-and-play	Requires expansion shield to connect to Arduino
Heart Rate Click module Sensor Blood Oxygen Pulse Module for Arduino MAX30100 (*Heart Rate Click module Sensor Blood Oxygen Pulse Module for Arduino MAX30100 | Alibaba Group*, no date)	Pulse oximetry functionality included	

Table 3-25: Summary of various pulse sensor components.

The pulse sensor selected in the design also needed to have an accuracy which effectively captured any changes that could threaten the wellbeing of the neonate. In order to determine the required accuracy, it was important to first determine the intervals at which heart rate could become critical. The 'healthy' heart rate of a neonate is between 100 and 165 beats per minute when awake, and between 90 and 160 beats per minute when sleeping (*Pediatric Vital Signs: Charts of Normal Ranges*, no date).

These pulse rate ranges are quite large, and so the accuracy and precision of the pulse sensor can be slightly more flexible than for the temperature sensor. Therefore, the accuracy of the pulse sensor could be as large as +/- 5 beats per minute, and the resolution could also be 5 beats per minute, and the critical changes could still be anticipated and prevented.

The advantages and disadvantages outlined in Table 3-25 have been directly translated into a decision matrix shown in Table 3-26. In this way, the advantages and disadvantages can be quantitatively compared in order to ensure that the ideal pulse sensor is included in the final design.

Pulse Sensor		Cost ($AUD)	Accuracy (%)	Repeatability	Sampling Rate	Operating Voltage (V)	Average Weighting
	Weighting	**0.10**	**0.40**	**0.20**	**0.20**	**0.10**	**N/A**
Pulse Sensor SEN11574 (*Pulse Sensor - SEN-11574 - SparkFun Electronics*, no date)	Value	42.44	N/A*	N/A*	N/A*	3.00 or 5.00	N/A
	Rating	0.40	N/A*	N/A*	N/A*	1.00	N/A
	Weighted Rating	0.04	N/A*	N/A*	N/A*	0.10	0.07
GY-MAX30100 Low Power Heart Rate Click Sensor Module for Arduino (*GY-MAX30100*	Value	7.49	N/A*	N/A*	N/A*	N/A*	N/A
	Rating	1.00	N/A*	N/A*	N/A*	N/A*	N/A

Pulse Sensor		Cost ($AUD)	Accuracy (%)	Repeatability	Sampling Rate	Operating Voltage (V)	Average Weighting
	Weighting	**0.10**	**0.40**	**0.20**	**0.20**	**0.10**	**N/A**
Low Power Heart Rate Click Sensor Moudle for Arduino | eBay, no date)	Weighted Rating	0.10	N/A*	N/A*	N/A*	N/A*	0.10
SparkFun Single Lead Heart Rate Monitor – AD8232 (*SparkFun Single Lead Heart Rate Monitor - AD8232 - SEN-12650 - SparkFun Electronics*, no date)	Value	29.19	N/A*	N/A*	N/A*	3.30	N/A
	Rating	0.60	N/A*	N/A*	N/A*	0.20	N/A
	Weighted Rating	0.06	N/A*	N/A*	N/A*	0.02	0.04
DFRobot Gravity: Analog Heart Rate Monitor Sensor ECG for Arduino (*DFRobot Gravity: Analog*	Value	26.18	N/A*	N/A*	N/A*	3.30 to 6.00	N/A
	Rating	0.60	N/A*	N/A*	N/A*	1.00	N/A

Pulse Sensor		Cost ($AUD)	Accuracy (%)	Repeatability	Sampling Rate	Operating Voltage (V)	Average Weighting
	Weighting	**0.10**	**0.40**	**0.20**	**0.20**	**0.10**	**N/A**
Heart Rate Monitor Sensor ECG for Arduino | eBay, no date)	Weighted Rating	0.06	N/A*	N/A*	N/A*	0.10	0.08
Heartbeat Detection Sensor Module By Finger 5V For Arduino Phototransistor (*5x Heartbeat Detection Sensor Module By Finger 5V For Arduino Phototransistor | eBay*, no date)	Value	12.88	N/A*	N/A*	N/A*	3.00 to 5.00	N/A
	Rating	0.80	N/A*	N/A*	N/A*	1.00	N/A
	Weighted Rating	0.08	N/A*	N/A*	N/A*	0.10	0.09

Table 3-26: Decision matrix comparing commercially available pulse sensors.

Considering the distinct lack of information provided in the specifications for each of the above pulse sensors, the decision matrix shown in Table 3-26 was used as a secondary form of decision making, after weighing the advantages and disadvantages of each sensor.

The SEN-11574 pulse sensor was ultimately decided upon due to its simple 'plug-and-play' nature, and the extra cost was deemed appropriate given the reduced time required for assembly. The decision matrix outlined in Table 3-26 does not reflect this compromise between cost and time requirements, but was still an important step in deciding upon the most relevant and appropriate pulse sensor.

4.6.2 PULSE SENSOR ASSEMBLY

The three pins of the selected pulse sensor were connected to the Arduino as shown in Figure 3-29, where each of the pulse sensor cables with the male header pins was inserted directly into an appropriate header strip on the microcontroller (*A PulseSensor library (for Arduino)*, no date).

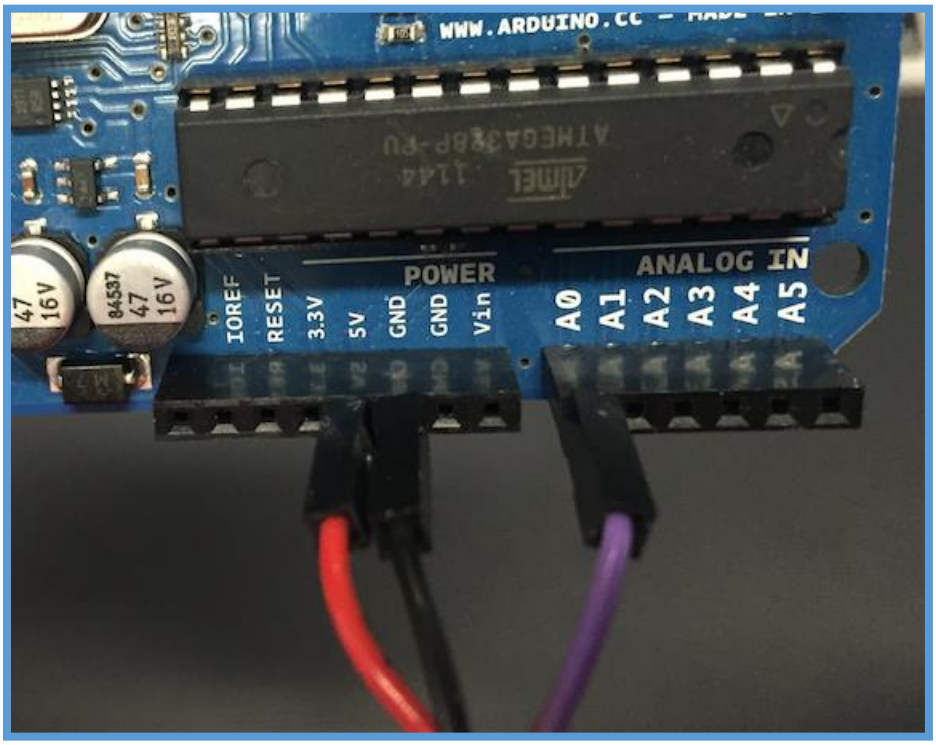


Figure 3-29: Connection of the pulse sensor to the Arduino (*PulseSensor_Amped_Arduino: PulseSensor Arduino code for BPM and Processing-Visualizer*, no date).

For additional clarification, the connections depicted in Figure 3-29 have been reiterated in Table 3-27.

Microcontroller Ports	Pulse Sensor Wires
5V	Red
GND	Black
A0	Purple

Table 3-27: Hardware connections between the microcontroller and the pulse sensor.

4.6.3 PULSE SENSOR CODE

Java code was used which displayed the pulse rate graphically and numerically in the Processing IDE. This was first employed individually in order to test the accuracy of the pulse sensor without the remaining components of the neonatal warming blanket. The code written only for this functionality is shown in Section 9 of this thesis. The resulting display is shown in Figure 3-30.

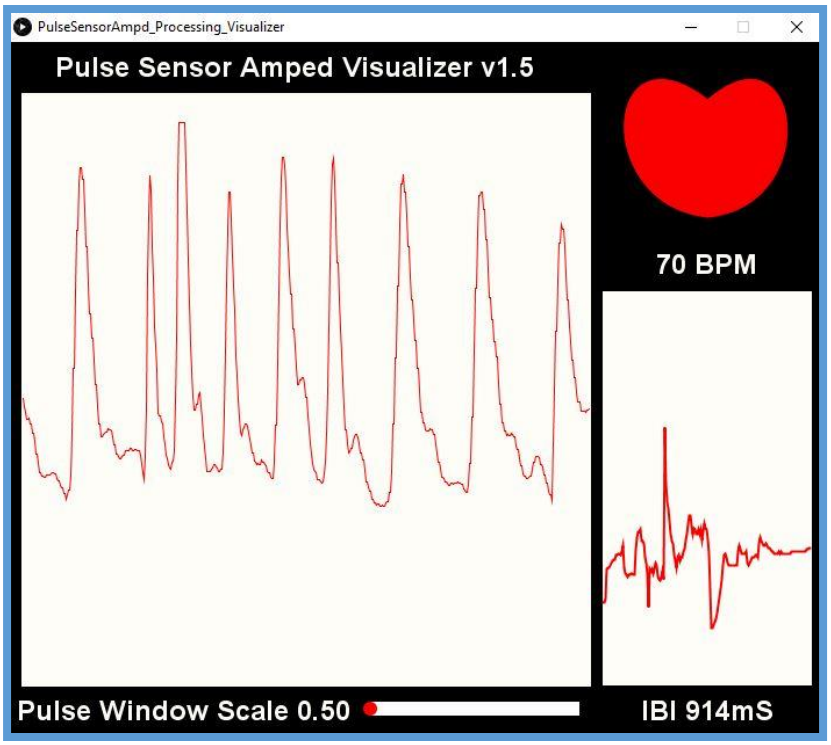


Figure 3-30: Example of the Processing sketch used to display heart rate graphically and numerically.

4.6.4 TESTING PULSE SENSOR INDEPENDENTLY

Android applications are available to measure heart rate, and these have varying degrees of accuracy. It was decided that this type of baseline sensor should be considered due to its convenience, provided the accuracy of these applications was sufficient. The documented accuracies of commercially available applications have been compared in Table 3-28.

Application	Accuracy (BPM)
iCare Health Monitor	"Heart rate error : Within plus or minus 3, Valid range: 50~150" (*iCare Health Monitor (BP & HR)*, no date)
Instant Heart Rate	4.5 ± 1.1 (Coppetti *et al.*, 2017)

Application	Accuracy (BPM)
Heart Fitness	2 ± 0.5 (Coppetti *et al.*, 2017)
What's my Heart Rate	7.1 ± 1.4 (Coppetti *et al.*, 2017)
Cardiio	8.1 ± 1.4 (Coppetti *et al.*, 2017)

Table 3-28: Comparison of various heart rate applications, in terms of measured accuracy.

There are also wearable measurement devices, such as those produced by FitBit. However, there has been evidence of inaccurate measurements produced by these devices. Namely, "the PurePulse Trackers exhibited an aggregate mean bias of -8.9 bpm and a mean absolute differential of 13.9 bpm when compared against ECG" (Jo and Dolezal, no date). This increases significantly to a bias of -16.8 bpm and a differential of 19.2 bpm when the user is undertaking strenuous activity (Jo and Dolezal, no date). Multiple studies reflect similar results, indicating a lack of accuracy in the measurement of heart rate by FitBit devices (Benedetto *et al.*, 2018). Wearable devices such as the BodyMedia FIT have a higher degree of accuracy, which has been tested by the manufacturing organization and found to be greater than 90%.

Electrocardiogram, or ECG, was also considered as an alternative, although the student did not have access to the equipment required. For this reason, one of the most accurate of the above heart rate sensors (namely, iCare Health Application with a variability of 3 BPM) was used. This accuracy was deemed suitable for the initial confirmation that the chosen pulse sensor was accurate. However, further refining of this accuracy measurement would be required upon completion of this thesis, were the blanket to be tested on neonates in the future.

The results obtained when comparing the iCare Health Application to the selected sensor have been detailed in Table 3-29.

Conditions	Data Source	Trial 1 (BPM)	Trial 2 (BPM)	Trial 3 (BPM)	Average (BPM)	Range (BPM)
At rest	iCare Health Monitor Application	91	90	93	91	3
	Arduino Pulse Sensor	86	87	85	86	2
After average-paced walk (Very Light Exercise)	iCare Health Monitor Application	100	105	106	104	6
	Arduino Pulse Sensor	99	97	98	98	2
After brisk walk (Light Intensity Exercise)	iCare Health Monitor Application	111	119	100	110	19
	Arduino Pulse Sensor	108	108	106	107	2

Table 3-29: Results obtained when testing the validity of the pulse sensor independently.

Based on the data detailed in Table 3-29, Figure 3-31 could be deduced.

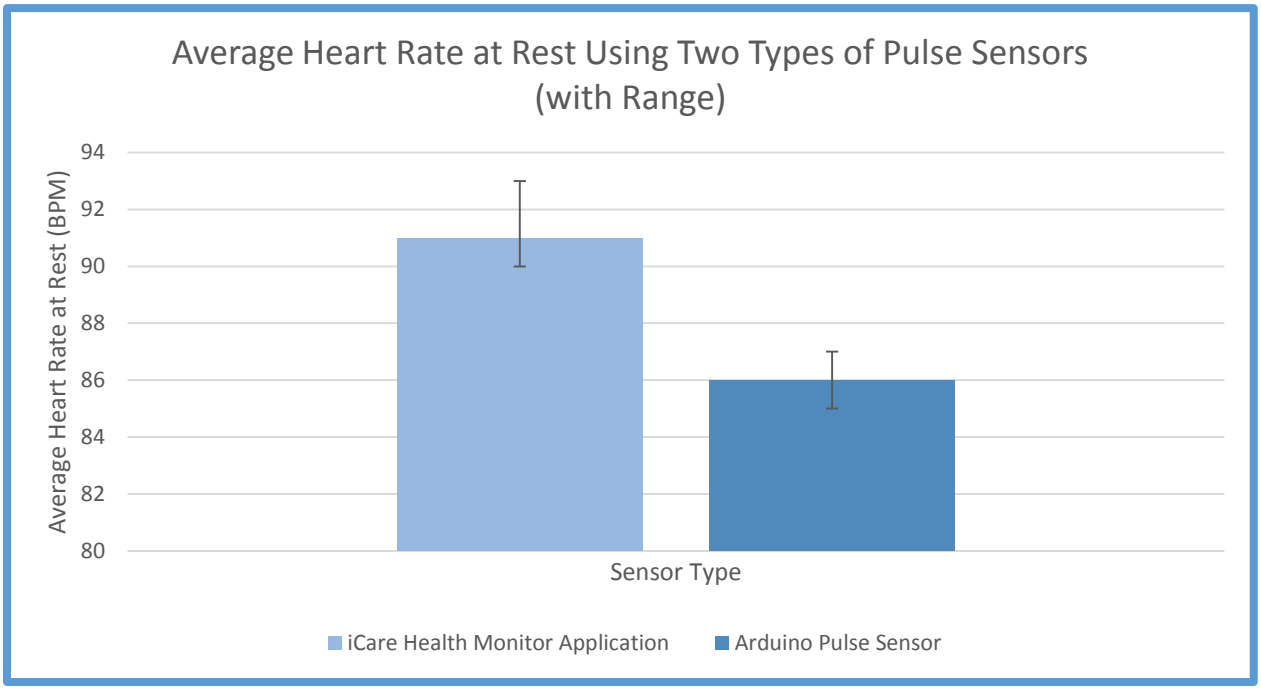


Figure 3-31: Comparison of measured heart rate using two separate sensors, while at rest.

Figure 3-31 highlights the difference between the average heart rate obtained using the iCare Application and the Arduino Pulse Sensor, as well as the difference in range. It is evident that the range is greater for the iCare Health Monitor Application, which was expected due to the decreased accuracy of this sensor compared to the Arduino Pulse Sensor.

However, in order to determine whether there was a statistically significant difference between the results obtained with these two sensors, a paired sample t-test was performed in Excel. The results of this test are shown in Table 3-30.

	iCare Health Monitor Application Data	Arduino Pulse Sensor Data
Mean	91.33333	86
Variance	2.333333	1
Observations	3	3
Pearson Correlation	-0.98198	

	iCare Health Monitor Application Data	Arduino Pulse Sensor Data
Hypothesized Mean Difference	0	
df	2	
t Stat	3.670652	
P(T<=t) one-tail	0.033431	
t Critical one-tail	2.919986	
P(T<=t) two-tail	0.066861	
t Critical two-tail	4.302653	

Table 3-30: Paired sample t-test for sensor data while at rest. The significance value for a two-tailed test has been highlighted in yellow.

From Table 3-30, the P value for a two-tailed test is approximately 0.0669. For the difference between these two sensors to be significant, this P value needs to be less than 0.05. As this is not the case, there is no statistical significance between the two sensors. This confirms that this sensor is appropriate for use in the thesis while at rest, as there was no significant difference between the values of this sensor and those of the reference, while at rest.

The percentage error of the Arduino Pulse Sensor could also be calculated using the following equation:

$$Percentage\ Error\ (\%) = \frac{Observed\ Value - Theoretical\ Value}{Theoretical\ Value} * 100$$

Therefore, the percentage error of the sensor while measurements were taken at rest can be calculated.

$$Percentage\ Error\ (\%) = \left|\frac{86 - 91}{91} * 100\right| = 5.49\%$$

Similarly, the data output from both sensors during very light exercise has been depicted in Figure 3-32.

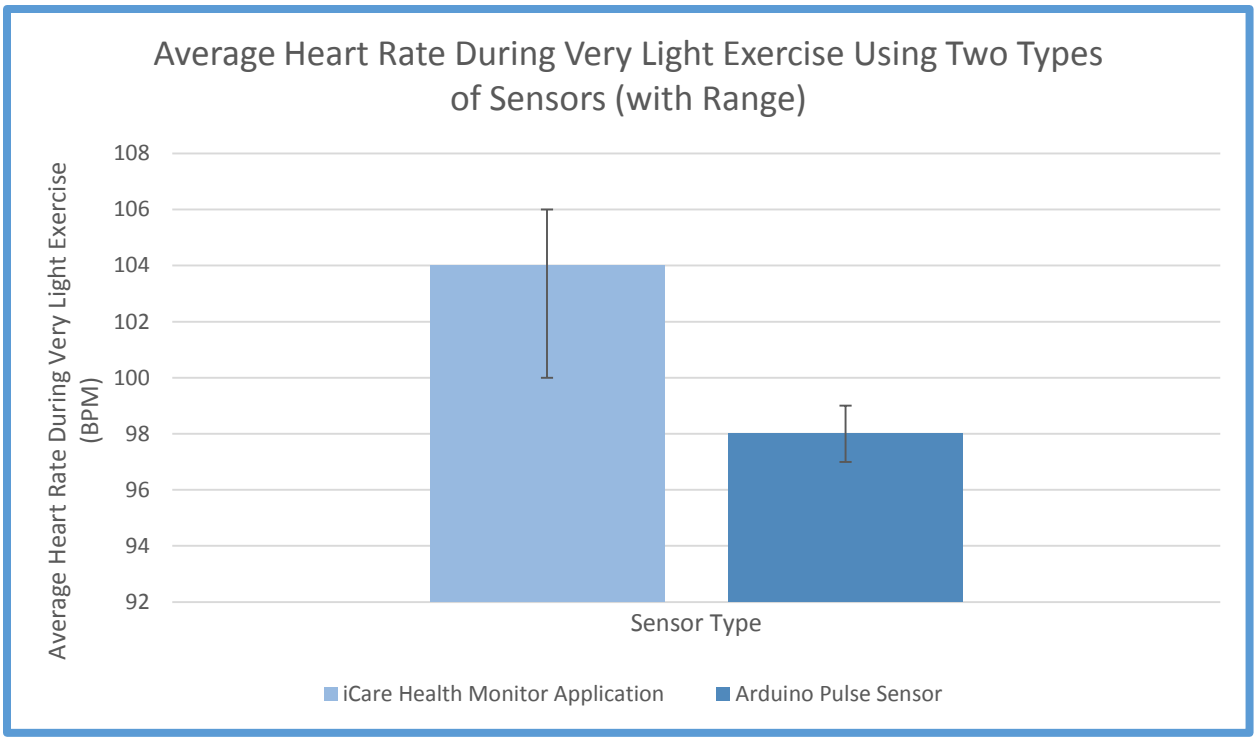


Figure 3-32: Comparison of measured heart rate using two separate sensors, during very light intensity exercise.

Figure 3-32 once again confirms that the range is much greater for the iCare Health Monitor Application. To determine whether the difference between these two sensors is statistically significant during very light intensity exercise, an additional paired sample t-test was conducted using Excel. This test is outlined in Table 3-31.

	iCare Health Monitor Application Data	Arduino Pulse Sensor Data
Mean	103.6667	98
Variance	10.33333	1
Observations	3	3
Pearson Correlation	-0.77771	
Hypothesized Mean Difference	0	
df	2	
t Stat	2.428571	

	iCare Health Monitor Application Data	Arduino Pulse Sensor Data
P(T<=t) one-tail	0.067921	
t Critical one-tail	2.919986	
P(T<=t) two-tail	0.135841	
t Critical two-tail	4.302653	

Table 3-31: Paired sample t-test for very light intensity exercise data. The significance value of a two-tailed test has been highlighted in yellow.

Table 3-31 shows that the P value for a two-tailed test is approximately equal to 0.136. Once again, this value is greater than 0.05, so the difference between these two sensors is not statistically significant during very light exercise.

The percentage error under these conditions can also be calculated as follows.

$$Percentage\ Error\ (\%) = \left|\frac{98 - 104}{104} * 100\right| = 5.77\%$$

This shows that the percentage error increases slightly as the activity level increases from rest to very light intensity exercise. Finally, Figure 3-33 was deduced in order to determine the difference between the two sensors during light intensity exercise.

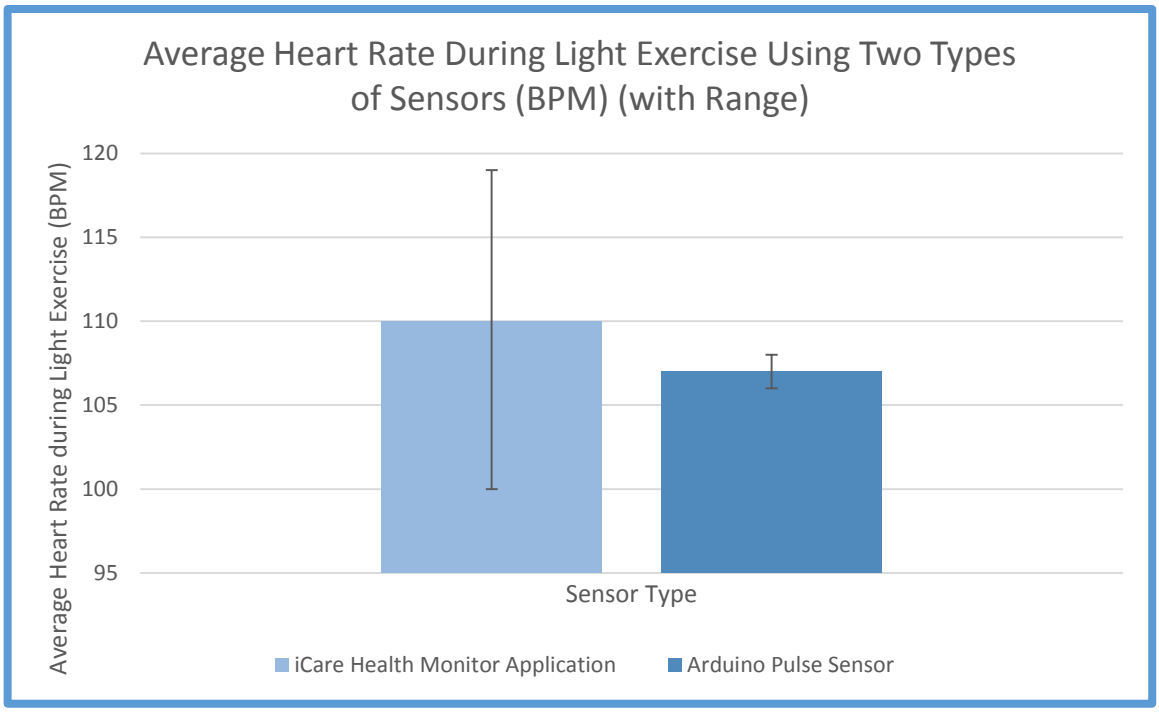


Figure 3-33: Comparison of measured heart rate using two separate sensors, during light intensity exercise.

Figure 3-33 once again confirms that the range is much greater for the iCare Health Monitor Application than the Arduino Pulse Sensor. In order to determine whether there is a significant difference between the two sensors during light intensity exercise, the paired sample t-test shown in Table 3-32 was conducted in Excel.

	iCare Health Monitor Application Data	Arduino Pulse Sensor Data
Mean	110	107.3333
Variance	91	1.333333
Observations	3	3
Pearson Correlation	0.907841	
Hypothesized Mean Difference	0	
df	2	
t Stat	0.543075	

	iCare Health Monitor Application Data	Arduino Pulse Sensor Data
P(T<=t) one-tail	0.320756	
t Critical one-tail	2.919986	
P(T<=t) two-tail	0.641511	
t Critical two-tail	4.302653	

Table 3-32: Paired sample t-test for the two sensors based on light exercise data. The significance value of a two-tailed test has been highlighted in yellow.

Table 3-32 shows that the P value for a two tailed test is approximately 0.642, which once again shows that the difference between these two sensors is not significant during light intensity exercise.

The percentage error in these conditions could be calculated as follows.

$$Percentage\ Error\ (\%) = \left|\frac{107 - 110}{110} * 100\right| = 2.72\%$$

Interestingly, this shows that the percentage error decreased significantly when the activity level increase to light exercise. This is promising provided the fact that neonatal heart rates would most likely fall in this range.

The last comparison for pulse sensor data involved comparing the obtained results to the range deemed healthy in literature. These values have ben outlined in Table 3-33.

Conditions	BPM
At rest	60-100 ('Resting heart rate', 2018)
After average-paced walk (Very Light Exercise)	97-116 (Bumgardner, 2017)
After brisk walk (Light Intensity Exercise)	108-130 (*Target Heart Rate Calculator*, no date)

Table 3-33: Healthy heart rate range according to literature for three separate conditions (at rest, very light exercise and light intensity exercise).

As can be seen from Figure 3-31, Figure 3-32 and Figure 3-33, the values obtained by the Arduino pulse sensor fell within the range deemed 'healthy' in literature, as did the values obtained by the iCare Health Monitor. When the heart rate was measured at rest, both values obtained by the sensors were at the higher end of the healthy range. When the heart rate was measured after very light or light intensity exercise, these sensor values were at the lower end of the healthy spectrum.

4.7 HUMIDITY SENSING

Humidity sensing, as mentioned previously, can assist in the prevention and monitoring of pressure sores in the neonatal sector, and is therefore a beneficial addition to the design.

4.7.1 HUMIDITY SENSOR SELECTION

As with the temperature and pulse sensors discussed in previous sections of this thesis, there are many options available for humidity sensing using an Arduino microcontroller. For the current application, a DHT22 Temperature and Relative Humidity Sensor Module was selected, although the various options are outlined in Table 3-34.

Humidity Sensor		Cost ($AUD)	Accuracy (%)	Sampling Rate (Hz)	Operating Power (V)	Average Weighting
	Weighting	**0.20**	**0.40**	**0.20**	**0.20**	**N/A**
AM2315 - Encased I2C Temperature/Humidity Sensor (*AM2315 - Encased I2C Temperature/Humidity Sensor*, no date)	Value	43.81	+/- 2.00	0.50	3.30 to 5.50	N/A
	Rating	0.30	1.00	0.50	1.00	N/A
	Weighted Rating	0.06	0.40	0.10	0.20	0.19
DHT11 basic temperature-humidity sensor (*DHT11 basic temperature-humidity sensor + extras*,	Value	9.00	+/- 5.00	1.00	3.00 to 5.00	N/A
	Rating	1.00	0.20	1.00	1.00	N/A

Humidity Sensor		Cost ($AUD)	Accuracy (%)	Sampling Rate (Hz)	Operating Power (V)	Average Weighting
	Weighting	**0.20**	**0.40**	**0.20**	**0.20**	**N/A**
no date)	Weighted Rating	0.20	0.08	0.20	0.20	0.17
Grove – Temperature & Humidity Sensor Pro (Seeed Studio) (*Grove - Temperature&Humidity Sensor Pro*, no date)	Value	25.35	+/- 2.00	N/A	3.30 to 6.00	N/A
	Rating	0.80	1.00	0.00	1.00	N/A
	Weighted Rating	0.16	0.40	0.00	0.20	0.19
Humidity and Temperature Sensor - RHT03 (*Humidity and Temperature Sensor - RHT03 - SEN-10167 - SparkFun Electronics*, no date)	Value	16.89	+/- 2.00	0.50	3.30 to 6.00	N/A
	Rating	0.80	1.00	0.50	1.00	N/A
	Weighted Rating	0.16	0.40	0.10	0.20	0.22

Humidity Sensor		Cost ($AUD)	Accuracy (%)	Sampling Rate (Hz)	Operating Power (V)	Average Weighting
	Weighting	**0.20**	**0.40**	**0.20**	**0.20**	**N/A**
DHT22 Temperature and Relative Humidity Sensor Module (*DHT22 Temperature and Relative Humidity Sensor Module Australia*, no date)	Value	25.00	+/- 2.00 to 5.00	0.50	3.00 to 5.00	N/A
	Rating	0.80	1.00	0.50	1.00	N/A
	Weighted Rating	0.16	0.40	0.10	0.20	0.22
SHT1x Humidity and Temperature Sensor (*Gravity: SHT1x Humidity and Temperature Sensor Australia*, no date)	Value	30.80	+/- 4.50	N/A	3.30 to 5.00	N/A
	Rating	0.40	0.20	0.00	1.00	N/A
	Weighted Rating	0.08	0.08	0.00	0.20	0.09

Table 3-34: Decision matrix comparing the various commercially available humidity sensors.

4.7.2 HUMIDITY SENSOR ASSEMBLY

The connection of the DHT22 Temperature and Relative Humidity Sensor Module to the microcontroller was straight forward. The leftmost pin was connected to the 3-5V pin on the Arduino, and the rightmost pin was connected to the GND pin on the Arduino. An example of these connections can be seen in Figure 3-34.

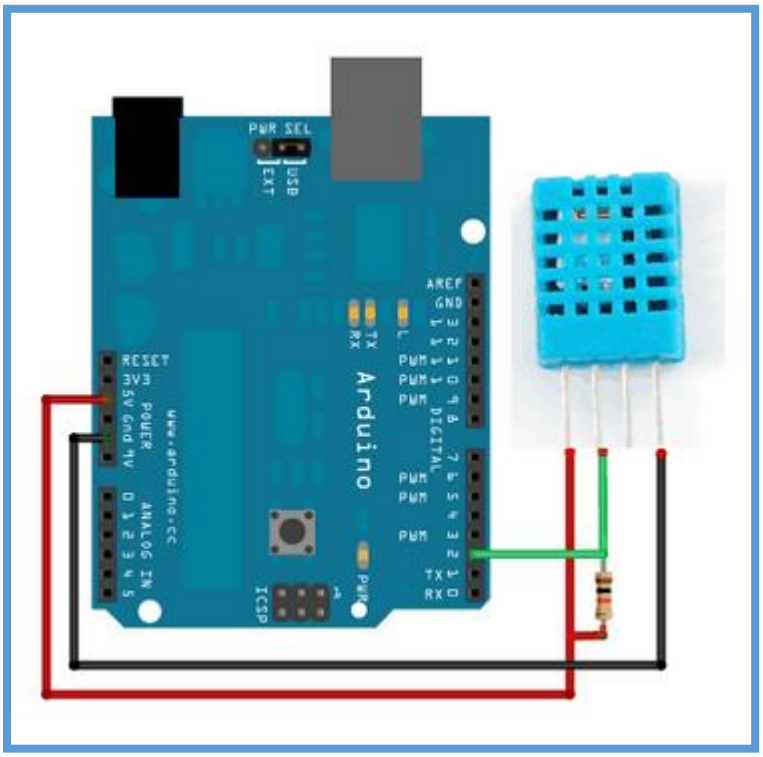


Figure 3-34: Connections made between an Arduino microcontroller, and a DHT22 sensor (*Connecting to a DHTxx Sensor | DHT11, DHT22 and AM2302 Sensors | Adafruit Learning System*, no date).

In order to pair the jumper wires with the humidity sensor pins, soldering was required. This is because the available female ports of the jumper wires were too large, and there was not an adequate physical connection without soldering.

Before soldering, four strips of wire were required. The tips of these wires were stripped in order to expose the wire beneath. Figure 3-35 shows the stripped wire as well as the frayed ends which were deemed unsuitable for soldering to the humidity sensor pins.

Figure 3-35: Stripped wires to be soldered onto the humidity sensor.

As these frayed wires were deemed unsuitable for soldering, they were simply twisted and soldered together in order to ensure that there were four distinct wires to attach to each of the four pins on the humidity sensor. The result when the ends of the wires were soldered is depicted in Figure 3-36.

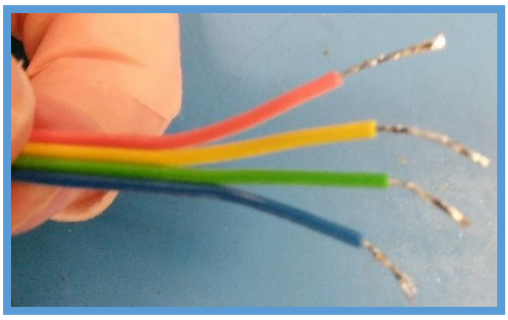


Figure 3-36: Result when the ends of the wires were twisted and solder was applied.

Four lengths of heat shrink wrap were then cut and slid onto each of the four wires, ensuring that they were placed low enough on the wires that eventual soldering of the wires would not cause premature shrinkage of the heat shrink wrap. Figure 3-37 depicts this step.

Figure 3-37: Heat shrink wrap placed onto each of the four wires.

Next, each of the four wires could be soldered onto the four pins of the humidity sensor, as shown in Figure 3-38.

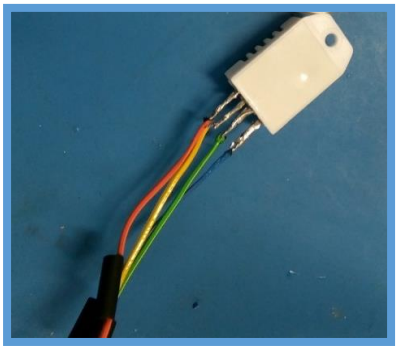


Figure 3-38: Soldered connections between the four wires and the pins of the humidity sensor.

The heat shrink wrap could then be slid up to lay flush with the edge of the humidity sensor, such that all exposed wires were covered. Not only was this necessary due to the health implications of touching a charged wire carrying a current once the sensor was connected to the blanket, but the wires could also create a short circuit if they were to touch, and this would render the sensor inoperable. Figure 3-39 depicts the use of this shrink wrap to cover the exposed wires.

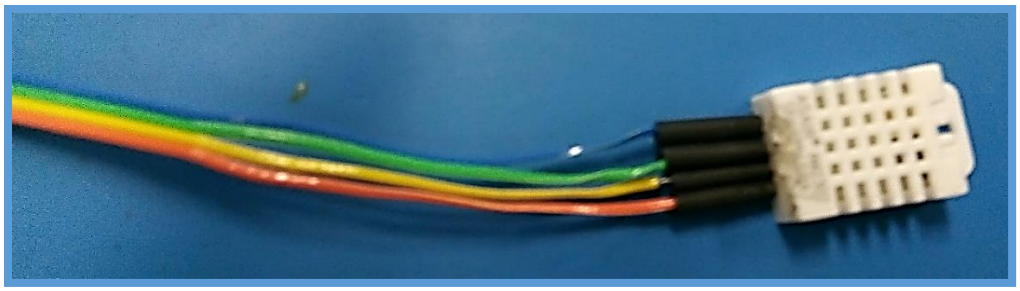


Figure 3-39: Application of heat shrink wrap to cover exposed wires.

One issue that is partially visible in Figure 3-39 is the tendency of the DHT22 sensor to melt when exposed to heat. This was discovered when a heat gun was used to warm the heat shrink, but instead resulted in the destruction of one of the humidity sensors.

For this reason, the more durable, encased DHT22 sensor was purchased, as this was a temperature sensitive application and failure of sensors due to heat produced by the heating elements was deemed unacceptable. This is due to the necessity for sensors to work accurately under warm conditions, but also because melted plastic from the sensor enclosure could burn the neonate. The enclosed DHT22 (renamed as AM2302) sensor is shown in Figure 3-40.

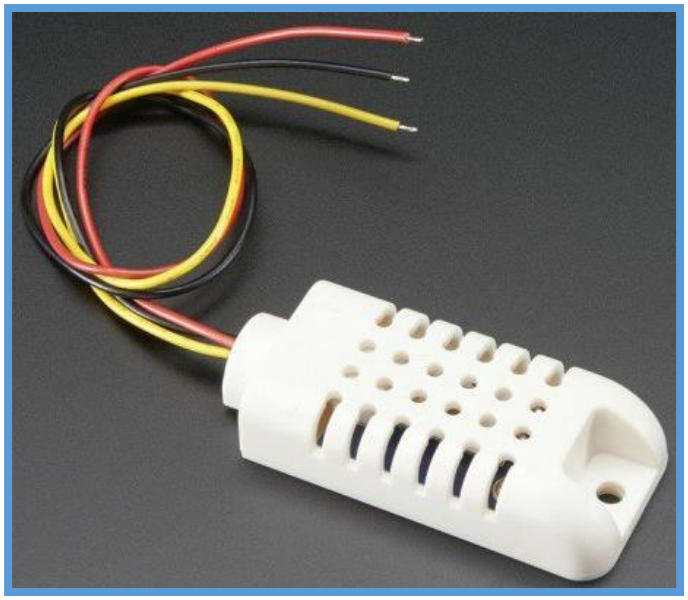


Figure 3-40: Enclosed DHT22 sensor, or AM2302 sensor (*AM2302 (wired DHT22) temperature-humidity sensor Australia*, no date).

4.7.3 HUMIDITY SENSOR CODE

The code used to test the humidity sensor independently is detailed in Appendix B (Individual Humidity Sensor Code). This code output data, as shown in Figure 3-41.

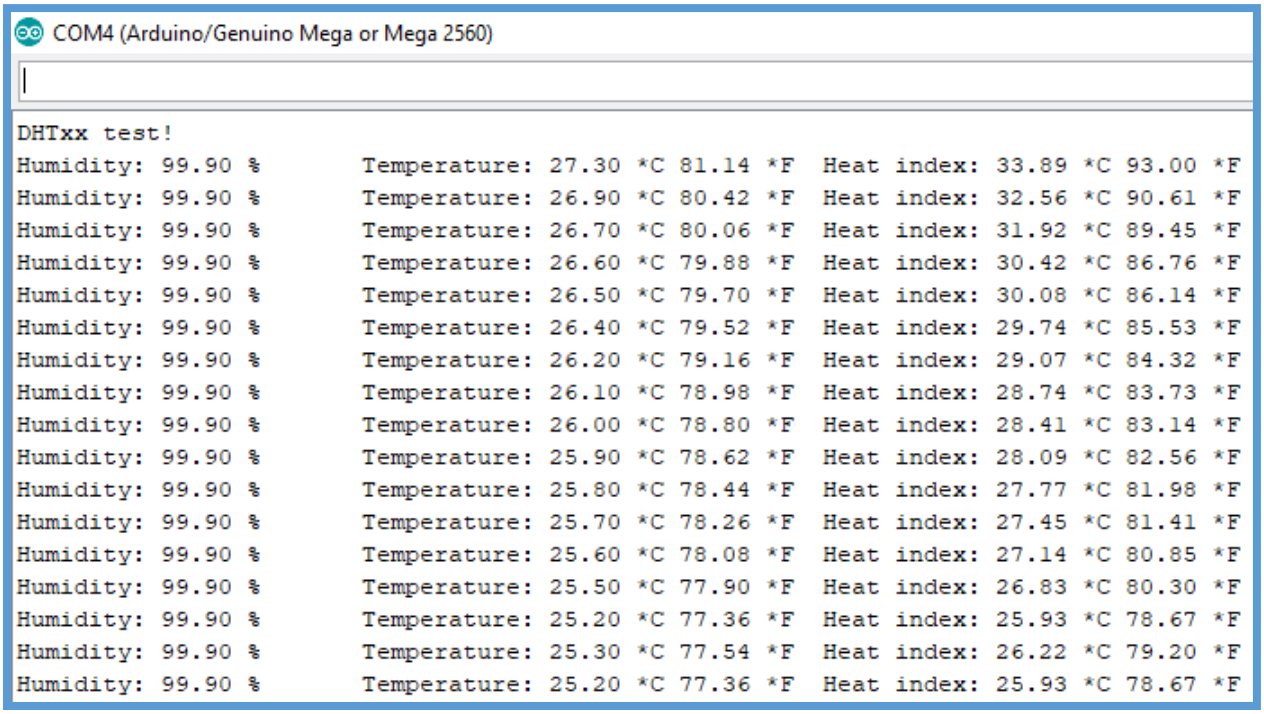


Figure 3-41: Example of the data output in the Serial Monitor of the Arduino IDE.

4.7.4 TESTING HUMIDITY SENSOR INDEPENDENTLY

Similarly to the process carried out above for the verification of the temperature and pulse sensors, the output of the selected humidity sensor was also compared to the output of a high accuracy humidity sensing device.

The DT-625 humidity sensor used by the Flinders Medical Centre has an accuracy of +/- 2% relative humidity (RH) when measuring humidity levels between 20% and 80% RH . This accuracy decreases slightly to ±2.5% RH when the humidity level falls outside of this range. Considering the fact that pressure sores are not likely to instantaneously occur, and will not occur suddenly if the humidity increases ±3% RH, this range is considered adequate for the current application. Therefore, the DT-625 humidity sensor has been used as the reference sensor in this case.

The tests outlined in Table 3-35 were necessary in order to ensure the effective functioning of the humidity sensor prior to its incorporation into the final design. Note that in subsequent sections of the thesis, RH is used as an abbreviation for the relative humidity.

Conditions	Data Source	Trial 1 (%RH)	Trial 2 (%RH)	Trial 3 (%RH)	Average (%RH)	Range (%RH)
Indoors (High Humidity)	DHT22 Temperature and Humidity Sensor	99.90	99.90	99.90	99.90	0.00
	DT-625 Thermometer and Humidity Meter	99.00	99.20	98.80	99.00	0.40
Indoors (Typical Conditions)	DHT22 Temperature and Humidity Sensor	68.50	68.60	69.60	68.90	1.10
	DT-625 Thermometer and Humidity Meter	71.20	71.30	71.30	71.27	0.10

Table 3-35: Results obtained when verifying the validity of the humidity sensor. The DT-625 Thermometer and Humidity Meter was used as the reference sensor in this case, as this was a hospital-grade sensor obtained from the Flinders Medical Centre.

The results outlined in Table 3-35 have been depicted in Figure 3-42 for high humidity conditions.

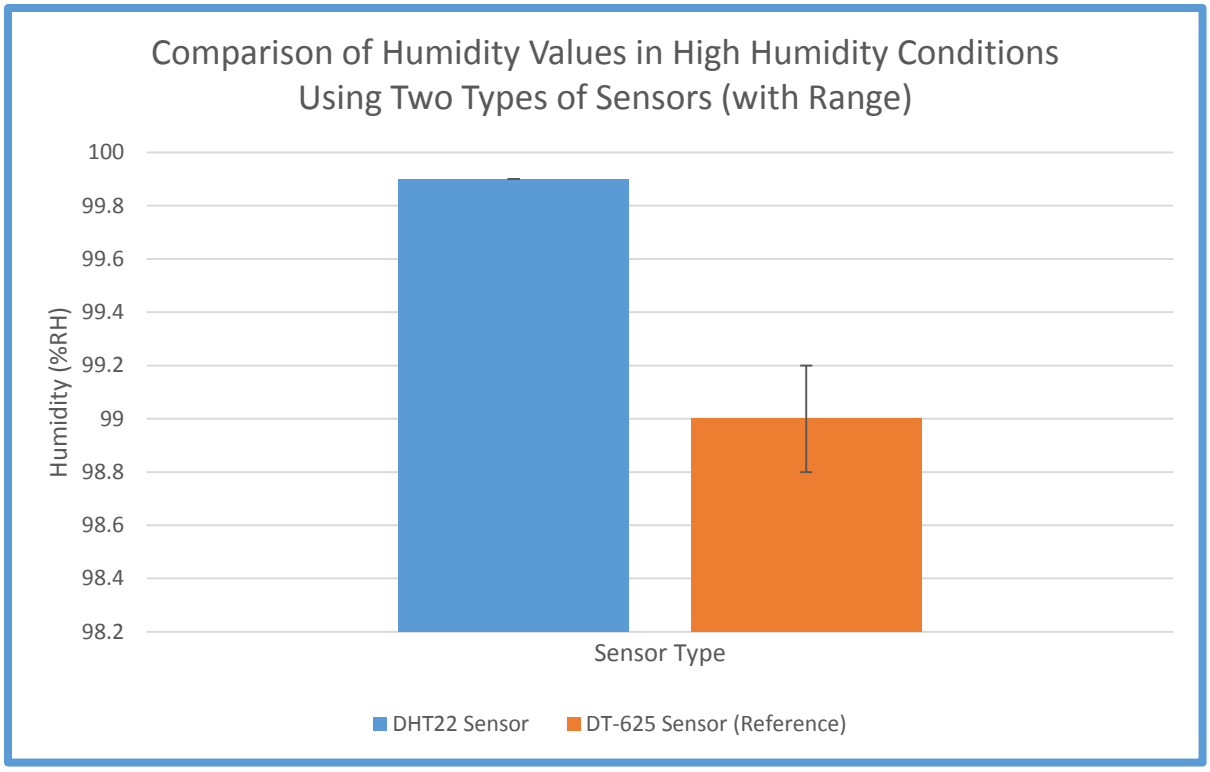


Figure 3-42: Comparison of humidity values measured using two separate sensors, in high humidity conditions. The bars on either set of data represent the range of data values obtained.

From Figure 3-42, the range of the DHT22 sensor was less than that of the reference sensor. In order to determine whether the difference between these two sensors is significant, the paired sample t-test shown in Table 3-36 was conducted.

	DHT22 Sensor	DT-625 Sensor
Mean	99.9	99
Variance	$3.03*10^{-28}$	0.04
Observations	3	3
Pearson Correlation	0	
Hypothesized Mean Difference	0	

	DHT22 Sensor	DT-625 Sensor
df	2	
t Stat	7.794229	
P(T<=t) one-tail	0.008033	
t Critical one-tail	2.919986	
P(T<=t) two-tail	0.016065	
t Critical two-tail	4.302653	

Table 3-36: Paired sample t-test results for both sensors under high humidity conditions. The significance value of a two-tailed test has been highlighted in yellow.

The P value for a two tailed test in high humidity conditions was approximately 0.016. As this value is less than 0.05, so the difference between these two sensors was statistically significant under high humidity conditions. This is important to consider as the humidity is expected to be high inside the blanket. Therefore, with greater funding anticipated in future work, the sensor selection can be much more comprehensive and could involve a more accurate and repeatable reference sensor. Due to cost considerations, the student needed to compromise on this in order to use sensors which were readily accessible.

It was also possible to calculate the percentage error of the DHT22 sensor in these conditions, assuming that the baseline data was the output from the DT-625 sensor. This calculation can be seen below.

$$Percentage\ Error\ (\%) = \frac{|99.9 - 99|}{99} * 100 = 0.00909 * 100 = \ 0.91\%$$

Therefore, the percentage error of the DHT22 humidity sensor under high humidity conditions was found to be 0.91%. Considering the fact that one of the humidity sensors was placed inside the warming blanket (in a humid environment), this result was positive.

The data obtained in typical conditions has also been depicted in Figure 3-43.

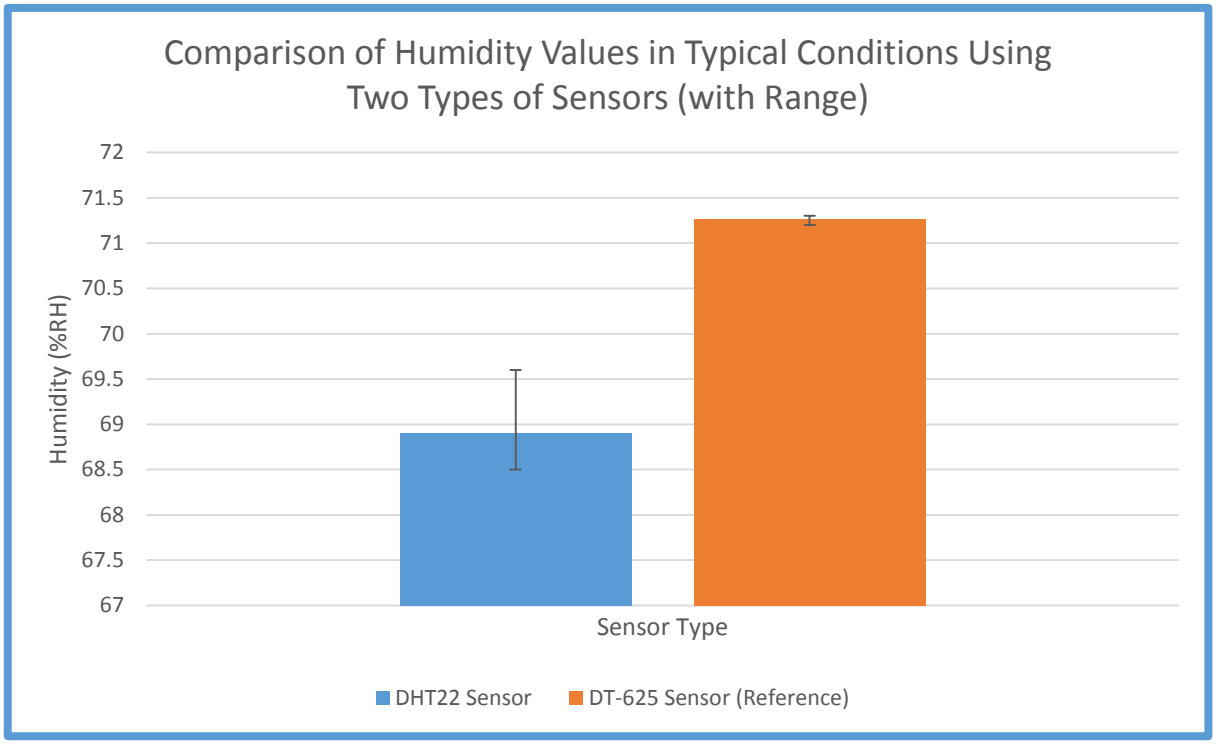


Figure 3-43: Comparison of humidity values measured using two separate sensors, under typical conditions. The bars on either set of data represent the range of data values obtained.

Figure 3-43 differs from Figure 3-42 as the range in this case was much greater for the DHT22 sensor than the DT-625 sensor. In order to determine whether there was a statistical significance between the humidity values shown in Figure 3-43, a paired sample t-test was conducted in Excel. The results of this test are shown in Table 3-37.

	DHT22 Sensor	DT-625 Sensor
Mean	68.9	71.26667
Variance	0.37	0.003333
Observations	3	3
Pearson Correlation	0.569495	
Hypothesized Mean Difference	0	
df	2	

	DHT22 Sensor	DT-625 Sensor
t Stat	-7.1	
P(T<=t) one-tail	0.009633	
t Critical one-tail	2.919986	
P(T<=t) two-tail	0.019266	
t Critical two-tail	4.302653	

Table 3-37: Results of the paired sample t-test under typical conditions. The significance value for a two-tailed test has been highlighted in yellow.

From Table 3-37, the P value of the two tailed test was approximately 0.019. As this value was less than 0.05, the difference between these two sensors under typical conditions was significant.

It was also possible to determine the percentage error of the DHT22 sensor indoors under typical conditions, as follows.

$$Percentage\ Error\ (\%) = \frac{|68.9 - 71.27|}{71.27} * 100 = 0.03325 * 100 = \mathbf{3.33\%}$$

Therefore, the percentage error of the DHT22 humidity sensor under typical conditions was found to be 3.33%. While higher than the percentage error under humid conditions, this error was still deemed reasonable provided the non-critical nature of the humidity data. Therefore, the DHT22 humidity sensor was proven to perform adequately under the required conditions, and formed an integral part of the final design.

However, with an increased budget in future, a more accurate humidity sensor would be recommended, such that the difference between the two sensors was no longer significant in typical and humid conditions.

4.8 BLANKET CONSTRUCTION

4.8.1 SENSOR POSITIONING

The positioning of the sensors and heating elements was an important consideration in this thesis. The temperature sensors were positioned in the mid-thoracic area as shown in Figure 3-44 due to the lack of heat flux in this region. Two of the three humidity sensors were positioned in the center of the warming blanket, and a means of sensing the humidity in the torso portion of the blanket.

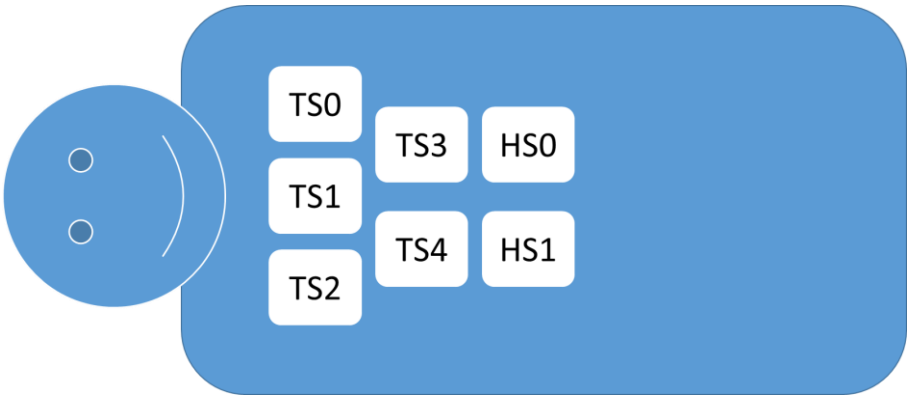


Figure 3-44: Layout of temperature and humidity sensors on the bottom layer of the warming blanket (inside layer).

The five heating elements were positioned evenly on the top layer of the warming blanket, as shown in Figure 3-45.

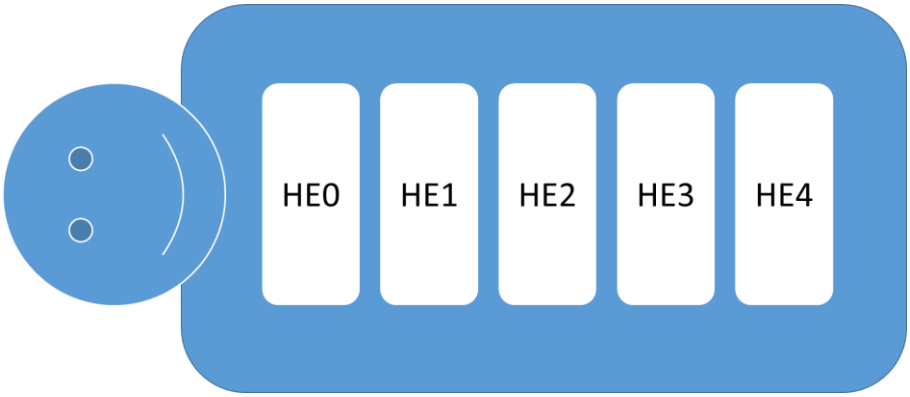


Figure 3-45: Layout of heating elements on the top layer of the warming blanket (inside layer).

There was also one humidity sensor placed on the top layer of the warming blanket to measure the humidity of the outside environment. This is depicted in Figure 3-46.

Figure 3-46: Layout of the humidity sensor on the top layer of the warming blanket (outside layer).

Lastly, the pulse sensor was to be attached to the finger or ear of the neonate, depending on the individual neonate in question.

4.8.2 PRINTED CIRCUIT BOARD

A Printed Circuit Board (PCB) was designed in order to most effectively utilise certain pins from the Bluno Mega microcontroller (namely, the GND, 5V and data pins). By designing a complementary daughter board to attach to the top of the Bluno board, the pins could extend to a female header, meaning that all components could be easily attached to the board. In addition to this, the five MOSFETs discussed previously were also included in the daughter board design.

This design of this PCB was carried out using Altium software. There were numerous considerations during this process, and these are detailed in Table 3-38.

Consideration	Explanation
Position of male headers on daughter board	The male headers enabled the attachment of the daughter board to the Arduino microcontroller, as they could attach to the female headers on the microcontroller. For this reason, the position of the male headers needed to correlate exactly to the position of the female headers on the Arduino. Furthermore, it was these male headers than provided access to pins on the Arduino from the daughter board.
Track width	The width of the track connecting any components needed to be sufficient in order to sustain the required current without causing damage to the daughter board. For example, around 3A was required to power the five heating elements, and so the track width needed to be sufficient in order to carry this current. This consideration was particularly significant due to the incorporation of multiple sensors, each having different power requirements and, for this reason, the determination of appropriate track width will be discussed in more depth in subsequent sections of the thesis.
Minimal track cross-over	In order to maintain the aesthetics of the PCB, and maintain signal integrity, it was important to produce a design with minimal cross-over of wires.
Use of Nets in schematic	Nets were used in the schematic as a method of connecting two pins without having physical wires running across the schematic diagram. While this did not affect the functionality of the schematic in terms of its ability to pair with the PCB diagram, it did provide a much simpler and more intuitive design format.

Consideration	Explanation
Auto-routing vs. manual routing	While convenient in theory, automatic routing was found to be more time consuming as many of the routed wires need to be re-drawn in order to produce a more intuitive PCB design. In addition to this, automatic routing did not incorporate the use of vias or similar techniques to simplify the design. For this reason, manual routing was used in this thesis as it was more efficient, and tended to produce a more aesthetically pleasing design.
Dimensions of daughter board	The daughter board needed to be designed such that it would fit neatly onto the microcontroller with minimal overhang. While the dimensions were flexible provided the male headers aligned with the female headers on the microcontroller, creating a daughter board with size similar to that of the microcontroller ensured an aesthetic design.
Use of vias	A via is used in the PCB design to move from the top layer to the bottom, and vice versa. This technique enabled the connection of wires which would otherwise not be possible because of overlap.
Use of address pins in temperature sensors	The temperature sensors used in the design were based on the I2C protocol and, as such, the five sensors were able to reference the same SDA and SCL pins, provided each sensor had a different address. This address designation was achieved using the three address pins on each sensor (A0, A1 and A2).

Consideration	Explanation
Ratio of routing on top layer to bottom layer	It is common protocol in engineering designs to have as much routing as possible on the top layer, with minimal routing on the bottom layer.
Angle of wire connecting two components	As a convention, the angle of the wire connecting two components should be greater than, or equal to, 90 degrees.
Ground plane on bottom layer	The incorporation of a ground plane on the bottom layer of the PCB ensured that it was possible to ground a component regardless of its position on the PCB. This meant that a lot of wiring was removed by also removing the need to connect all components to one ground pin. Rather, a via was used in these situations to connect a component to GND without extensive wiring.

Table 3-38: Outline of the various considerations when designing the PCB.

5.7.1.1 DETERMINATION OF TRACK WIDTH

There were five heating elements used, each with a required current input of 300mA. Therefore, the current required to power all five heating elements was around 3A, and this required a thicker wire than that used to connect the other components in the design (such as the sensors and the headers).

For this reason, the literature was consulted to determine the required track width for this application. The blue line shown in Figure 3-47 was used to determine the track width required to carry a current of up to 3A.

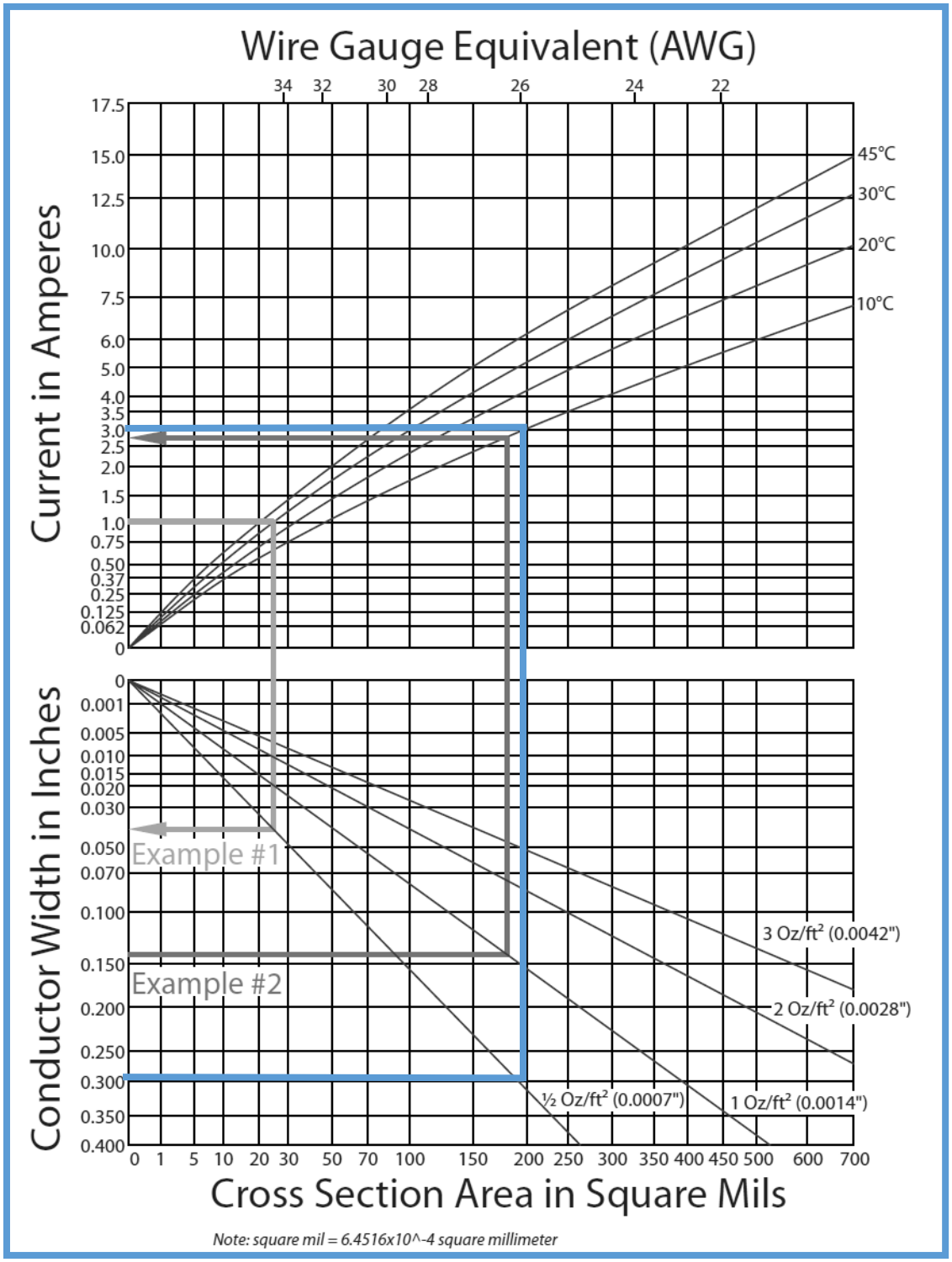


Figure 3-47: Chart used to determine adequate track width for the heating elements (*Sizing a trace on a PCB to carry 2.5 amps*, no date).

5.7.1.2 PCB REVISIONS

Table 3-39 details the gradual development of the final PCB design.

Revision	Edits and Description
1	Separate headers were included for 5V Power, GND, data pins, etc. Board shape was outlined without locked dimensions. Specific MOSFET footprint was included in the design.
2	Changed categorisation of headers from function to component i.e. removed headers that were connected only to GND pins, or 5V pins, and added new headers designated to each component individually. Included vias so that tracks did not cross each other, but rather moved from the top layer to the bottom layer when a cross-over did occur.

Revision	Edits and Description
3	Altered layout of components. Added through-hole resistors to schematic and PCB for protection of MOSFET circuit. Thickened wire tracks that would carry 3A (from 1mm to 3mm). Redefined origin in Altium, so that it aligned with the board.
4	Added border to mechanical layer. Added power socket footprint to schematic and PCB design. Removed through-hole resistors from PCB and schematic, and replaced with smaller surface-mount resistors. Set coordinates of male headers to correlate with the female headers on the Bluno Mega 2560. Ensured all headers were rotated correctly, so that the pins of the male headers would insert into the correct female header sockets on the microcontroller.

Revision	Edits and Description
5	Added the ground plane on the second layer of the PCB, for the vias to connect to. Added name to top overlay. Ensured all connections were made between components.

Table 3-39: Outline of revisions to the PCB, and the corresponding alterations that occurred at each revision.

A two-dimensional view of the final PCB design is shown in Figure 3-48.

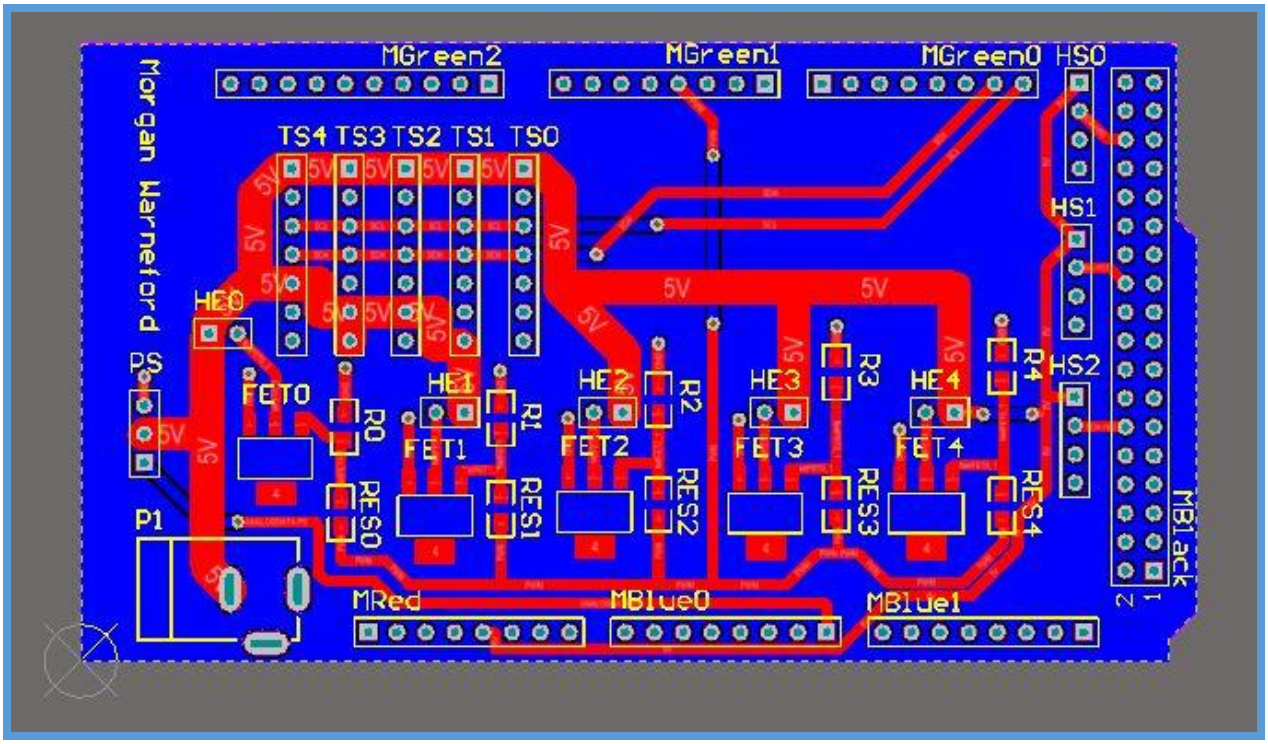


Figure 3-48: Two dimensional PCB design, where the red tracks indicate the tracks of copper on the top layer of the board, and the blue tracks indicate the tracks of copper on the bottom layer of the board. The yellow text is part of the overlay.

As can be seen in Figure 3-48, seven male headers were included on the daughter board to plug into the female headers on the Bluno Mega 2560 motherboard. The final PCB, with all headers attached can be seen in Figure 3-49.

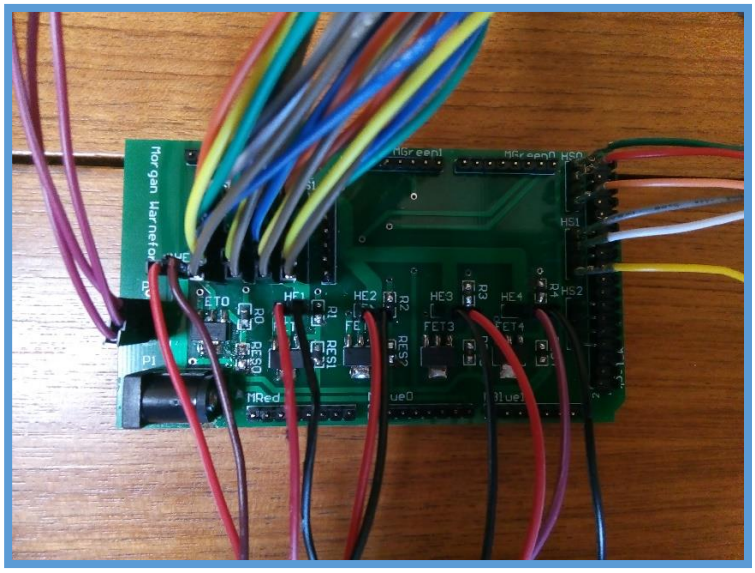


Figure 3-49: Final PCB with all headers, resistors and MOSFETs attached.

In Figure 3-48, the names of each component or footprint were labelled. For simplicity, these names and their corresponding components have been included in Table 3-40.

Name on PCB	Corresponding Footprint/Component
MRed	Male header to plug into red female header on Bluno Mega 2560 motherboard
MBlue0	Male header to plug first blue female header on Bluno Mega 2560 motherboard
MBlue1	Male header to plug into blue female header on Bluno Mega 2560 motherboard
MBlack	Male header to plug into black female header on Bluno Mega 2560 motherboard
MGreen0	Male header to plug into green female header on Bluno Mega 2560 motherboard
MGreen1	Male header to plug into green female header on Bluno Mega 2560 motherboard
MGreen2	Male header to plug into green female header on Bluno Mega 2560 motherboard
P1	Power Socket
PS	Pulse Sensor

Name on PCB	Corresponding Footprint/Component
HS0	DHT22 Humidity Sensor 0
HS1	DHT22 Humidity Sensor 1
HS2	DHT22 Humidity Sensor 2
HE0	Heating Element 0
HE1	Heating Element 1
HE2	Heating Element 2
HE3	Heating Element 3
HE4	Heating Element 4
TS0	MCP9808 Temperature Sensor 0

Name on PCB	Corresponding Footprint/Component
TS1	MCP9808 Temperature Sensor 1
TS2	MCP9808 Temperature Sensor 2
TS3	MCP9808 Temperature Sensor 3
TS4	MCP9808 Temperature Sensor 4
FET0	N Channel Surface Mount MOSFET Transistor 0
FET1	N Channel Surface Mount MOSFET Transistor 1
FET2	N Channel Surface Mount MOSFET Transistor 2
FET3	N Channel Surface Mount MOSFET Transistor 3
FET4	N Channel Surface Mount MOSFET Transistor 4

Name on PCB	Corresponding Footprint/Component
R0	100 Ohm Surface Mount Resistor 0
R1	100 Ohm Surface Mount Resistor 1
R2	100 Ohm Surface Mount Resistor 2
R3	100 Ohm Surface Mount Resistor 3
R4	100 Ohm Surface Mount Resistor 4
RES0	100K Ohm Surface Mount Resistor 0
RES1	100K Ohm Surface Mount Resistor 1
RES2	100K Ohm Surface Mount Resistor 2
RES3	100K Ohm Surface Mount Resistor 3

Name on PCB	Corresponding Footprint/Component
RES4	100K Ohm Surface Mount Resistor 4

Table 3-40: Footprints associated with the names shown on the PCB design.

5.7.1.1 PCB CHALLENGES

Altium was a new piece of software that required a fast and steep learning curve in order to design the PCB shown in Figure 3-49. This required a high level of organization to ensure that sufficient time was allocated to become familiar with the software, and to complete the desired objective. This learning curve also necessitated the utilization of technical staff assistance, in order to meet deadlines and complete this portion of the project as efficiently as possible.

Another challenge experienced with the PCB was related to the power socket highlighted in Figure 3-50.

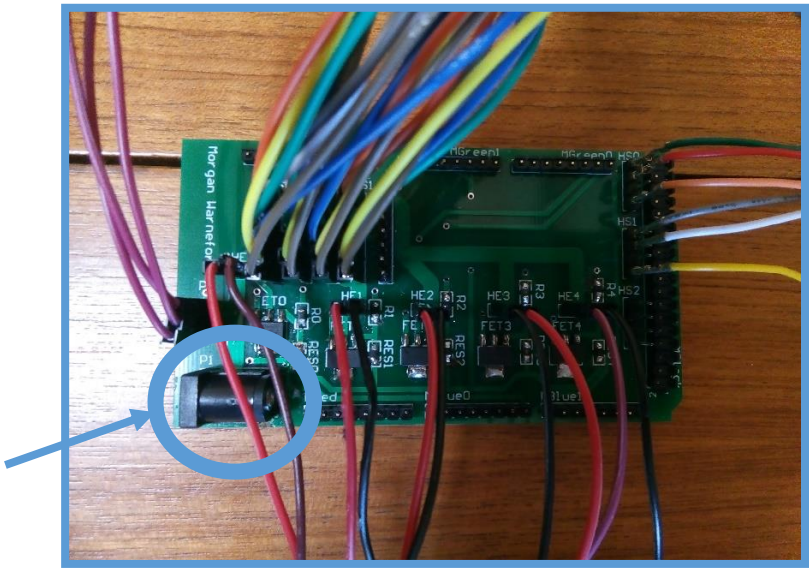


Figure 3-50: Final PCB with the power socket highlighted.

Initially, the power socket supplied by the University was incorrect, in that it did not match the provided footprint from the technical staff. This was soon realised and another power socket was obtained. However, this socket did not match the footprint provided by the technical staff either. This challenge took time to realise, as the heating elements were not being powered and, at the time, this may have been due to a myriad of software or hardware errors. It was eventually realised that the layout of the power socket footprint was that shown in Figure 3-51.

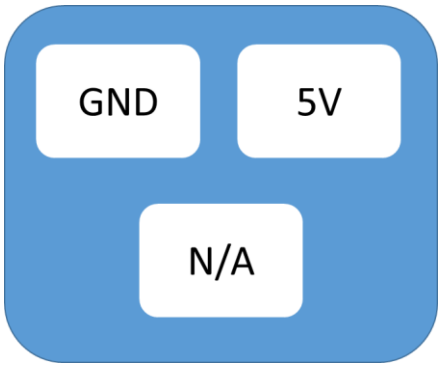


Figure 3-51: Actual footprint of the power socket.

However, the PCB was designed with the footprint shown in Figure 3-52.

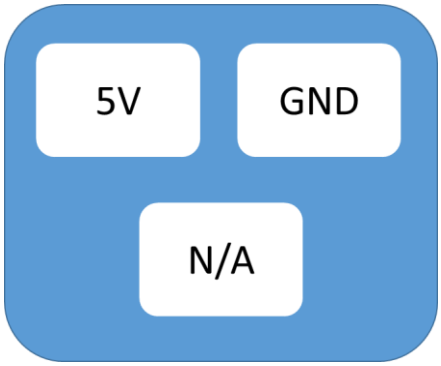


Figure 3-52: Incorrect footprint of the power socket.

The difference between Figure 3-51 and Figure 3-52 is the position of the 5V pin, and the GND pin. The footprint provided by the technical staff had these pins in the opposite position to the actual power socket. This was overcome by reversing the position of the power socket on the PCB.

4.9 GRAPHICAL USER INTERFACE (GUI)

A simple graphical user interface (GUI) was created using the Processing IDE in order to display vital neonatal information such as heart rate and body temperature. The aim of this interface was to ensure ease of use and access to information, regardless of the technical skill of the user. This was an important consideration as this system could eventually be used by nurses or parents and it was not reasonable to assume a certain level of technical knowledge.

In addition to this, an aesthetically pleasing interface can make the information more perceptible. For example, the effective use of color can mean that there is less dependence on numerical understanding, and more dependence on visual perception of information. In other words, the user is not required to read and understand the numerical values for humidity and temperature, but rather understand that a green level indicates a healthy value, and red indicates a dangerous, or risky, value.

4.9.1 SERIAL DATA COMMUNICATION

This section of the thesis will detail the method of communication between the Arduino and Processing IDEs. This was important as the sensor data was sent to the Arduino IDE, but displayed to the user using the Processing IDE. This was ultimately achieved through the use of serial communication protocols.

Serial communication involves the transmission or reception of one bit of data for every clock pulse. The rate at which data is sent over a serial line is referred to as the baud rate (measured in bits-per second, or bps). Both the receiving and transmitting device need to run at the same baud rate in order for serial data communication to be successful. A higher baud rate indicates faster data transmission, but usually does not exceed 115200 bps. In the case of this thesis, a typical baud rate of 9600 bps was used.

The data transmitted using serial communication is separated into several bits, based on the configuration shown in Figure 3-53.

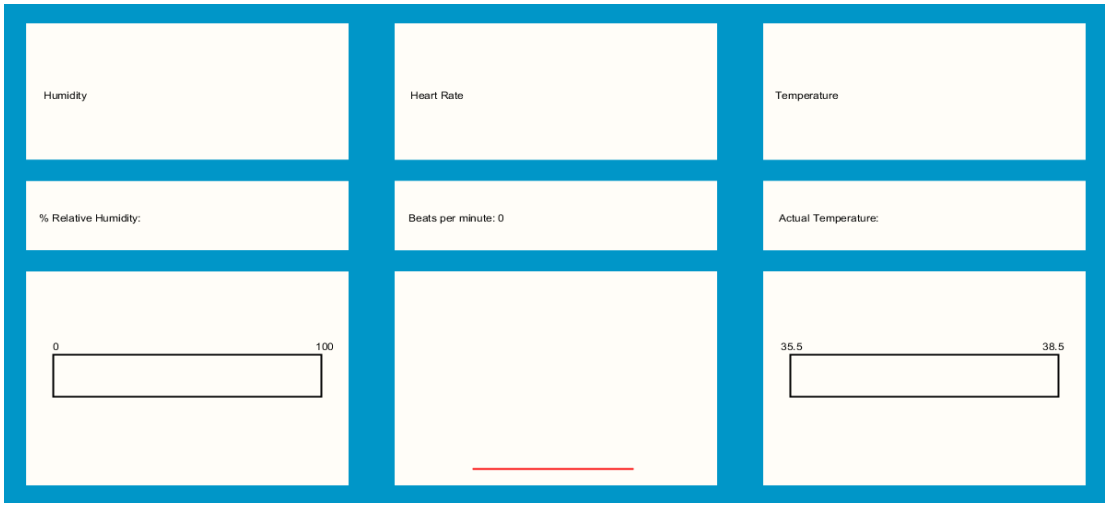
Figure 3-53: Separation of bits communicated using the serial communication protocol (*Serial Communication*, no date).

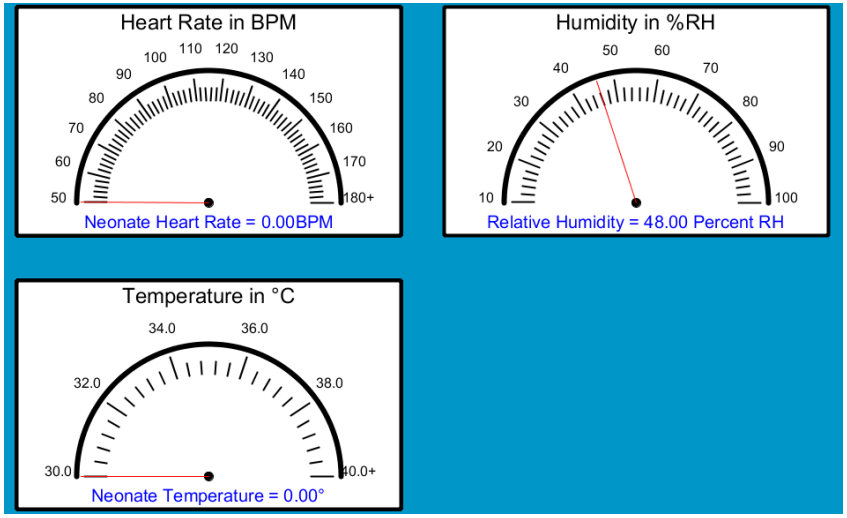
The endianness of the data is determined by either of the serial devices. This means that the protocol of data order must be pre-determined, so data is either sent with the most significant bit first, or last. Unless otherwise stated, the least significant data bit is usually sent first.

The parity bits provide a low-level of error scanning. All data bits are added together and the result is either even or odd. This functionality will not be used in the thesis as it is not commonly used in industry. Furthermore, the data transfer is slowed when the parity bits are utilised, and this is not ideal in a high-pressure sensing application. Lastly, incorrect data transferral from the sensor (for example, if the sensor reported a neonate temperature of 0°C instead of 37°C) would have little effect on the PID algorithm, assuming normal data transfer returns after that one error. In addition to this, a parity bit only detects 50% of errors so these bits may not be able to detect errors that do occur.

4.9.2 GUI REVISIONS

The first GUI revision involved an attempt to use software that would connect PC to Arduino microcontroller (Blynk software). However, the full functionality of the software required additional costs, so this option was disregarded and the second version of the GUI was begun. Subsequent revisions are shown in Table 3-41.

Revision Number	Edits and Description	Image
2	The Processing IDE was used to create a simple three-column GUI. One column displayed pulse sensor data in the form of BPM values and a heart rate waveform. Another column was used to display the temperature sensor information, in numerical form, as well as on a visual bar display. The last column was used to display the humidity data in a similar manner to the temperature data.	

Revision Number	Edits and Description	Image
3	The Processing IDE was used again in order to produce an alternative design, using numerical values and gauge images in order to depict the sensor data.	

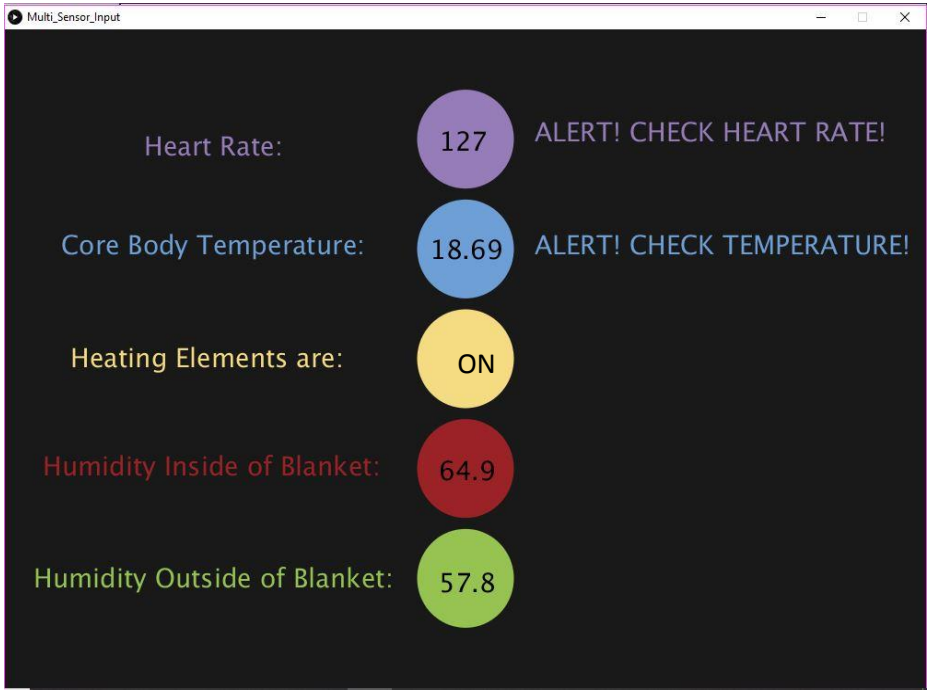
Revision Number	Edits and Description	Image
4	The Processing IDE was used to produce an alternative design that showed only alerts and numerical values for simplicity. The intent behind this design was to eliminate any subjectivity in interpretation of the results. For example, it was decided that a heart rate waveform could lead to increased confusion for some, with little to no benefit. At this revision, the alarm text was displayed, but no audio alarm had yet been added.	

Table 3-41: Details of user interface revisions.

In the final version of the GUI, audio alarms were added when the sensor data fell into the 'dangerous' range. The colour of the humidity data inside the blanket was also changed from red to pink. This was a decision made based on the fact that red is often attributed to danger or errors, and this was not always the case. This final change is depicted in Figure 3-54.

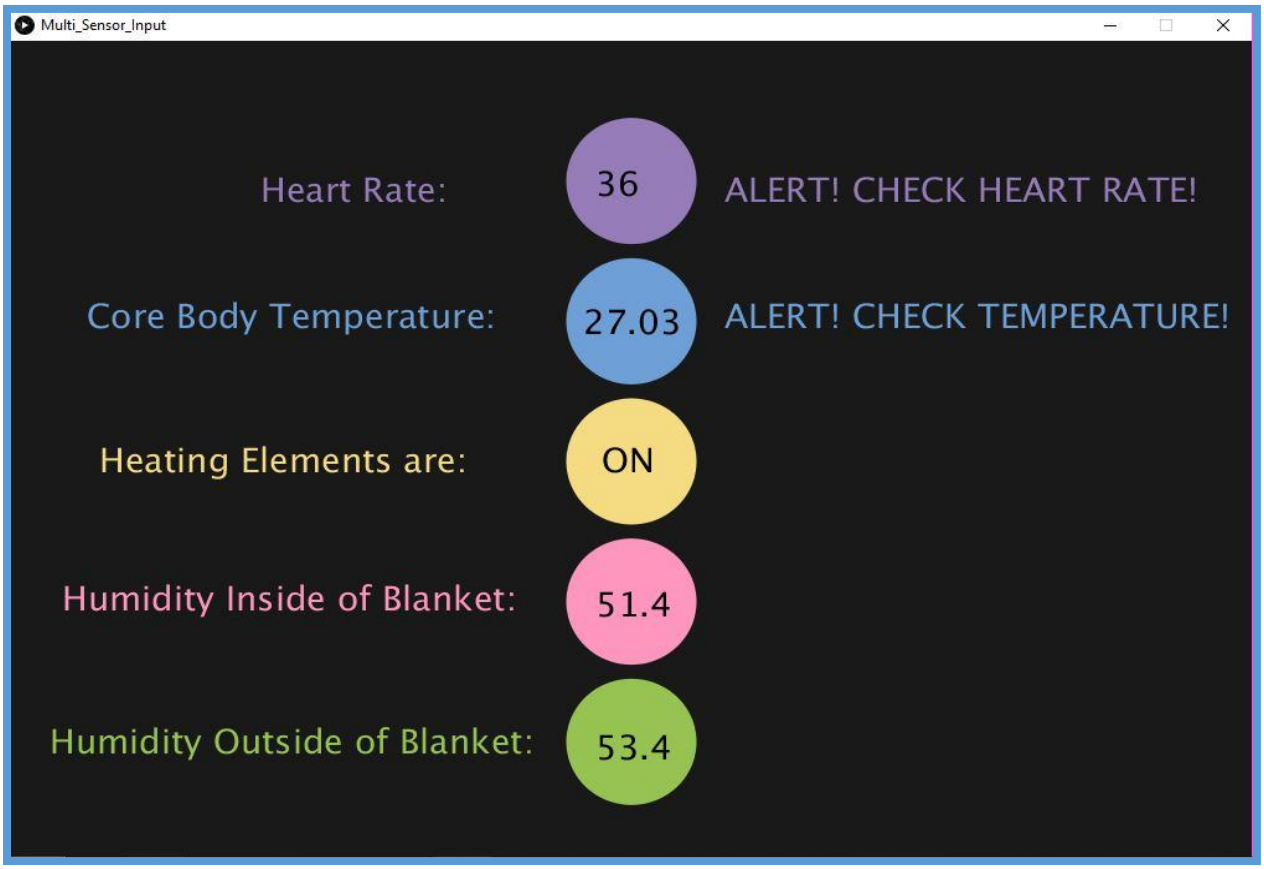


Figure 3-54: Final version of the GUI.

4.9.3 ALARMS

A document was sourced which outlined the various types of alarm signals used in clinical applications in accordance with ISO/IEC 60601-1-8. The alarm tones relevant to the thesis are represented in Table 3-42.

Category	Medium	High Alarm*	Notes

General	C4 C4 C4	C4 C4 C4 – C4 C4	Fixed pitch
Cardiovascular	C4 E4 G4	C4 E4 G4 – G4 G5	N/A
Temperature	C4 D4 E4	C4 D4 E4 – F4 G4	Pitch slowly rises
Power Failure	C5 C4 C4	C5 C4 C4 – C5 C4	Pitch falls
General Information	E4 C4		Slow, door-bell like

Table 3-42: Relevant alarm signals in a clinical setting. Note that all high alarm signals are depicted by a (*) and all high alarm bursts comprise of the same 5 note pattern played twice, separated by 2 seconds of silence - 10 pulses in all.

4.9.4 ERRORS ENCOUNTERED

The errors encountered during the software development phase of the project are detailed in Table 3-43. Consistent documentation of these errors ensured that errors could be efficiently noticed and rectified if they had occurred previously.

Error Text or Description	Error Cause	Solution
Error opening serial port COM4: Port busy	Trying to run code in Processing IDE while Serial Monitor was open in Arduino IDE.	Close Serial Monitor in Arduino IDE.
Heating elements not warming up	Referenced PWM pin 4 in code instead of pin 3.	Altered the code so that the correct pin was being referenced.
Nothing written to serial port despite writing to it in code.	One of the temperature sensors (TS4) had a pin that had fallen out. Therefore, the code did not make it past the initialization of this temperature sensor.	Short-term: reinserted the male pin into the female header. Long-term: added code to continue running remainder of program even if one sensor was not recognized.
avrdude: verification error; content mismatch	N/A	Unplugged Arduino and plugged into the computer once more.
Board not recognized through USB port.	Power socket pins were in the reverse position to the footprint on the PCB.	Power socket was rotated 180 degrees and re-soldered onto the daughter board.
Temperature sensors outputting values of -0.06.	SDA and SCL pins on the daughter board did not contact the Arduino sufficiently i.e. there was no electronic connection.	Resoldered the SDA and SCL pins to ensure sufficient contact between the Arduino and the daughter board.

Error Text or Description	Error Cause	Solution
Humidity sensor data output was not a number (NAN).	The wrong pin was referenced as the data pin. The microcontroller was attempting to get sensor data from pin 32 when pin 34 was being used.	The correct pin was referenced in the code.
Microcontroller kept disconnecting from the PC.	Micro USB port on the Bluno Mega 2560 became disconnected.	Resoldered the micro USB port.

Table 3-43: Outline of errors encountered during the software creation phase, with corresponding solutions provided.

5 ANALYSIS AND DISCUSSION

5.1 ABILITY OF DESIGN TO SATISFY SPECIFICATIONS

The design specifications outlined at the beginning of this thesis formed the basis for the designed product, and are a significant metric when deducing the success of the project overall. Therefore, the specifications and achieved metrics have been detailed in Table 5-1. Note that any specifications not met by the design have been highlighted, and a justification for any variance has been included.

Metric	Specified Value	Actual Value	Specification Met (Y/N)	Justification for Variance
Steps required to operate	**1**	**2**	**N**	The need for two separate IDEs meant that each one needed to be started individually before data acquisition and display could occur. In future, this could be improved such that these two steps were combined into one.
Minimum number of temperature sensors	3	5	Y	N/A
Minimum number of pulse sensors	1	1	Y	N/A
Minimum number of humidity sensors	2	2	Y	N/A

Metric	Specified Value	Actual Value	Specification Met (Y/N)	Justification for Variance
Minimum number of pressure sensors	1	0	N	This is an addition that will be included in future versions of the design. It was excluded from this thesis due to time constraints and the risk of scope creep, but would be a beneficial addition in the future.
Temperature sampling rate	≤ 1 second	4Hz	Y	N/A
Pulse sampling rate	≤ 1 second	500Hz	Y	N/A
Humidity sampling rate	≤ 10 seconds	0.5Hz (once every 2 seconds) (*DHT22 temperature-humidity sensor + extras*, no date)	Y	N/A
Temperature sensor resolution	≤ 0.1 °C	0.0625°C (*MCP9808 High Accuracy I2C Temperature Sensor Breakout Board*, no date)	Y	N/A

Metric	Specified Value	Actual Value	Specification Met (Y/N)	Justification for Variance
Pulse sensor resolution	≤ 2 bpm	1 bpm	Y	N/A
Humidity sensor resolution	≤ 5%	0.1%	Y	N/A
Temperature sensor accuracy	±0.1 °C (maximum)	±0.25°C (*MCP9808 High Accuracy I2C Temperature Sensor Breakout Board*, no date)	N	The temperature sensor in the design was chosen due to the strict budget requirements, and the prototyping nature of the project meant that functionality could be tested with a higher accuracy sensor used in a final product.
Pulse sensor accuracy	±5 bpm (maximum)	Not provided	N/A	N/A

Metric	Specified Value	Actual Value	Specification Met (Y/N)	Justification for Variance
Humidity sensor accuracy	±10% (maximum)	2-5% (*DHT22 temperature-humidity sensor + extras*, no date)	Y	N/A
Production cost	< $600	$344.42	Y	N/A
Warm-up time	≤10 minutes	10 minutes	Y	N/A

Table 5-1: Outline of initial specifications, and actual achieved metrics.

5.2 SUGGESTED CONSIDERATIONS

The proposal seminar conducted as a portion of the required coursework led to the consideration of multiple factors including the effect of swaddling on sensor data, and on system performance, as well as the use of the blanket in a home environment. These considerations will be discussed in the subsequent sections of this thesis.

5.2.1 EFFECT OF SWADDLING

Swaddling is a technique commonly used with babies of varying ages and weights, and has been found to lead to reduced arousal and longer periods of sleep (van Sleuwen *et al.*, 2007). In addition to this, swaddled infants have been found to have "improved neuromuscular development, less physiologic distress, better motor organization, and more self-regulatory ability" (van Sleuwen *et al.*, 2007).

The most relevant consequence of swaddling to the thesis, however, is its role in thermoregulation of preterm and low weight neonates. Studies have shown that core neonatal body temperature can increase by 0.2 degrees Celsius when swaddled (Short, 1998), but hyperthermia can result when this increased body temperature is coupled with a covered head, or an infection. Therefore, considering the core body temperature of the neonate is likely to increase if swaddled, the PID algorithm discussed will directly compensate for the increased body temperature, and the heating elements will either operate with a decreased duty cycle, or turn off accordingly.

5.2.2 INCORPORATION WITH HOSPITAL SYSTEMS

Another future consideration for the project involves the incorporation of this warming blanket device with existing systems at the Flinders Medical Centre. This would increase the likelihood of adoption, and also minimize the amount of training required to use the system.

Furthermore, if this device were to be commercially produced, the Flinders Medical Centre would likely design their own hardware platform. This would mean that a new PCB design would incorporate the functionality of both the Bluno Mega and the daughter board from this thesis. While this would

incur high upfront costs due to the PCB design, it would greatly reduce material costs overall, and result in a physically smaller and more reliable system.

5.2.4 USING THE BLANKET IN A HOME ENVIRONMENT

At this stage, the proposed neonatal warming blanket has been developed as a prototype, and further testing is required before a commercialization process was to be undertaken, and hospital and home use could be achieved. However, a comprehensive feasibility study is the first stage in the commercialization process, and this has largely been completed in the thesis. A functioning, tested prototype has been produced and has been found to be technically, socially and financially feasible.

5.3 LIMITATIONS

A major limitation to this research became evident in the preliminary planning stages. As the selected target market consisted of neonates and their families, this meant that testing would eventually need to be carried out on these neonates to ensure that adequate temperature management was achieved when using the warming blanket. However, given the extent of a proposed ethics application for that testing, the approval of such an application was not feasible in the given time to complete the project. This is a limitation to the results discussed in this thesis, of course, because the ideal test would involve the interaction of the designed warming blanket with several real life neonates requiring assistance in temperature management.

Another limitation is one experienced in most, if not all projects, and that was the budget. The budget allocated to this project was $600 (AUD). Therefore, resource usage was considered to a high degree by both the student and supervisor, to ensure that purchases made were necessary and contributed to the success of the project overall. Similarly, the time allocated to complete the project served as another limitation.

In addition to this, the initial stages of software implementation led to some limitations, as the IDE had limited language capabilities, but was efficient for the creation of an attractive and functional GUI. The IDE did not support C++, which was the language used to devise the initial sensor code during the independent testing of the sensors.

5.4 POTENTIAL IMPROVEMENTS

In terms of perhaps more superficial work to be completed in the future, the aesthetics of the design could be improved. This includes creating a housing method for the Arduino, heating element, and portable battery. Furthermore, an LCD or OLED screen could be incorporated to complement the interaction of the system with a mobile phone or computer.

Similarly, the portability of the device could be improved by also powering the Arduino by a portable battery, rather than using a microUSB as was the case in this prototype.

In addition to this, more expensive sensors could be used to increase the accuracy of the measurements obtained by the device. This would be a vital step to ensure the safety of the neonates upon testing and eventual adoption. While increased sensor accuracy would likely increase the cost of the blanket, it is a necessary compromise if it will provide further reassurance of the safety of the user.

5.5 FUTURE WORK

This section of the thesis will expand on some areas for additional work to be completed on the project. These largely relate to additional and formal testing, which necessitates an approved ethics application through the relevant ethics board. Furthermore, material selection is a significant consideration for any design to be used in a hospital setting, so this will also be discussed. Lastly, the treatment of jaundice is a reasonable inclusion in the design, and will be detailed in terms of its importance and potential benefits.

5.5.1 INCORPORATION OF PRESSURE SENSORS

This is one aspect of the project which was involved in the initial design and later removed due to time constraints, as well as the fact that this sensor was arguably the least crucial to the success of the warming blanket when compared to the pulse, humidity and temperature sensors. However, were the project to be continued in the future, this added functionality could prove beneficial in the prevention of pressure sores in the neonatal sector. Appendix F (Incorporation of a Pressure Sensor) details the preliminary research conducted into pressure sensing technology.

5.5.2 ETHICS APPROVAL

In relation to the ethics approval discussed previously, this would be an important and necessary part of the future of this project. This approval will make it possible to test the neonatal warming blanket on neonates that require assistance in thermoregulation, therefore either confirming or disputing the overall validity of the design.

An ethics application would be submitted to an appropriate ethics committee and, once approval was granted, the testing phase of the project could begin. The relevant ethics board in this case would be the Southern Adelaide Clinical Human Research Ethics Committee (SAHREC) as this committee manages ethics applications for the Flinders Medical Centre (FMC).

5.5.3 SIMULATION TESTING

Before testing on real life neonates, however, another future step could be to use a simulation doll which has varying body temperature within the range commonly experienced by premature neonates. This would provide a confirmation or rejection of the design components, based on a neonate's typical response to the blanket, without causing any harm to real life infants. This would also increase the likelihood of parents volunteering to take part in testing on real neonates, as the design would have been verified through an alternative method first.

5.5.4 MATERIAL SELECTION

Material selection is an important part of any design process. With regards to the hospital environment specifically, material composition relates to the prevention of bacteria, and the minimization of vibration which can have lasting effects on neonatal growth and development. Lastly, fabric-embedded LEDs and other indicators could provide useful information to neonatal staff at a glance, and this will also be discussed as a potential development on the current design.

5.8.4.1 THERMOREGULATING MATERIALS

The mechanical properties of the blanket material used for neonatal purposes are significant because the blanket needs to be flexible enough to fully envelop the neonate, while also containing a heating element which are notoriously inflexible in nature. One material discussed in literature is referred to in industry as Phase Change Material (PCM).

A product called MiraCradle uses this material in its design. Conversely to the thesis, this product aims to cool neonates rather than warming them. While this seems counter-intuitive with neonates often requiring warming, induced hypothermia in neonates can be therapeutic in treating asphyxia (oxygen deficiency).

Applicable to this thesis, however, is the ability of PCM to remain within a specified temperature tolerance for a specific time reliably. The material used in the MiraCradle system remains stable at "33-34°C for a period of 72 hours with minimal manual supervision and no requirement of constant electricity supply" (*MiraCradle® - Neonate Cooler*, no date). This ability to function without electricity is

ideal in neonatal applications as any power outage could prove fatal if there were not contingency plans in place. Phase Change Materials store thermal energy, and this energy is released when the phase of the material shifts from liquid to solid or vice versa (*MiraCradle® - Neonate Cooler*, no date).

The Miracradle product is further unique in the way that it utilises a specific type of PCM, called a Form Stable PCM. This type of material ensures that the product maintains its shape and structural integrity despite any phase changes. Not only does this provide further aesthetic and functional benefit, but it also ensures the safety of the user as there is reduced chance of leakage when phase change does occur. In this specific product, multiple Form Stable PCMs are "cascaded", with each of these layers having a different point at which phase change occurs. This creates a sort of automatic temperature controlled unit, without the need for a constant power supply.

A paper by Matahari, Putra, Ariantara, Amin and Prawiro (2017) also investigated the efficacy of PCMs in neonatal applications. The end result of this study was that "the most efficient mass of PCM is 3kg, which has 2.45 hours of running time for maintaining temperature of incubator in range of 32-36°C" (Matahari *et al.*, 2017). Therefore, PCM could be utilised in future versions of the blanket designed in this thesis. PCM meets the requirements of the blankets as it is lightweight, while also maximising the amount of time that heat is maintained.

5.8.4.2 BACTERIA PREVENTION

It is regularly mentioned in this thesis that temperature regulation is vital for the survival of a neonate because some may have significant difficulty maintaining their own body temperature. However, temperature regulation is also important to reduce the risk of microbial growth and contamination.

Specifically, a study by De Gaffau et al. (2011) raised the concern that cold spots in a neonatal incubator or bassinet can increase the risk of microbial growth. "In incubators with high average temperature (≥34°C) and relative humidity (≥60% RH) values, the level of microbial contamination was significantly higher at cold spots than at hot spots" (de Goffau *et al.*, 2011). This was found to be caused by the fact that the equilibrium in these positions was optimal for the growth of bacteria. Specifically, De Gaffau et al. (2011) noticed staphylococci were found to (Teufel *et al.*, 2010) prevail in cold areas of the incubator, and this strain of bacteria can lead to late-onset sepsis in neonates. The

same principles apply to neonatal warming blankets. Cold spots in these blankets can result in the same microbial growth.

The degree of this microbial growth depends on the material composition of the neonatal warming blanket. A study by Teufel, et al. (2010) found that cotton, polyamide and polyester materials harboured significantly more bacteria than TENCEL, which harboured more bacteria than polypropylene (Teufel *et al.*, 2010). Figure 5-1 illustrates this trend.

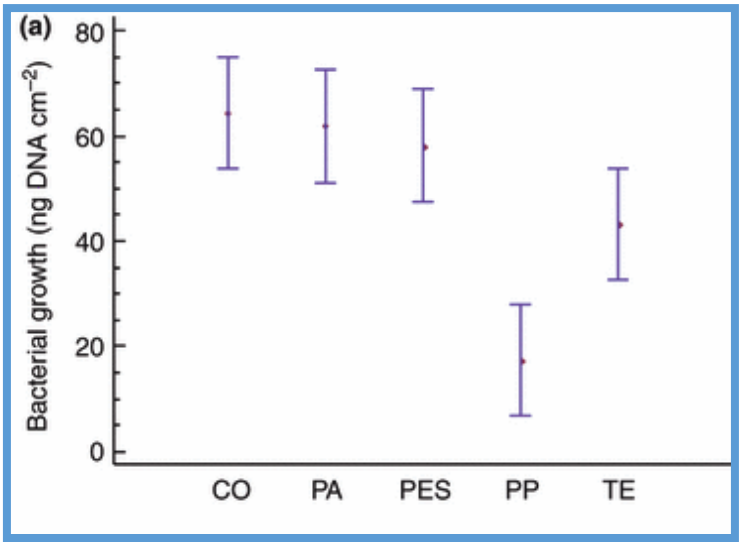


Figure 5-1: Effect of material type on bacterial growth (Teufel et al., 2010).

It is also important to note that natural textiles and materials typically harbour more bacteria than synthetic materials due to their composition of "water, oxygen and nutrients" (Boryo, 2013).

Literature has also shown methods of synthetically creating antimicrobial materials, through "molecularly bonded unconventional technology" (Boryo, 2013). This method involves the attachment of a silane, which kills microbes upon contact. Chitosan is also used to perform antimicrobial actions when used in materials. Chitosan is a natural polymer which forms chemical cross-links when applied to cellulose. These cross-links cause the antimicrobial properties that would be desired in future warming blanket designs. Furthermore, while the adverse effects of synthetic materials are under-represented in literature, natural polymers such as Chitosan have not been identified as a hazard to human health.

5.8.4.3 VIBRATION MINIMISATION

Neonates are considerably less tolerant of external stimuli when compared to adults, with the International Standards Organisation (ISO) stating that vibrations can "have adverse effects on cardiorespiratory function, the peripheral and central nervous system, electroencephalographic activity, body temperature, metabolic and endocrine function, and the gastrointestinal system" (Browning *et al.*, 2008). This issue could be mitigated in future versions of the neonatal blanket design by considering material compositions that could reduce vibrations.

In order to decide which material is best for minimising vibration, it is important to first determine the level of vibration that these neonates are exposed to when transported within the hospital. "Surprisingly, there have been few previous studies investigating the levels of mechanical shock and vibration hazard present during this vehicular transport pathway" (Blaxter *et al.*, 2017). This study found that most of the vibrational energy that neonates were exposed to fell between 5 and 20Hz.

A material called Sorbothane significantly reduces the vibrational effect of travelling within the hospital. Figure 5-2 shows this vibration dampening well, when compared that of alternative material options. In numerical terms, Sorbothane "absorbs up to 94.7% of impact shock" (*Shock Absorbing Material*, 2018).

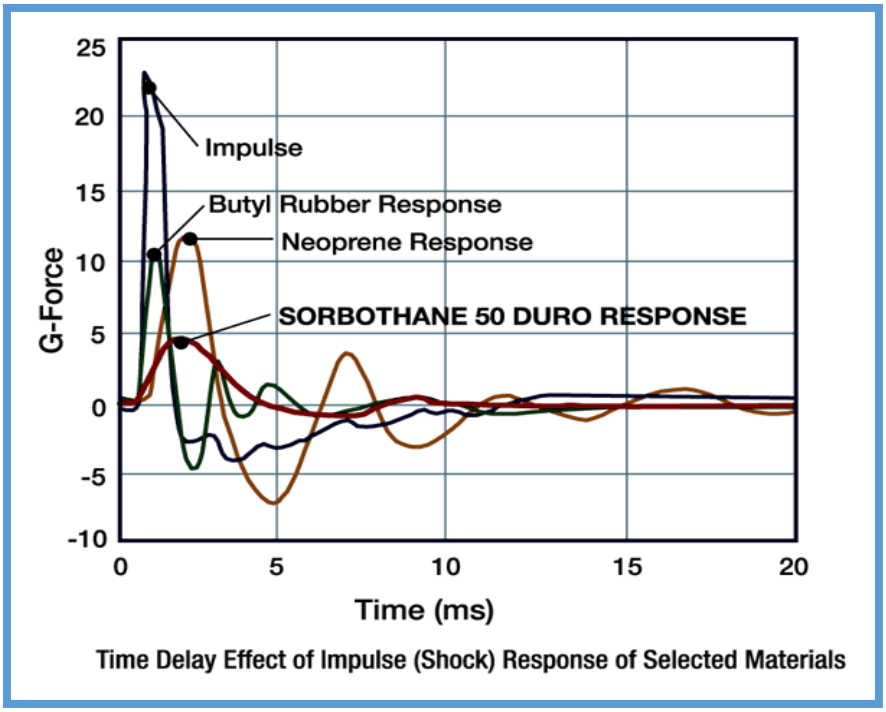


Figure 5-2: Ability of various materials to act as shock and vibration absorbers (Shock Absorbing Material, 2018).

Another material option is called PORON, which is commonly used by clothing organisations in order to reduce shock and vibration during physical activity. While this material is arguably less efficient at reducing shock to the body when compared to Sorbothane, it still "absorbs more than 90% of energy upon repeated impact" (*Rogers Introduces PORON® XRD™ Extreme Impact Protection Material for Individual and Team Sportswear & Equipment - Rogers News*, 2009).

These materials could be included in future neonatal warming blanket designs in order to minimise the effect of vibration on neonatal wellbeing and development.

5.8.4.4 INFORMATION PORTRAYAL

Another interesting addition to the project would be the incorporation of thermo-chromatic material, which would change color depending on the temperature of either the heating elements embedded in the blanket or the temperature of the neonate. This would provide another visual indication of temperature, and as such would make it easier for neonatal staff to determine when this temperature was changed significantly. However, the sensitivity of this material would need to be such that

temperature changes of 0.1° would cause a noticeable color change, and this would require further investigation.

5.5.5 UTILIZATION OF LIGHT

Another area for further investigation relates to the use of light in blanket design, particularly in the area of jaundice treatment.

5.5.5.1 TREATMENT OF JAUNDICE

Jaundice, or hyperbilirubinemia, is a condition inherently common in premature and low birth weight neonates due to deficient liver function. Correlations have been established between the likelihood of jaundice, and the "method of birth, perinatal complications, blood group incompatibilities, birth weight, and method of feeding" (Osborn, Reiff and Bolus, 1984).

Jaundice is typically treated in neonatal situations using phototherapy. Jaundice is caused by increased levels of bilirubin[26], so phototherapy aims to reduce these levels using blue lighting. The neonate's skin is illuminated with this blue light, which it is absorbed by the bilirubin. Bilirubin is then converted into waste products that are more easily removed from the body (Hansen 2017). In terms of optimising the effectiveness of this treatment, the neonate is undressed so that the surface area exposed to blue light is maximised. The eyes of the neonate are also covered because, as mentioned previously, neonates are extremely sensitive to strong stimuli such as bright light.

[26] Bilirubin is a "reddish yellow pigment made during the normal breakdown of red blood cells" (*Bilirubin Test: Normal & High Levels in Adults & Newborns*, no date). It is typically removed as waste in bile, and jaundice can result when the body is not breaking down these red blood cells effectively.

However, this form of therapy can also lead to complications. Some neonates have experienced the formation of rashes, nerve damage to the eyes and dehydration (Hansen 2017). It also becomes more difficult to regulate the temperature of a neonate because the baby needs to be exposed to light to break down the bilirubin, but the temperature of the neonate is not easily maintained at a steady level with this constant light exposure. This is a significant consideration for future warming blanket designs, as the incorporation of jaundice treatment in the blanket would produce a more comprehensive design, but perhaps at the cost of reliable thermoregulation.

An alternative solution is the use of a fiber-optic blanket (Hansen 2017). In the neonatal sector, these fiber-optic blankets are commonly referred to in industry as BiliBlankets (*Your Baby, Jaundice and Phototherapy*, no date). An image depicting the characteristic blue light emitted from these blankets is shown in Figure 5-3.

Figure 5-3: Characteristic blue light emitted from a BiliBlanket. (Body Iron | chemistryinmedicine, no date).

These blankets consist of fiber-optic pads which are covered in woven fabric to prevent burns or irritation to the skin. There is no need to move the baby or BiliBlanket, because bilirubin is primarily in the blood stream and so, as the neonate's blood moves around the body, the blanket will be able to break down this bilirubin without the need for rotation (*Your Baby, Jaundice and Phototherapy*, no date).

5.5.6 FURTHER TESTING

In terms of future studies related to this thesis, it would be interesting to compare three sample groups such that one involves neonates using only the designed blanket, one involves neonates using an incubator for temperature regulation, and the third involves neonates using both the blanket and an incubator in conjunction. Each of these study groups would be compared in terms of power required

to maintain the neonate's body temperature, as well as the ability of these systems to regulate temperature and maintain it within a healthy range. Qualitative information would also be determine through questionnaires and interviews with the neonatal staff participating in the study. Additionally, the effect of motion artefact should be analyzed in depth with regards to all included sensors.

5.5.6.1 TESTING PID ALGORITHM CONTROL

There are numerous tests utilized in industry to determine the control of a PID algorithm, and these should be incorporated into future work on this project. The step test is the first of these, and involves a substantial change to the set point[27] (SP) which results in a corresponding process variable[28] (PV) response. The change in set point is depicted in Figure 5-4.

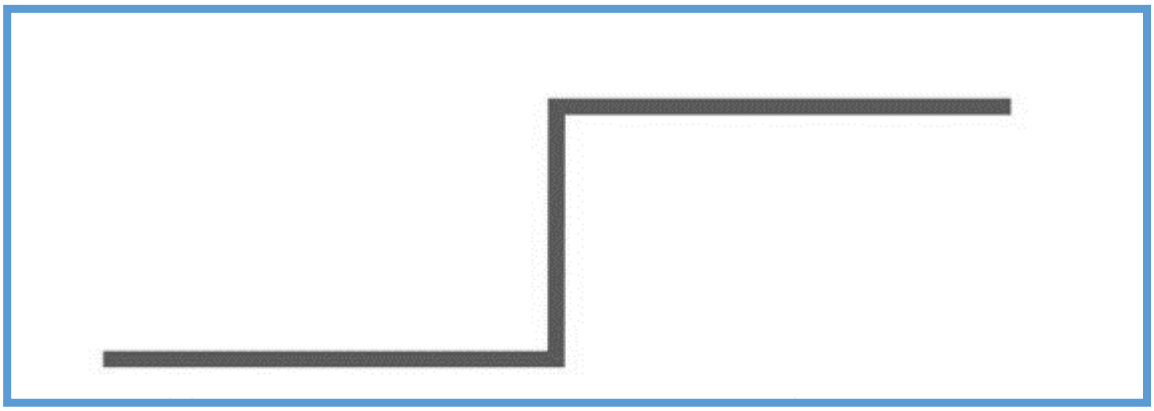


Figure 5-4: Example stimulus applied during a step test (*How to Perform a Step Test - Control Station*, no date).

The second test is the bump test which is essentially two step tests performed in quick succession, as shown in Figure 5-5. The set point in a bump test is first stepped in one direction, before being stepped in the opposite direction.

[27] The set point in this case is the desired temperature of the neonate.

[28] The process variable is "the actual value in the control loop" (*Process Control and Common Terms*, no date) which is, in this case, the actual temperature of the neonate as determined by the temperature sensors.

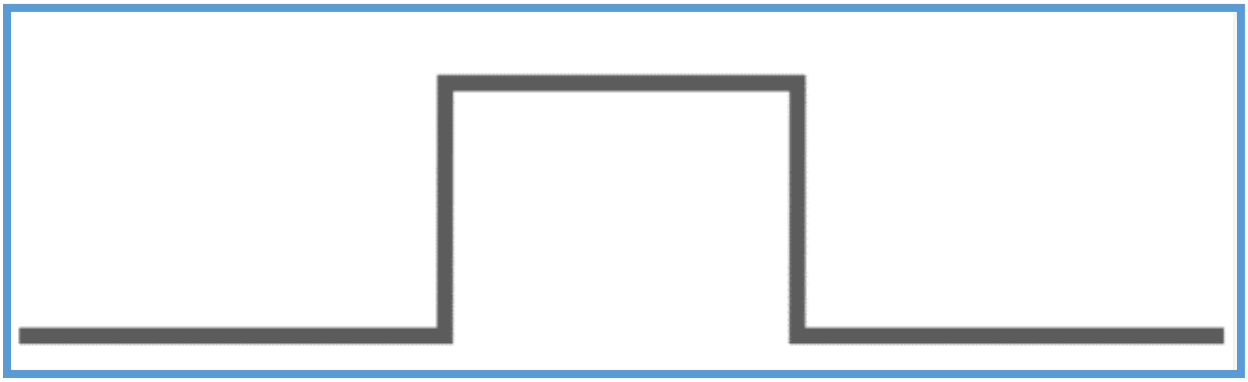


Figure 5-5: Example stimulus applied during a bump test (*How to Perform a Step Test - Control Station*, no date).

The third test to be performed is the pseudo-random binary sequence (PRBS) is essentially based on a series of bump tests performed in quick succession. These bump tests are "uniform in amplitude, alternating in direction, and of random duration" (*How to Perform a Step Test - Control Station*, no date), as represented in Figure 5-6.

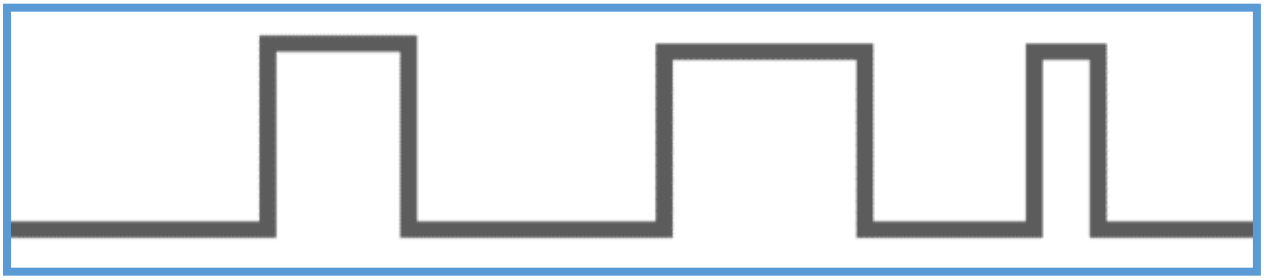


Figure 5-6: Example stimulus applied during a pseudo-random binary sequence test (*How to Perform a Step Test - Control Station*, no date).

Lastly, the doublet test is the equivalent of two bump tests performed in quick succession and the second bump test begins immediately after the system responds to the first bump test, as shown in Figure 5-7. The second bump returns the set point to its initial value.

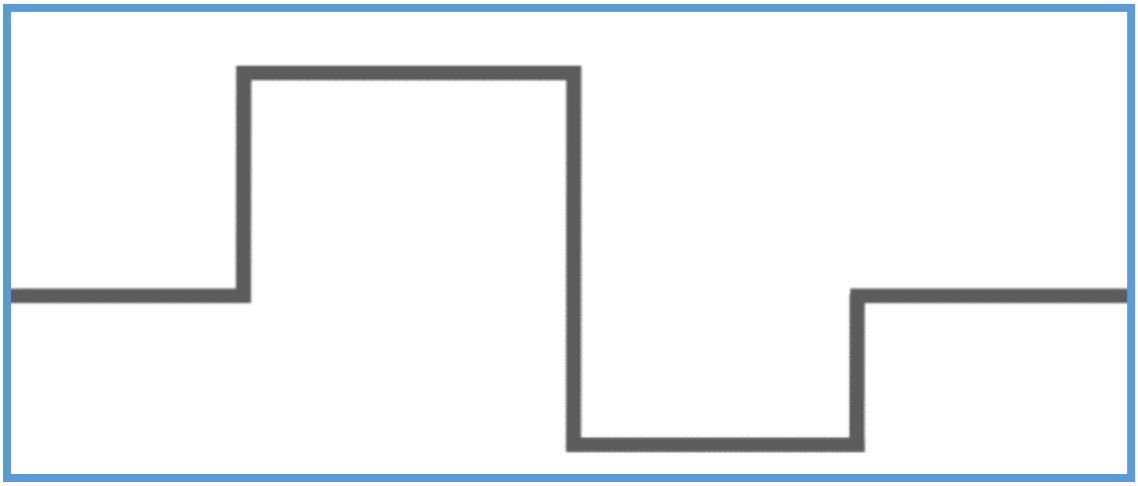


Figure 5-7: Example stimulus applied during a doublet test (*How to Perform a Step Test - Control Station*, no date).

Each of these PID testing methodologies is utilised in industry with varying advantages and disadvantages, as listed in Table 5-2.

Test	Advantages	Disadvantages
Step Test	Can be used if there is no modelling or tuning software available (parameter values can be manually calculated).	Time intensive. Results must be disregarded if disturbances take place during testing.
Bump Test	More thorough than step test.	Requires tuning software.
PRBS Test	Suited to systems sensitive to large changes in PV.	Complex. Time intensive.
Doublet Test	Most comprehensive. Least disruptive.	

Table 5-2: Comparison of various testing methods used in industry for algorithm tuning (Rice, 2017). The ideal test for future work on this project has been highlighted in green.

Experimentation in the engineering field has shown that the most thorough method of testing a PID algorithm is through the implementation of multiple doublet tests (*How to Perform a Step Test - Control Station*, no date). A set of three doublet tests has been shown to frequently provide the most accurate PID system, and this would be an important consideration for future work on this project.

5.5.7 TESTING GUI USABILITY

The usability of a user interface can be used to describe the ease of access that the interface provides. ISO 9241-11 defines product usability as "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use" (*What is Usability?*, no date). This involves the ability of the user to achieve their objective, as well as ease of use when first using the interface, and during subsequent experiences.

The usability of an application has been investigated in numerous studies, and a summary of common usability metrics is shown in Table 5-3.

Constantine and Lockwood (1999)	ISO 9241-11 (1998)	Schneiderman (1992)	Nielsen (1993)	Preece et al. (1994)	Shackel (1991)
Efficiency in use	Efficiency	Speed of performance	Efficiency of use	Throughput	Effectiveness (speed)
Learnability		Time to learn	Learnability (ease of learning)	Learnability (ease of learning)	Learnability (time to learn)
Rememberability		Retention over time	Memorability		Learnability (retention)
Reliability in use		Rate of errors by users	Errors/safety	Throughput	Effectiveness (errors)
User satisfaction	Satisfaction (comfort and acceptability of use)	Subjective satisfaction	Satisfaction	Attitude	Attitude

Table 5-3: Usability metrics mentioned in various publications and models (Seffah *et al.*, 2006).

In future, when assessing the usability of the user interface shown in Table 3-41, the metrics from Table 5-3 could be used. In terms of the user interface designed in this thesis, the most important criteria would be 'reliability in use', followed closely by 'efficiency in use' and 'learnability'. These three factors are assessed in five of the six models highlighted in Table 5-3. These usability metrics would be incorporated into a survey or questionnaire such as that shown in Appendix G (Survey: Usability of GUI).

5.5.7.1 NEONATAL STAFF INPUT

Another useful feature of the user interface would be the incorporation of neonatal staff input. It was mentioned earlier than the desired temperature for a neonate is dependent on the weight and age of the individual neonate using the warming blanket. Therefore, this information could be entered directly into the system by neonatal staff using the GUI, and the PID algorithm variables would be calculated based on this information. This would provide a more customisable system, which will ensure more responsive care for each individual neonate.

6 CONCLUSION

Temperature management is a significant requirement in the neonatal sector, and commercially available warming blankets tend to be overly basic in their approach. Commercial warming blankets do not respond to the real-time temperature of the neonate, nor do they monitor temperature for dangerous fluctuations.

The blanket designed in this thesis has the potential to contribute to improved temperature management for a particular group of neonates. This is due to the incorporation of a PID algorithm, which compares the neonatal temperature to the desired temperature, and changes the heat output of the heating elements accordingly.

In addition to this, the design incorporates the sensing capabilities of neonatal incubators, with regards to temperature, pulse and humidity sensing. Therefore, this project can 'bridge the gap' between neonatal incubators and warming blankets, serving as a more comprehensive and safe system overall when compared to existing warming blankets.

7 REFERENCES

5x Heartbeat Detection Sensor Module By Finger 5V For Arduino Phototransistor | eBay (no date). Available at: https://www.ebay.com.au/i/282830733831?chn=ps (Accessed: 21 April 2018).

AD8232 ECG Heart Monitor Measurement Sensor Module Pulse Signal Board +Cable | eBay (no date). Available at: https://www.ebay.com.au/i/392008473641?chn=ps (Accessed: 21 April 2018).

Advanced Temperature Control Strategies for Thermal Testing (no date). Available at: http://info.totaltemptech.com/advanced-temperature-control-strategies-for-thermal-testing (Accessed: 21 April 2018).

AM2320 Digital Temperature and Humidity Sensor Australia (no date). Available at: https://core-electronics.com.au/am2320-digital-temperature-and-humidity-sensor.html (Accessed: 16 April 2018).

Analogue Temperature Sensor For Arduino ARD2-2213 | eBay (no date). Available at: https://www.ebay.com.au/itm/Analogue-Temperature-Sensor-For-Arduino-ARD2-2213-/331837679615?_trksid=p2385738.m4383.l4275.c10 (Accessed: 16 April 2018).

Arduino Due - DEV-11589 - SparkFun Electronics (no date). Available at: https://www.sparkfun.com/products/11589 (Accessed: 21 April 2018).

Arduino K Type Thermocouple Temperature Sensor (no date). Available at: https://www.auselectronicsdirect.com.au/arduino-k-type-thermocouple-temperature-sensor?gclid=CjwKCAiA4vbSBRBNEiwAMorERyyek5y9zFtXe2uKfaVpqZl21LcF6MmGbCFugiPIzfin__iMBrrloxoCVloQAvD_BwE (Accessed: 16 April 2018).

Arduino MKR GSM 1400 (no date). Available at: https://store.arduino.cc/usa/mkr-gsm-1400 (Accessed: 21 April 2018).

Arduino MKR WAN 1300 (LoRa connectivity) (no date). Available at: https://store.arduino.cc/usa/mkr-wan-1300 (Accessed: 21 April 2018).

Arduino Pulse Heart Rate Sensor (no date). Available at: http://www.dx.com/p/pulsesensor-arduino-pulse-heart-rate-sensor-black-white-233467?tc=AUD&ta=AU&gclid=Cj0KCQjwn-bWBRDGARIsAPS1svvJQm3ygRJ0f7LPChNpbXJgPgtVNVuIZNL1AQkE0AW_7V5ld2U_mGwaAr1BEALw_wcB#.WtqjbZe-k1l (Accessed: 21 April 2018).

Arduino Temperature Humidity and Pressure Sensor Module BME280 | Catch.com.au (no

date). Available at: https://www.catch.com.au/product/arduino-temperature-humidity-and-pressure-sensor-module-bme280-437494/?offer_id=529727&&iv_=__iv_p_1_a_934060956_g_49294210511_c_222157459584_k__m__w_pla-386781407710_n_g_d_c_v__l__t__r_1o2_x_pla_y_11248162_f_online_o_4374 (Accessed: 16 April 2018).

Arduino Temperature Sensor Module (no date). Available at: https://www.auselectronicsdirect.com.au/arduino-temperature-sensor-module~316?gclid=CjwKCAiA4vbSBRBNEiwAMorER1zp0K6KA1fLOOYw5m42GzE6T-QHNx2QCQ8ZVkeFL3cL4jmP9V8ysRoCXdIQAvD_BwE (Accessed: 16 April 2018).

Arduino Temperature Sensor Module | Catch.com.au (no date). Available at: https://www.catch.com.au/product/arduino-temperature-sensor-module-425421/?offer_id=500579&&iv_=__iv_p_1_a_934060956_g_49294210511_c_222157459584_k__m__w_aud-295505314062:pla-386281237187_n_g_d_c_v__l__t__r_1o3_x_pla_y_11248162_f_online_o_425421_z_AU_i_en (Accessed: 16 April 2018).

Battle Of The IDEs: Xcode Vs. Eclipse (2013). Available at: https://blog.rackspace.com/battle-of-the-ides-xcode-vs-eclipse (Accessed: 22 April 2018).

BeagleBone Black - Rev C - DEV-12857 - SparkFun Electronics (no date). Available at: https://www.sparkfun.com/products/12857 (Accessed: 21 April 2018).

Biomedical Sensor Pad (10 pack) - SEN-12969 - SparkFun Electronics (no date). Available at: https://www.sparkfun.com/products/12969 (Accessed: 21 April 2018).

Block, F. E., Nuutinen, L. and Ballast, B. (1999) 'Optimization of alarms: a study on alarm limits, alarm sounds, and false alarms, intended to reduce annoyance.', *Journal of clinical monitoring and computing*, 15(2), pp. 75–83. Available at: http://www.ncbi.nlm.nih.gov/pubmed/12578080 (Accessed: 16 April 2018).

Bluno Mega 2560 - A Bluetooth 4.0 Micro-controller Compatible with Arduino Mega Australia (no date). Available at: https://core-electronics.com.au/bluno-mega-2560-a-bluetooth-4-0-micro-controller-compatible-with-arduino-mega.html (Accessed: 21 April 2018).

Bridge Circuits (no date). Available at: https://www.slideshare.net/akshata_thorat/bridge-ckt-eim (Accessed: 21 April 2018).

Bridge Configurations for Pressure Sensors (2018). Available at: https://www.azosensors.com/article.aspx?ArticleID=1044 (Accessed: 21 April 2018).

Coppetti, T. *et al.* (2017) 'Accuracy of smartphone apps for heart rate measurement', *European*

Journal of Preventive Cardiology, 24(12), pp. 1287–1293. doi: 10.1177/2047487317702044.

Cvach, M. (2012) 'Monitor alarm fatigue: An integrative review', *Biomedical Instrumentation and Technology*, pp. 268–277. doi: 10.2345/0899-8205-46.4.268.

DFRobot Gravity: Analog Heart Rate Monitor Sensor ECG for Arduino | eBay (no date). Available at: https://www.ebay.com.au/i/282806223896?chn=ps (Accessed: 21 April 2018).

DS18B20 DC 5V Digital Temperature Sensor Module For Arduino (no date). Available at: https://www.banggood.com/DS18B20-DC-5V-Digital-Temperature-Sensor-Module-For-Arduino-p-930691.html?gmcCountry=AU&currency=AUD&createTmp=1&utm_source=googleshopping&utm_medium=cpc_elc&utm_content=zouzou&utm_campaign=pla-au-elc2-pc&gclid=CjwKCAiA4vbSBRBNEiw (Accessed: 16 April 2018).

DS18B20 Temperature Sensor + DS18B20 Adapter Module for Arduino (no date). Available at: http://www.dx.com/p/waterproof-ds18b20-temperature-sensor-100cm-ds18b20-adapter-module-for-arduino-414361?tc=AUD&ta=AU&gclid=CjwKCAiA4vbSBRBNEiwAMorERxyd36Q2Apwev3eVZcGdo02VWXNnd1zMgphc5d5xqzEh_yDmMWZfnRoCWDUQAvD_BwE#.WtPtoJe-k1l (Accessed: 16 April 2018).

DS18B20 Temperature Sensor for Arduino Lovers (no date). Available at: https://www.gearbest.com/sensors/pp_423491.html?wid=81&currency=AUD&vip=985807&gclid=CjwKCAiA4vbSBRBNEiwAMorER_X7SPNXp2p5dZDCF4BNfmWuWCDU-YvJASsC8dwzMvFZROIechVtfxoCaM0QAvD_BwE (Accessed: 16 April 2018).

Edworthy, J. and Hellier, E. (2006) 'Alarms and human behaviour: implications for medical alarms', *British Journal of Anaesthesia*, 97(1), pp. 12–17. doi: 10.1093/bja/ael114.

FilteredFlo® | Patient Warming Blanket (no date). Available at: https://www.cszmedical.com/filtered-flo-overview (Accessed: 19 April 2018).

FreeSoC2 Development Board - PSoC5LP - DEV-13714 - SparkFun Electronics (no date). Available at: https://www.sparkfun.com/products/13714 (Accessed: 21 April 2018).

Fujii, K. *et al.* (2010) 'Incidence and risk factors of pressure ulcers in seven neonatal intensive care units in Japan: A multisite prospective cohort study', *International Wound Journal*, 7(5), pp. 323–328. doi: 10.1111/j.1742-481X.2010.00688.x.

Garancsi, T., Robinson, P. and Maldonado, N. (no date) *What are the advantages and disadvantages of NetBeans and Eclipse?* Available at: https://www.quora.com/What-are-the-advantages-and-disadvantages-of-NetBeans-and-Eclipse (Accessed: 22 April 2018).

GY-MAX30100 Low Power Heart Rate Click Sensor Moudle for Arduino | eBay (no date). Available at: https://www.ebay.com.au/i/162944735140?chn=ps (Accessed: 21 April 2018).

Heart Rate Click module Sensor Blood Oxygen Pulse Module for Arduino MAX30100 | Alibaba Group (no date). Available at: https://www.aliexpress.com/item/MAX30100-Heart-Rate-Sensor-Blood-Oxygen-Pulse-Heart-Rate-Module-Blood-Oxygen-Sensor-Module/32783707295.html?src=google&albslr=120472223&isdl=y&aff_short_key=UneMJZVf&source=%7Bifdyn:dyn%7D%7Bifpla:pla%7D%7Bifdbm:DBM&albch=D (Accessed: 21 April 2018).

High Accuracy LM75A I2C Temperature Sensor Module for Arduino (no date). Available at: https://www.geekbuying.com/item/High-Accuracy-LM75A-I2C-Temperature-Sensor-Module-for-Arduino-367998.html?Currency=AUD&gclid=CjwKCAiA4vbSBRBNEiwAMorER2g8ThVBBkbgFzFfi0m9lIpMiaQqEWXhf7F8U-Ta0tBCZxdL5h4uGhoC6KEQAvD_BwE (Accessed: 16 April 2018).

Hughes, M. (2016) *$5 Microcontrollers: Arduino, Raspberry Pi Zero, Or NodeMCU?* Available at: https://www.makeuseof.com/tag/5-microcontrollers-arduino-raspberry-pi-zero-nodemcu/ (Accessed: 21 April 2018).

iCare Health Monitor (BP & HR) (no date). Available at: https://play.google.com/store/apps/details?id=comm.cchong.BloodAssistant&hl=en (Accessed: 22 April 2018).

Keyes Temperature Humidity Sensor DHT11 For Arduino (no date). Available at: https://hobbyking.com/en_us/keyes-temperature-humidity-sensor-dht11-for-arduino.html?countrycode=AU&gclid=CjwKCAiA4vbSBRBNEiwAMorER2erG6vfVZxH4f--pu4knzLY5d2NRZjswclUwzWcLcaeDIg6YWmYqBoCuIkQAvD_BwE&gclsrc=aw.ds&___store=en_us (Accessed: 16 April 2018).

Keyestudio Analog Temperature Sensor Compatible with Arduino (no date). Available at: http://www.dx.com/p/keyestudio-analog-temperature-sensor-compatible-with-arduino-black-429462?tc=AUD&ta=AU&gclid=CjwKCAiA4vbSBRBNEiwAMorER31PzNYvGkbax5dOZFYS1JeZrSmJzsHQghZVp5MdRbx2jofvRsqNCxoCX1EQAvD_BwE#.WtPvLJe-k1l (Accessed: 16 April 2018).

Konkani, A., Oakley, B. and Bauld, T. J. (2012) 'Reducing hospital noise: A review of medical device alarm management', *Biomedical Instrumentation and Technology*, pp. 478–487. doi: 10.2345/0899-8205-46.6.478.

Lecture Notes: Strain Gage Details (no date). Available at: http://classes.mst.edu/civeng120/lessons/flexure/gages/details/index.html (Accessed: 21 April

2018).

MCP9808 High Accuracy I2C Temperature Sensor Breakout Board Australia (no date). Available at: https://core-electronics.com.au/mcp9808-high-accuracy-i2c-temperature-sensor-breakout-board.html?utm_source=google_shopping&gclid=CjwKCAiA4vbSBRBNEiwAMorER7QKWAA0zimN_s4tsygbo197lF7C7QrsWDy_fMPgEMwq4X-DVNXy4RoCvuMQAvD_BwE (Accessed: 16 April 2018).

MicroPython pyboard v1.1 (with Headers) - DEV-14413 - SparkFun Electronics (no date). Available at: https://www.sparkfun.com/products/14413 (Accessed: 21 April 2018).

Mistral-Air Neonatal Plus | The37Company (no date). Available at: https://the37company.com/product_overview/forced_air_warming_1/blankets_plus/mistral-air-neonatal-plus (Accessed: 19 April 2018).

Mylar Bunting - Sterile Mylar Infant Warming Blankets - Maternova Inc. (no date). Available at: https://maternova.net/products/mylar-infant-bunting (Accessed: 19 April 2018).

NetBeans IDE vs Komodo IDE 2018 Comparison (2018). Available at: https://stackshare.io/stackups/komodo-ide-vs-netbeans (Accessed: 22 April 2018).

NetBeans vs Komodo IDE detailed comparison (2018). Available at: https://www.slant.co/versus/810/815/~netbeans_vs_komodo-ide (Accessed: 22 April 2018).

'Perinatal deaths in Australia' (1993) *Australian Institute of Health and Welfare*. Available at: https://www.aihw.gov.au/getmedia/7cd62505-c674-4926-8035-b879c009f5e3/19440.pdf.aspx?inline=true (Accessed: 16 April 2018).

PID Theory Explained - National Instruments (2011). Available at: http://www.ni.com/white-paper/3782/en/ (Accessed: 15 April 2018).

Pinouts | Adafruit AM2320 Sensor | Adafruit Learning System (no date). Available at: https://learn.adafruit.com/adafruit-am2320-temperature-humidity-i2c-sensor/pinouts (Accessed: 16 April 2018).

Poeggel, S. *et al.* (2015) 'Optical Fibre Pressure Sensors in Medical Applications', *Sensors*, 15(7), pp. 17115–17148. doi: 10.3390/s150717115.

'Power MOSFET Basics' (no date). Available at: http://www.aosmd.com/res/application_notes/mosfets/Power_MOSFET_Basics.pdf (Accessed: 16 April 2018).

Pressure-Sensitive Conductive Sheet (Velostat/Linqstat) Australia (no date). Available at: https://core-electronics.com.au/pressure-sensitive-conductive-sheet-velostat-

linqstat.html?utm_source=google_shopping&gclid=CjwKCAjw-6bWBRBiEiwA_K1ZDVCgXimp7hzSWIOdrO0qQZPSLO4SpTRe-zXcbrb-q80iOxr_Lvsr9xoCkRQQAvD_BwE (Accessed: 16 April 2018).

Pressure Measurement Overview - National Instruments (2016). Available at: http://www.ni.com/white-paper/13034/en/ (Accessed: 21 April 2018).

Pujar, R. (2017) *Learn how a heartbeat sensor works*. Available at: http://www.raviyp.com/embedded/140-learn-how-a-heart-beat-sensor-works (Accessed: 19 April 2018).

Pulse Sensor - SEN-11574 - SparkFun Electronics (no date). Available at: https://www.sparkfun.com/products/11574 (Accessed: 21 April 2018).

Pulse Sensor Australia (no date). Available at: https://core-electronics.com.au/pulse-sensor.html (Accessed: 16 April 2018).

Raspberry Pi 3 Model B+ - Raspberry Pi (no date). Available at: https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/ (Accessed: 21 April 2018).

Raspberry Pi Zero W - Raspberry Pi (no date). Available at: https://www.raspberrypi.org/products/raspberry-pi-zero-w/ (Accessed: 21 April 2018).

READY-HEAT (no date). Available at: http://www.jbmedicalsupplies.com.au/our-products/ready-heat (Accessed: 16 April 2018).

Simmons, M. (2017) *Visual Studio 2017: What You Need to Know*. Available at: http://www.tomsitpro.com/articles/visual-studio-2017-faq,1-3551.html (Accessed: 22 April 2018).

Solntsev, A. (2012) *Intellij vs. Eclipse: Why IDEA is Better*. Available at: https://dzone.com/articles/why-idea-better-eclipse (Accessed: 22 April 2018).

Sound and Buzzer Module | Freetronics (no date). Available at: https://www.freetronics.com.au/products/sound-and-buzzer-module#.WtP1LZe-nIV (Accessed: 16 April 2018).

SparkFun Single Lead Heart Rate Monitor - AD8232 - SEN-12650 - SparkFun Electronics (no date). Available at: https://www.sparkfun.com/products/12650 (Accessed: 21 April 2018).

Suriadi *et al.* (2008) 'Development of a new risk assessment scale for predicting pressure ulcers in an intensive care unit.', *Nursing in critical care*, 13(1), pp. 34–43. doi: 10.1111/j.1478-5153.2007.00250.x.

Temperature Controller Basics Handbook | Instrumart (no date). Available at:

https://www.instrumart.com/pages/283/temperature-controller-basics-handbook (Accessed: 21 April 2018).

Temperature Sensor - TMP36 Australia (no date). Available at: https://core-electronics.com.au/temperature-sensor-tmp36.html?utm_source=google_shopping&gclid=Cj0KCQjw8MvWBRC8ARIsAOFSVBUI1EJVTwbgwgbZU4AFOKOecRDXUg9NKAa3YASQPwJwNdWiE8z1j9oaAke9EALw_wcB (Accessed: 16 April 2018).

Top Reasons to Switch to the NetBeans IDE (no date). Available at: https://netbeans.org/switch/why.html (Accessed: 16 April 2018).

Volkov, V., Dovgodko, A. and Super, D. (2017) *What are the pros and cons of using Visual Studio versus MonoDevelop for Unity game development?* Available at: https://www.quora.com/What-are-the-pros-and-cons-of-using-Visual-Studio-versus-MonoDevelop-for-Unity-game-development (Accessed: 22 April 2018).

Vu, B. (2018) *The Best Electric Blanket of 2018*. Available at: https://www.yourbestdigs.com/reviews/the-best-electric-blanket/ (Accessed: 16 April 2018).

Wardlaw, T. *et al.* (2014) 'UNICEF Report: enormous progress in child survival but greater focus on newborns urgently needed', *Reproductive Health*, 11(82). doi: 10.1186/1742-4755-11-82.

Warmth and Temperature Regulation (no date). Available at: http://www.stanfordchildrens.org/en/topic/default?id=warmth-and-temperature-regulation-90-P02425 (Accessed: 16 April 2018).

Wezorek, J. and Connors, M. (2013) *What are the Pros and Cons of using QT framework for cross platform programming (Win/Mac)?* Available at: https://www.quora.com/What-are-the-Pros-and-Cons-of-using-QT-framework-for-cross-platform-programming-Win-Mac (Accessed: 22 April 2018).

What's New in IntelliJ IDEA (2018). Available at: https://www.jetbrains.com/idea/whatsnew/ (Accessed: 22 April 2018).

Why aren't more desktop apps written with Qt? (no date). Available at: https://softwareengineering.stackexchange.com/questions/88685/why-arent-more-desktop-apps-written-with-qt (Accessed: 22 April 2018).

Why Eclipse instead of word processors and command prompt? (2012). Available at: https://altabel.wordpress.com/2012/05/13/why-eclipse-instead-of-word-processors-and-command-prompt/ (Accessed: 22 April 2018).

Woodward, S. (2000) *Take Back Half: A Novel Integrating Temperature-Control Algorithm | Electronic Design*. Available at: http://www.electronicdesign.com/analog/take-back-half-novel-integrating-temperature-control-algorithm (Accessed: 21 April 2018).

Woodward, S. (2005) *Temperature controller has "take-back-half" convergence algorithm | EDN*. Available at: https://www.edn.com/design/analog/4325740/Temperature-controller-has-take-back-half-convergence-algorithm (Accessed: 21 April 2018).

9 APPENDIX A (INDIVIDUAL TEMPERATURE SENSOR CODE)

```
// Arduino Library for Microchip MCP9808 Temperature Sensor.
// Example sketch.
// Prints the initial register values, then changes the limit registers,
// then prints the register values every two seconds.
//
// Jack Christensen Jun 2015
// Tested with Arduino 1.8.5 and an Arduino Uno.

#include <MCP9808.h>      // http://github.com/JChristensen/MCP9808
#include <Streaming.h>    // http://arduiniana.org/libraries/streaming/

MCP9808 mySensor;

void setup()
{
    Serial.begin(115200);
    Serial << F( "\n" __FILE__ " " __DATE__ " " __TIME__ "\n" );

    delay(1000);
    // time for the sensor to perform an initial temperature conversion
    uint8_t status = mySensor.begin();  // initialise the hardware
    if ( status != 0 )
    {
        Serial << "Error reading sensor, status=" << status << endl;
        Serial.flush();
        while (1);                      // loop until reset
    }
    displayData();                      // print the initial sensor data
    mySensor.tUpper = 25 * 4;           // 25C (77F)
    mySensor.tLower = -10 * 4;          // -10C (14F)
    mySensor.tCritical = 30 * 4;        // 30C (86F)
    mySensor.write();                   // write the changed values
}

void loop()
{
    delay(2000);
    displayData();
```

```
}

// print the data from the sensor on the serial monitor
void displayData()
{
    uint8_t status;

    if ( (status = mySensor.read()) == 0 )
    {
        float C = mySensor.tUpper / 16.0;
        float F = C * 9.0 / 5.0 + 32.0;
        Serial << F("\nUpper Limit\t") << C << F("C\t") << F << F("F\tAlert=") << mySensor.alertUpper << endl;
        C = mySensor.tLower / 16.0;
        F = C * 9.0 / 5.0 + 32.0;
        Serial << F("Lower Limit\t") << C << F("C\t") << F << F("F\tAlert=") << mySensor.alertLower << endl;
        C = mySensor.tCritical / 16.0;
        F = C * 9.0 / 5.0 + 32.0;
        Serial << F("Critical Limit\t") << C << F("C\t") << F << F("F\tAlert=") << mySensor.alertCritical << endl;
        C = mySensor.tAmbient / 16.0;
        F = C * 9.0 / 5.0 + 32.0;
        Serial << F("Ambient\t\t") << C << F("C\t") << F << F("F\n");
        Serial << F("Config 0x") << _HEX(mySensor.config) << F(" Resolution 0x") << _HEX(mySensor.resolution) << endl;
        Serial << F("Mfr ID 0x") << _HEX(mySensor.mfrID) << F( " Device ID 0x") << _HEX(mySensor.deviceID) << F( " Device Rev 0x") << _HEX(mySensor.deviceRev) << endl;
    }
    else
    {
        Serial << "Error reading sensor, status=" << status << endl;
    }
}
```

8 APPENDIX B (INDIVIDUAL HUMIDITY SENSOR CODE)

```
// Example testing sketch for various DHT humidity/temperature sensors
// Written by ladyada, public domain

#include "DHT.h"

#define DHTPIN 2     // what digital pin we're connected to

// Uncomment whatever type you're using!
//#define DHTTYPE DHT11   // DHT 11
#define DHTTYPE DHT22   // DHT 22  (AM2302), AM2321
//#define DHTTYPE DHT21   // DHT 21 (AM2301)

// Connect pin 1 (on the left) of the sensor to +5V
// NOTE: If using a board with 3.3V logic like an Arduino Due connect pin 1
// to 3.3V instead of 5V!
// Connect pin 2 of the sensor to whatever your DHTPIN is
// Connect pin 4 (on the right) of the sensor to GROUND
// Connect a 10K resistor from pin 2 (data) to pin 1 (power) of the sensor

// Initialize DHT sensor.
// Note that older versions of this library took an optional third parameter to
// tweak the timings for faster processors.  This parameter is no longer needed
// as the current DHT reading algorithm adjusts itself to work on faster procs.
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println("DHTxx test!");

  dht.begin();
}

void loop() {
  // Wait a few seconds between measurements.
  delay(2000);
```

```
  // Reading temperature or humidity takes about 250 milliseconds!
  // Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
  // Read temperature as Fahrenheit (isFahrenheit = true)
  float f = dht.readTemperature(true);

  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t) || isnan(f)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }

  // Compute heat index in Fahrenheit (the default)
  float hif = dht.computeHeatIndex(f, h);
  // Compute heat index in Celsius (isFahreheit = false)
  float hic = dht.computeHeatIndex(t, h, false);

  Serial.print("Humidity: ");
  Serial.print(h);
  Serial.print(" %\t");
  Serial.print("Temperature: ");
  Serial.print(t);
  Serial.print(" *C ");
  Serial.print(f);
  Serial.print(" *F\t");
  Serial.print("Heat index: ");
  Serial.print(hic);
  Serial.print(" *C ");
  Serial.print(hif);
  Serial.println(" *F");
}
```

9 APPENDIX C (INDIVIDUAL PULSE SENSOR CODE)

```
/*
THIS PROGRAM WORKS WITH PulseSensorAmped_Arduino ARDUINO CODE
THE PULSE DATA WINDOW IS SCALEABLE WITH SCROLLBAR AT BOTTOM OF SCREEN
PRESS 'S' OR 's' KEY TO SAVE A PICTURE OF THE SCREEN IN SKETCH FOLDER (.jpg)
PRESS 'R' OR 'r' KEY TO RESET THE DATA TRACES
MADE BY JOEL MURPHY AUGUST, 2012
UPDATED BY JOEL MURPHY SUMMER 2016 WITH SERIAL PORT LOCATOR TOOL
UPDATED BY JOEL MURPHY WINTER 2017 WITH IMPROVED SERIAL PORT SELECTOR TOOL

THIS CODE PROVIDED AS IS, WITH NO CLAIMS OF FUNCTIONALITY OR EVEN IF IT WILL WORK
      WYSIWYG
*/

import processing.serial.*;  // serial library lets us talk to Arduino
PFont font;
PFont portsFont;
Scrollbar scaleBar;

Serial port;

int Sensor;      // HOLDS PULSE SENSOR DATA FROM ARDUINO
int IBI;         // HOLDS TIME BETWEN HEARTBEATS FROM ARDUINO
int BPM;         // HOLDS HEART RATE VALUE FROM ARDUINO
int[] RawY;      // HOLDS HEARTBEAT WAVEFORM DATA BEFORE SCALING
int[] ScaledY;   // USED TO POSITION SCALED HEARTBEAT WAVEFORM
int[] rate;      // USED TO POSITION BPM DATA WAVEFORM
float zoom;      // USED WHEN SCALING PULSE WAVEFORM TO PULSE WINDOW
float offset;    // USED WHEN SCALING PULSE WAVEFORM TO PULSE WINDOW
color eggshell = color(255, 253, 248);
int heart = 0;   // This variable times the heart image 'pulse' on screen
//  THESE VARIABLES DETERMINE THE SIZE OF THE DATA WINDOWS
int PulseWindowWidth = 490;
int PulseWindowHeight = 512;
int BPMWindowWidth = 180;
int BPMWindowHeight = 340;
boolean beat = false;    // set when a heart beat is detected, then cleared when the BPM graph is advanced
```

```
// SERIAL PORT STUFF TO HELP YOU FIND THE CORRECT SERIAL PORT
String serialPort;
String[] serialPorts = new String[Serial.list().length];
boolean serialPortFound = false;
Radio[] button = new Radio[Serial.list().length*2];
int numPorts = serialPorts.length;
boolean refreshPorts = false;

void setup() {
  size(700, 600);  // Stage size
  frameRate(100);
  font = loadFont("Arial-BoldMT-24.vlw");
  textFont(font);
  textAlign(CENTER);
  rectMode(CENTER);
  ellipseMode(CENTER);
// Scrollbar constructor inputs: x,y,width,height,minVal,maxVal
  scaleBar = new Scrollbar (400, 575, 180, 12, 0.5, 1.0);  // set parameters for the scale bar
  RawY = new int[PulseWindowWidth];          // initialize raw pulse waveform array
  ScaledY = new int[PulseWindowWidth];       // initialize scaled pulse waveform array
  rate = new int [BPMWindowWidth];           // initialize BPM waveform array
  zoom = 0.75;                               // initialize scale of heartbeat window

 // set the visualizer lines to 0
 resetDataTraces();

 background(0);
 // DRAW OUT THE PULSE WINDOW AND BPM WINDOW RECTANGLES
 drawDataWindows();
 drawHeart();

// GO FIND THE ARDUINO
  fill(eggshell);
  text("Select Your Serial Port",245,30);
  listAvailablePorts();

}
```

```
void draw() {
if(serialPortFound){
  // ONLY RUN THE VISUALIZER AFTER THE PORT IS CONNECTED
  background(0);
  noStroke();
  drawDataWindows();
  drawPulseWaveform();
  drawBPMwaveform();
  drawHeart();
// PRINT THE DATA AND VARIABLE VALUES
  fill(eggshell);                                  // get ready to print text
  text("Pulse Sensor Amped Visualizer v1.5",245,30);    // tell them what you are
  text("IBI " + IBI + "mS",600,585);                   // print the time between heartbeats in mS
  text(BPM + " BPM",600,200);                          // print the Beats Per Minute
  text("Pulse Window Scale " + nf(zoom,1,2), 150, 585); // show the current scale of Pulse Window

//  DO THE SCROLLBAR THINGS
  scaleBar.update (mouseX, mouseY);
  scaleBar.display();

} else { // SCAN BUTTONS TO FIND THE SERIAL PORT

  autoScanPorts();

  if(refreshPorts){
    refreshPorts = false;
    drawDataWindows();
    drawHeart();
    listAvailablePorts();
  }

  for(int i=0; i<numPorts+1; i++){
    button[i].overRadio(mouseX,mouseY);
    button[i].displayRadio();
  }

}

}  //end of draw loop
```

```
void drawDataWindows(){
    // DRAW OUT THE PULSE WINDOW AND BPM WINDOW RECTANGLES
    noStroke();
    fill(eggshell);  // color for the window background
    rect(255,height/2,PulseWindowWidth,PulseWindowHeight);
    rect(600,385,BPMWindowWidth,BPMWindowHeight);
}

void drawPulseWaveform(){
  // DRAW THE PULSE WAVEFORM
  // prepare pulse data points
  RawY[RawY.length-1] = (1023 - Sensor) - 212;   // place the new raw datapoint at the end of the array
  zoom = scaleBar.getPos();                      // get current waveform scale value
  offset = map(zoom,0.5,1,150,0);                // calculate the offset needed at this scale
  for (int i = 0; i < RawY.length-1; i++) {      // move the pulse waveform by
    RawY[i] = RawY[i+1];                         // shifting all raw datapoints one pixel left
    float dummy = RawY[i] * zoom + offset;       // adjust the raw data to the selected scale
    ScaledY[i] = constrain(int(dummy),44,556);   // transfer the raw data array to the scaled array
  }
  stroke(250,0,0);                               // red is a good color for the pulse waveform
  noFill();
  beginShape();                                  // using beginShape() renders fast
  for (int x = 1; x < ScaledY.length-1; x++) {
    vertex(x+10, ScaledY[x]);                    //draw a line connecting the data points
  }
  endShape();
}

void drawBPMwaveform(){
// DRAW THE BPM WAVE FORM
// first, shift the BPM waveform over to fit then next data point only when a beat is found
 if (beat == true){   // move the heart rate line over one pixel every time the heart beats
   beat = false;      // clear beat flag (beat flag waset in serialEvent tab)
   for (int i=0; i<rate.length-1; i++){
     rate[i] = rate[i+1];                  // shift the bpm Y coordinates over one pixel to the left
   }
// then limit and scale the BPM value
```

```
  BPM = min(BPM,200);                    // limit the highest BPM value to 200
  float dummy = map(BPM,0,200,555,215);  // map it to the heart rate window Y
  rate[rate.length-1] = int(dummy);      // set the rightmost pixel to the new data point value
 }
 // GRAPH THE HEART RATE WAVEFORM
 stroke(250,0,0);                        // color of heart rate graph
 strokeWeight(2);                        // thicker line is easier to read
 noFill();
 beginShape();
 for (int i=0; i < rate.length-1; i++){  // variable 'i' will take the place of pixel x position
   vertex(i+510, rate[i]);               // display history of heart rate datapoints
 }
 endShape();
}

void drawHeart(){
  // DRAW THE HEART AND MAYBE MAKE IT BEAT
    fill(250,0,0);
    stroke(250,0,0);
    // the 'heart' variable is set in serialEvent when arduino sees a beat happen
    heart--;                    // heart is used to time how long the heart graphic swells when your heart beats
    heart = max(heart,0);       // don't let the heart variable go into negative numbers
    if (heart > 0){             // if a beat happened recently,
      strokeWeight(8);          // make the heart big
    }
    smooth();   // draw the heart with two bezier curves
    bezier(width-100,50, width-20,-20, width,140, width-100,150);
    bezier(width-100,50, width-190,-20, width-200,140, width-100,150);
    strokeWeight(1);          // reset the strokeWeight for next time
}

void listAvailablePorts(){
  println(Serial.list());    // print a list of available serial ports to the console
  serialPorts = Serial.list();
  fill(0);
  textFont(font,16);
  textAlign(LEFT);
  // set a counter to list the ports backwards
```

```
  int yPos = 0;
  int xPos = 35;
  for(int i=serialPorts.length-1; i>=0; i--){
    button[i] = new Radio(xPos, 95+(yPos*20),12,color(180),color(80),color(255),i,button);
    text(serialPorts[i],xPos+15, 100+(yPos*20));

    yPos++;
    if(yPos > height-30){
      yPos = 0; xPos+=200;
    }
  }
  int p = numPorts;
   fill(233,0,0);
  button[p] = new Radio(35, 95+(yPos*20),12,color(180),color(80),color(255),p,button);
    text("Refresh Serial Ports List",50, 100+(yPos*20));

  textFont(font);
  textAlign(CENTER);
}

void autoScanPorts(){
  if(Serial.list().length != numPorts){
    if(Serial.list().length > numPorts){
      println("New Ports Opened!");
      int diff = Serial.list().length - numPorts;      // was serialPorts.length
      serialPorts = expand(serialPorts,diff);
      numPorts = Serial.list().length;
    }else if(Serial.list().length < numPorts){
      println("Some Ports Closed!");
      numPorts = Serial.list().length;
    }
    refreshPorts = true;
    return;
  }
}

void resetDataTraces(){
  for (int i=0; i<rate.length; i++){
     rate[i] = 555;      // Place BPM graph line at bottom of BPM Window
    }


        for (int i=0; i<RawY.length; i++){
           RawY[i] = height/2; // initialize the pulse window data line to V/2
         }
       }
```

10 APPENDIX D (FINAL ARDUINO ASSEMBLY CODE)

```
/*************************************************************************************************
  MCP9808 Temperature Sensor Declarations and Initialisations
*************************************************************************************************/
#include <Wire.h>
#include "Adafruit_MCP9808.h"

// Create the MCP9808 temperature sensor objects
Adafruit_MCP9808 tempsensor0 = Adafruit_MCP9808();
Adafruit_MCP9808 tempsensor1 = Adafruit_MCP9808();
Adafruit_MCP9808 tempsensor2 = Adafruit_MCP9808();
Adafruit_MCP9808 tempsensor3 = Adafruit_MCP9808();

/*************************************************************************************************
  DHT22 Humidity Sensor Declarations and Initialisations
*************************************************************************************************/
#include "DHT.h"
#define DHTPIN0 24     // the digital pin that humidity sensor 0 is connected to
#define DHTPIN1 34     // the digital pin that humidity sensor 1 is connected to
#define DHTTYPE DHT22

// Create the DHT22 humidity sensor objects
DHT dht0(DHTPIN0, DHTTYPE);
DHT dht1(DHTPIN1, DHTTYPE);



/*************************************************************************************************
  Pulse Sensor Declarations and Initialisations
*************************************************************************************************/
#include <PulseSensorPlayground.h>
#define USE_ARDUINO_INTERRUPTS true
const int PulseWire = 0;       // PulseSensor PURPLE WIRE connected to ANALOG PIN 7
const int LED13 = 13;          // The on-board Arduino LED, close to PIN 13.
int Threshold = 50;           // Determine which signal to "count as a beat" and which to ignore.

//Create pulse sensor object
PulseSensorPlayground pulseSensor;

/*************************************************************************************************
  Heat Elements Declarations and Initialisations
*************************************************************************************************/
// PWM is connected to the transistors that turn heating elements on/off
#include <SoftPWM.h>

void setup() {

  Serial.begin(9600);
  // The default address is 0x18 and
  // The addresses can be calculated by 'adding' the A0/A1/A2 to the base of 0x18
  tempsensor0.begin(0x18);
```

```
  tempsensor1.begin(0x19);
  tempsensor2.begin(0x1A);
  tempsensor3.begin(0x1B);

  // Initialise two humidity sensors
  dht0.begin();
  dht1.begin();

  // Initialise pulse sensor
  pulseSensor.analogInput(PulseWire);
  pulseSensor.blinkOnPulse(LED13);
  pulseSensor.setThreshold(Threshold);

  //  Initialise PWM pin 3 for heating elements
  SoftPWMBegin();
  SoftPWMSet(3, 0);
}

void loop() {
  // Store temp sensor data in separate variables
  double c0 = tempsensor0.readTempC();
  double c2 = tempsensor2.readTempC();
```

```
// Calculate average using the temp sensor values
double averageTemp = (c0 + c2) / 2;
Serial.print(averageTemp);
Serial.print(",");

// Store humidity sensor values in two separate variables
double h0 = dht0.readHumidity();
double h1 = dht1.readHumidity();

// Print humidity sensor values in Serial Monitor
Serial.print(h0);
Serial.print(",");
Serial.print(h1);
Serial.print(",");

// Store pulse sensor value in a variable (BPM)
int BPM = analogRead(PulseWire);

// Print pulse (BPM) to Serial Monitor
Serial.print(BPM);
Serial.println(",");

// PID NOTES
// Input to PID = averageTemp from temp sensors
```

```
// Output of PID = analog value (higher value = more time high in PWM signal, lower value = more time low in PWM signal)

// PWM NOTES
// Output of PWM = PWM signal (HIGH or LOW)
// Input of PWM = output of PID algorithm
// Heat -> PID -> PWM -> PWM signal that drives MOSFETs -> heat elements

// If neonate temperature is less than 28 degrees (severe hypothermia)
// Set duty cycle to 100% (i.e. PID output = 1)
if (averageTemp < 28) {
  SoftPWMSet(3, 255);
}

// If neonate temperature is between 28 and 32 degrees (moderate hypothermia)
// Set duty cycle to 85% (i.e. PID output = 0.85)
if (averageTemp >= 28 && averageTemp < 32) {
  SoftPWMSet(3, 217);
}

// If neonate temperature is between 32 and 36.5 degrees (mild hypothermia)
// Set duty cycle to 75% (i.e. PID output = 0.75)
if (averageTemp >= 32 && averageTemp < 36.5) {
  SoftPWMSet(3, 191);
}
```

```
  // If neonate temperature is between 36.5 and 37.5 degrees (normal)
  // Set duty cycle to 50% (i.e. PID output = 0.50)
  if (averageTemp >= 36.5 && averageTemp < 37.5) {
    SoftPWMSet(3, 128);
  }

  // If neonate temperature is between 37.5 and 38.8 degrees (mild hyperthermia)
  // Set duty cycle to 25% (i.e. PID output = 0.25)
  if (averageTemp >= 37.5 && averageTemp < 38.8) {
    SoftPWMSet(3, 64);
  }

  // If neonate temperature is greater than or equal to 38.8 degrees (severe hyperthermia)
  // Set duty cycle to 0% (i.e. PID output = 0)
  if (averageTemp >= 38.8) {
    SoftPWMSet(3, 0);
  }
  delay(1000);
}
```

11 APPENDIX E (FINAL PROCESSING ASSEMBLY CODE)

```
import processing.serial.*;
import java.text.DecimalFormat;

//GUI Variables
//final color PAPER = color(240);
final color GREY = color(25, 25, 25);
final color[] INKS = new color[] {
  color(150, 125, 185), // purple
  color(110, 160, 215), // blue
  color(245, 220, 130), // yellow
  color(255, 150, 190), // pink
  color(150, 195, 82) // green
};

//Pulse Sensor Variables
int pulseSensor1;
int BPM; // holds heart rate value from Arduino

// Pulse Sensor
final int pulseSensor = 1;
int sensorValuePulse = 0; // Default for starting.

// Temperature Sensors
double averageTemp;
int sensorValueTemp = 0; // Default for starting.
```

```
// Humidity Sensors
double humSensor0, humSensor1;
int sensorValueHumidity = 0; // Default for starting.

// Heating Elements
boolean heatElement0, heatElement1, heatElement2, heatElement3, heatElement4;

// Calculation Variables
final int allSensors = 2;
final int noSensors = 0;
final int sensorCount = 5; // Number of sensors being used
String[] sensorList = {"1", "2", "3", "4"};
int i = 0;  // sensor list index
boolean dataReceived = false;
int sensorNumber = 1; // Expected by microprocessor.
String val;

// Serial Communication Variables
Serial port4;
boolean usingMicroprocessor = true;

// All sensor data stored in 'data' array with the following elements
//averageTemp = Double.parseDouble(data[0]);
//humSensor0 = Double.parseDouble(data[1]);
//humSensor1 = Double.parseDouble(data[2]);
//heartRate = Double.parseDouble(data[3]);


void setup() {
  size(1000, 700);
  if (usingMicroprocessor == true) {
    port4 = new Serial(this, "COM4", 9600);
    // Set condition to read bytes into a buffer until a newline is received
    port4.bufferUntil('\n');
  }
}

void draw() {

  // Process current sensor data from serial event
  // read it and store it in val
  val = port4.readStringUntil('\n');
  if (val!=null) {

    String[] data = val.split(",");

    // GUI
    float delta = (float)height/(INKS.length+1);
    float size = 0.9*delta;
    translate(width/2.0, 0);
    background(GREY);

    noStroke();
```

```
 for (int i=0; i<INKS.length; i++) {
   translate(0, delta);
   fill(INKS[i]);
   ellipse(0, 0, size, size);
 }

 // Humidity Outside Blanket (Using Humidity Sensor #0)
 humSensor0 = Double.parseDouble(data[1]);
 String humOutsideDataLabel = String.valueOf(humSensor0);
 String humidityOutTextLabel = "Humidity Outside of Blanket: ";
 print("Humidity Sensor 0: ");
 println(humSensor0);
 textSize(28);
 text(humidityOutTextLabel, -470, 10);
 textSize(28);
 fill(0);
 text(humOutsideDataLabel, -30, 15);

 // Humidity Inside Blanket (Using Humidity Sensor #1)
 humSensor1 = Double.parseDouble(data[2]);
 String humInsideDataLabel = String.valueOf(humSensor1);
 String humidityInTextLabel = "Humidity Inside of Blanket: ";
 print("Humidity Sensor 1: ");
 println(humSensor1);
 textSize(28);
 fill(255, 150, 190);
 text(humidityInTextLabel, -460, -110);
 textSize(28);


fill(0);
text(humInsideDataLabel, -30, -105);

// Core Body Temperature
averageTemp = Double.parseDouble(data[0]);
//To display most accurate core temperature data:
//We would place all four temp sensors in very close proximity
//Find average and display this
// Initialise warning if averageTemp is too high or too low
// Use values determined in thesis
// For now, I am using alarms if <20 and >30 to practice in room temp environment
if(averageTemp <= 36.5 || averageTemp >= 37.5){
  String tempAlertLabel = "ALERT! CHECK TEMPERATURE!";
      textSize(28);
fill(110, 160, 215);
text(tempAlertLabel, 75, -345);
}
String temperatureTextLabel = "Core Body Temperature: ";
DecimalFormat dfTemp = new DecimalFormat("###.##");
String tempDataLabel = String.valueOf(averageTemp);
tempDataLabel=(dfTemp.format(averageTemp));
textSize(28);
fill(110, 160, 215);
text(temperatureTextLabel, -440, -345);
```

```
fill(0);
text(tempDataLabel, -40, -340);
print("Average Temperature: ");
println(averageTemp);

    // Heat Element Operation Label
String heatElementStatusOn = "ON";
String heatElementStatusOff = "OFF";
String heatElementTextLabel = "Heating Elements are: ";
textSize(28);
fill(245, 220, 130);
text(heatElementTextLabel, -430, -225);
if(averageTemp > 38.8)
{
  fill(0);
  text(heatElementStatusOff,-440, -200);
}
else
{
fill(0);
text(heatElementStatusOn, -440, -200);
}

// Pulse Sensor
String pulseTextLabel = "Heart Rate: ";
textSize(28);


    fill(150, 125, 185);
    text(pulseTextLabel, -300, -450);
    pulseSensor1 = Integer.parseInt(data[3]);
    // Subtract 235 to account for noise added to the heart rate signal
    pulseSensor1 = pulseSensor1-235;
        // Initialise warning if pulseSensor1 is too high or too low
    // Use values determined in thesis
    // For now, I am using alarms if <80 and >120 to pracice
    if(pulseSensor1 <= 80 || pulseSensor1 >= 120){
      String tempAlertLabel = "ALERT! CHECK HEART RATE!";
          textSize(28);
    fill(150, 125, 185);
    text(tempAlertLabel, 75, -450);
    }
    print("Pulse Sensor: ");
    println(pulseSensor1);
    String pulseDataLabel = String.valueOf(pulseSensor1);
    textSize(28);
    fill(0);
    text(pulseDataLabel, -30, -455);

    // Request new sensor data from microprocessor if previous data has been processed
    if (usingMicroprocessor == true && dataReceived == true) {
      dataReceived = false;  // Waiting for new data
    }
  }
}
```

12 APPENDIX F (INCORPORATION OF A PRESSURE SENSOR)

Pressure sores[29] are another significant issue in the neonatal sector, since regularly moving the neonate can cause undue distress. pressure sores have been found to occur when the external pressure exceeds 33 mm Hg[30] (Agrawal and Chauhan, 2012). At this pressure, the blood vessel becomes occluded, and so cell death and consequent pressure sores result.

A study by Fujii et al. (2010) has concluded that "a total of 14 pressure ulcers occurred in 13 infants during the 11-month study period" (Fujii *et al.*, 2010), indicating that this is a common occurrence. It was determined during this study that skin texture and "endotracheal intubation usage" (Fujii *et al.*, 2010) were the greatest contributing factors. Additional features found to increase the prevalence of pressure sores in the NICU include pressure at the interface between skin and surface and body temperature (Suriadi *et al.*, 2008).

12.1 IDENTIFYING PRESSURE SORES

Contributing factors such as body temperature and pressure have been incorporated into risk assessment scales commonly adopted in literature as a means of predicting and identifying pressure sore formation. One such scale is the Suriadi and Sanada (S.S.) scale (Suriadi *et al.*, 2008). A form used to implement this assessment scale in a clinical setting is shown in Figure 12-1.

[29] Pressure sores are defined as "an area of localized soft tissue ischemic necrosis caused by prolonged pressure higher than the capillary pressure with or without shear, related to posture which usually occurs over a bony prominence" (Agrawal and Chauhan, 2012)

[30] The unit of measurement mm Hg refers to a " unit of pressure equal to the pressure exerted by a column of mercury 1 millimeter high at 0°C and under the acceleration of gravity and nearly equivalent to 1 torr (about 133.3 pascals)" (*Mm Hg Medical Definition*, no date).

Patient name:……..…… Medical record:……Unit:…….……				Date observation		
Observer:……						
Interface pressure	(3) Risk Interface pressure ≥ 35 mmHg (bony prominence; at sacrum).	(0) No risk Interface pressure < 35 mmHg (bony prominence; at sacrum).				
Body temperature	(4) Risk Body temperature ≥ 37.4 C °.	(0) No risk Body temperature < 37.4 C °.				
Cigarette smoking	(2) Risk Cigarette smoking ≥ 10 cigarettes per day before this admission. Stopped smoking ≥ 10 cigarettes per day between 1 month and 1 year before this admission.	(0) No risk Cigarette smoking < 10 cigarettes per day before this admission. Former; cigarette smoking < 10 cigarettes per day, or ≥ 10 cigarettes per day and stopped smoking > 1 year. And/or never smoking.				
			Total score			

Figure 12-1: Example of a form used to implement the Suriadi and Sanada (S.S.) scale (Suriadi et al., 2008).

As can be seen in Figure 12-1, each risk factor is assigned a numerical value, based on its relevance in pressure sore formation. The sum of these numerical values provides an indication of the likelihood of pressure sore formation, where a higher total score indicates a higher probability that pressure sore formation will occur.

12.2 TREATING PRESSURE SORES

Literature has shown that alternating pressure mattresses[31] can significantly reduce the likelihood and severity of pressure sore formation. Commercially available mattresses designed for this function are summarised in Table 12-1.

[31] An alternating pressure mattress "has air-filled channels that alternately fill and empty to keep bearing weight off bony prominences of immobilized or weak patients who are unable to shift their weight frequently" (*Alternating pressure air mattress*, no date).

Product Name	Features	Cost ($AUD)
Med Aire Variable Pressure Pump and Deluxe Pad System (*Med Aire Variable Pressure Pump and Deluxe Pad | Drive - 14001EFD*, no date)	130 cells with variable pressure Pump produces 4 litres per minute 5-minute cycle time Built-in bracket so it can be mounted to the bed 198 cm x 86 cm	126.97
Med Aire Low Air Loss Alternating Pressure Mattress Overlay with Pump (*Med Aire Alternating Pressure Mattress Overlay | Drive - 14025N*, no date)	8 litres per minute LED indicators for pressure levels 17 air cells Spring lined air tube to reduce likelihood of kinks 20 second deflation time 91 cm x 203 cm x 13 cm	469.83

Product Name	Features	Cost ($AUD)
Medline Aeroflow II Static Air Mattress with Hand Pump (*Medline Aeroflow II Static Air Mattress W/Hand Pump - MSC061000*, no date)	Antimicrobial Instant deflation 86 cm x 193 cm Inflation indicator	57.13
Supra DPS Mattress (*Supra DPS Mattress - MDT24SUPRADPS*, no date)	Automatic pump Auto firm functionality Four different alternating pressure cycle times Ability to remain inflated in power outages Remains inflated during transport	1015.87
Professional Medical Imports Probasics Satin Air APM (*Professional Medical Imports Probasics Satin Air APM - PMISATINAIRAPM*, no date)	20 air cells Fast deflation time	634.91

Product Name	Features	Cost ($AUD)
Advantage Therapeutic Homecare Foam Mattress by Medline (*Advantage Therapeutic Homecare Foam Mattress by Medline - MSCADVHC80*, no date)	High density all foam mattress Nylon cover to reduce shear and friction 91 cm x 203 cm x 15 cm	190.46
Premium Guard Water Mattress by Drive (*Premium Guard Patient Relief Water Mattress By Drive - 14400*, no date)	173 cm x 81 cm x 10 cm Conforms to varying body weight Double sealed seams	54.59
Multi-Ply Elite Pressure Foam Mattress by Mason (*Multi-Ply Elite Pressure Foam Mattress by Mason - 6500-DE-2-FB*, no date)	Gel infused foam to draw away heat/keep skin dry 25-degree heel slope 203 cm x 91 cm x 15 cm	412.69

Table 12-1: Sample of commercially available mattresses designed to minimize likelihood and severity of pressure sores.

12.3 PRESSURE SENSOR SELECTION

Pressure sensing has the potential to reduce the likelihood and severity of pressure sores. This is due to the fact that regions of high pressure can be identified by neonatal staff, and corrective actions can be taken. A range of sensors used to measure the pressure have been identified in Table 12-2. Each of these has their own distinct advantages and disadvantages, and these were considered before deciding on the ideal pressure sensor for the application.

Sensor Type	Description	Advantages	Disadvantages
Bridge-based pressure sensor	Incorporation of Wheatstone bridge[32] Relate physical deformity to change in resistance and therefore to applied pressure (*Pressure Measurement Overview - National Instruments*, 2016)	Can measure absolute[33], gauge[34] and differential[35] pressure Can function under high or low pressure Variable accuracy Small size Durable Affordable	Resistance is affected by temperature (*Bridge Configurations for Pressure Sensors*, 2018) Increased inaccuracy due to "contact resistance" (*Bridge Circuits*, no date) Requires calibration High current required

[32] A Wheatstone bridge is "a simple circuit for measuring an unknown resistance by connecting it so as to form a quadrilateral with three known resistances and applying a voltage between a pair of opposite corners" (*Wheatstone bridge | Definition of Wheatstone bridge in English by Oxford Dictionaries*, no date).

[33] Absolute pressure is the pressure compared to that experienced in a perfect vacuum (*What's the Difference Between Gauge, Absolute, Differential, and Sealed Pressure? | Machine Design*, no date).

[34] Gauge pressure "uses a reference to the atmosphere around the sensor" (*What's the Difference Between Gauge, Absolute, Differential, and Sealed Pressure? | Machine Design*, no date) to ensure that the sensor takes pressure due to altitude into consideration.

[35] Differential pressure relates to the difference in pressure between two different media (*What's the Difference Between Gauge, Absolute, Differential, and Sealed Pressure? | Machine Design*, no date). This might be relevant where there is a pressure rise or drop from one end of a solid to the other.

Sensor Type	Description	Advantages	Disadvantages
Capacitive pressure sensor	Uses variable capacitance pressure transducer Change in capacitance is measured, and relate to applied pressure (*Pressure Measurement Overview - National Instruments*, 2016)	Low cost "Can be operational with small magnitude of force" (*Advantages and Disadvantages of Capacitive sensor*, no date) High resolution Can detect through containers Adjustable	Sensitive to high temperatures Difficult to set up
Piezoelectric pressure sensors	Incorporate quartz crystals which generate a charge when they undergo strain Electrodes transfer this charge to an amplifier (*Pressure Measurement Overview - National Instruments*, 2016)	Do not require external source of energy Durable Fast response to pressure changes	Sensitive to shock and vibration Sensitive to high temperatures Difficult to set up Expensive Require protection of the crystal

Sensor Type	Description	Advantages	Disadvantages
Conditioned pressure sensor	Include integrated circuitry Can be constructed by bridge-based, capacitive or piezoelectric transducers (*Pressure Measurement Overview - National Instruments*, 2016)	Immunity to noise Often incorporate filtration and amplification	Expensive External power required

Sensor Type	Description	Advantages	Disadvantages
Optical pressure sensor	Intensity modulation through attenuation Broadband light source emits continuous spectrum, aimed at sensor (Poeggel *et al.*, 2015)	Small size "Immunity to electromagnetic interferences" (Poeggel *et al.*, 2015) Lightweight Flexible Can be used in vivo[36] Affordable Very sensitive to strain Effect of temperature is linear for ease of compensation	Instable after prolonged use Many causes of error (degradation of components, output power drifts, etc.)

[36] The term 'in vivo' is used to describe measurements that occur within the human body, or within a living organism in general (*Definition of In vivo*, no date).

Table 12-2: Types of available pressure sensors.

Referring to Table 12-3, it was possible to determine whether one of these sensors was more suitable for the given application. Firstly, any sensor which was largely affected by a variation in temperature was excluded as that was the very nature of the neonatal warming blanket design. Therefore, the bridge-based, capacitive and piezoelectric sensors were eliminated.

With optical and conditioned pressure sensors the only remaining sensors, it was important to consider additional factors such as cost and lifetime. Optical sensors have been found to degrade after extended use, finding these sensors unsuitable for the current project. Conditioned pressure sensors, on the other hand, can be expensive due to the incorporation of numerous electronic components. Therefore, after thorough analysis of the independent pressure sensor types, it is clear that no sensor is completely suited to the given application, and some compromises would need to be made.

In order to decide on an ideal pressure sensor, the specifications of those currently in the market were identified. This was a necessary action as the temperature sensitivity and cost could be quantified, and an ideal sensor could be selected based on the specifications. The most suitable sensors have been outlined in Table 12-3.

Pulse Sensor		**Cost ($AUD)**	**Accuracy (mbar)**	**Surface Resistivity (Ohm/sq.cm)**	**Sampling Rate**	**Operating Voltage (V)**	**Average Weighting**
	Weighting	**0.10**	**0.40**	**0.20**	**0.20**	**0.10**	**N/A**
GY-68 BMP180 Digital Pressure Sensor Module (*GY-68 BMP180 Digital Pressure Sensor Module (Replaces BMP085)*, no date)	Value	9.50	N/A*	N/A*	N/A*	N/A*	N/A
	Rating	1.00	N/A*	N/A*	N/A*	N/A*	N/A
	Weighted Rating	0.10	N/A*	N/A*	N/A*	N/A*	0.10
littleBits Pressure Sensor (*littleBits*	Value	39.50	N/A*	N/A*	N/A*	N/A*	N/A

Pulse Sensor		Cost ($AUD)	Accuracy (mbar)	Surface Resistivity (Ohm/sq.cm)	Sampling Rate	Operating Voltage (V)	Average Weighting
	Weighting	**0.10**	**0.40**	**0.20**	**0.20**	**0.10**	**N/A**
Pressure Sensor Australia, no date)	Rating	0.70	N/A*	N/A*	N/A*	N/A*	N/A
	Weighted Rating	0.07	N/A*	N/A*	N/A*	N/A*	0.07
Flexiforce Pressure Sensor - 100lbs (*Flexiforce Pressure Sensor - 100lbs. Australia*, no date)	Value	34.81	N/A*	300000.00	N/A*	N/A*	N/A
	Rating	0.80	N/A*	1.00	N/A*	N/A*	N/A
	Weighted Rating	0.08	N/A*	0.20	N/A*	N/A*	0.14

Pulse Sensor		Cost ($AUD)	Accuracy (mbar)	Surface Resistivity (Ohm/sq.cm)	Sampling Rate	Operating Voltage (V)	Average Weighting
	Weighting	**0.10**	**0.40**	**0.20**	**0.20**	**0.10**	**N/A**
Pressure Sensor - MS5803-14BA Breakout (*Pressure Sensor - MS5803-14BA Breakout Australia*, no date)	Value	91.13	+/- 20.00	N/A*	N/A*	1.80 to 3.60	N/A
	Rating	0.10	0.70	N/A*	N/A*	0.60	N/A
	Weighted Rating	0.01	0.28	N/A*	N/A*	0.06	0.12
Extra-long force-sensitive resistor (FSR) - Interlink 408 (*Extra-long force-sensitive resistor (FSR) - Interlink 408 Australia*, no date)	Value	38.50	N/A*	N/A*	Continuous (Analog)	N/A*	N/A
	Rating	0.70	N/A*	N/A*	1.00	N/A*	N/A

Pulse Sensor		**Cost ($AUD)**	**Accuracy (mbar)**	**Surface Resistivity (Ohm/sq.cm)**	**Sampling Rate**	**Operating Voltage (V)**	**Average Weighting**
	Weighting	**0.10**	**0.40**	**0.20**	**0.20**	**0.10**	**N/A**
	Weighted Rating	0.07	N/A*	N/A*	0.20	N/A*	0.14
Square Force-Sensitive Resistor (FSR) - Interlink 406 (*Square Force-Sensitive Resistor (FSR) - Interlink 406 Australia*, no date)	Value	16.95	N/A*	N/A*	Continuous (Analog)	N/A*	N/A
	Rating	0.90	N/A*	N/A*	1.00	N/A*	N/A
	Weighted Rating	0.09	N/A*	N/A*	0.20	N/A*	0.15
Pressure-Sensitive Conductive	Value	6.34	N/A*	< 31000.00	N/A*	N/A*	N/A

Pulse Sensor		**Cost ($AUD)**	**Accuracy (mbar)**	**Surface Resistivity (Ohm/sq.cm)**	**Sampling Rate**	**Operating Voltage (V)**	**Average Weighting**
	Weighting	**0.10**	**0.40**	**0.20**	**0.20**	**0.10**	**N/A**
Sheet (Velostat/Linqstat) (*Pressure-Sensitive Conductive Sheet (Velostat/Linqstat) Australia*, no date)	Rating	1.00	N/A*	1.00	N/A*	N/A*	N/A
	Weighted Rating	0.10	N/A*	0.20	N/A*	N/A*	0.15
EeonTex Pressure Sensing Fabric (*EeonTex Pressure Sensing Fabric Australia*, no date)	Value	40.95	N/A*	2000.00	500.00 cycles per second	N/A*	N/A
	Rating	0.70	N/A*	1.00	0.80	N/A*	N/A
	Weighted Rating	0.07	N/A*	0.20	0.16	N/A*	0.14

Table 12-3: Decision matrix comparing commercially available pressure sensors. The ideal sensor has been highlighted in green.

As shown in Table 12-3, a Pressure-Sensitive Conductive Sheet was found to be the ideal option. While many of the sensors outlined in Table 12-3 were small in size and measured pressure at only one point on the neonate's body, the pressure-sensitive conductive mat was large enough to measure pressure throughout the entirety of the neonate's body at a given time.

Velostat or Linqstat are names commonly used to describe the conductive material that makes up this pressure sensitive sheet (*Pressure-Sensitive Conductive Sheet (Velostat/Linqstat) Australia*, no date). When the material is compressed, its resistance decreases, so there are no complications associated with flexing this type of sensor. This is an ideal characteristic of any sensor included in the design outlined in this thesis, as the blanket will be folded around the neonate.

One consideration of pressure sensitive conductive materials for this application is that they are only able to operate within a particular temperature range. For a temperature control project such as this, there could be difficulties associated with this. However, this particular sheet was chosen as it can safely operate between the temperatures of -45° and 65°C (*Pressure-Sensitive Conductive Sheet (Velostat/Linqstat) Australia*, no date). The temperature of a neonate will likely not exceed 38 °C, or fall below 36°C. Furthermore, the maximum temperature of the selected heating element was shown to be around 60°C. Therefore, the sheet will not malfunction when exposed to these temperatures.

In addition to this, there were further reasons regarding the choice of this sheet rather than strain gauges or other pressure measuring apparatus. Strain gauges are commonly used in industry, and work in a similar manner to that explained with the example of the pressure-sensitive conductive sheet. With this type of sensor, the strain gauge is flexed and there is a consequent change in resistance which is measured accurately by the strain gauge. For this reason, a strain gauge might seem applicable for this design, and it is in many ways. However, these sensors can be bulky, and require long gauge lengths in order to measure the change in resistance (*Lecture Notes: Strain Gage Details*, no date). Maintaining portability of the neonatal warming blanket was considered one of the main project outcomes, and so the strain gauge option was disregarded in favor of the pressure-sensitive conductive sheet which was far less bulky, and therefore contributed to the overall portability of the design.

12.4 TESTING PRESSURE SENSOR INDEPENDENTLY

In future studies, the pressure measured by the Pressure-Sensitive Conductive Sheet would be confirmed mathematically. "IEC 61298-2 states that accuracy must include Hysteresis, Non-Repeatability and Non-Linearity" (*Understanding Pressure Sensor Accuracy | Building Automation Systems | Industry Articles | Dwyer Instruments*, no date). Hysteresis is defined as the maximum difference in the output of the sensor when the pressure on the sensor is increased then decreased. This is depicted in Figure 12-2.

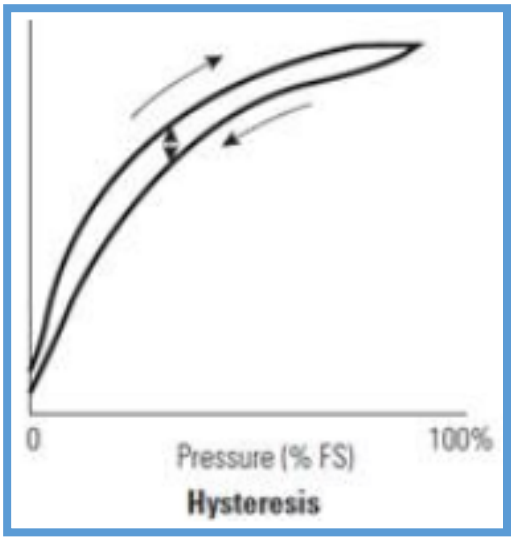


Figure 12-2: Example of hysteresis measurement, used to determine accuracy of a pressure sensing system (*Understanding Pressure Sensor Accuracy | Building Automation Systems | Industry Articles | Dwyer Instruments*, no date).

Non-repeatability can be defined as the maximum difference between the outputs of the sensor when the same pressure is applied, many times over. This property is shown in Figure 12-3.

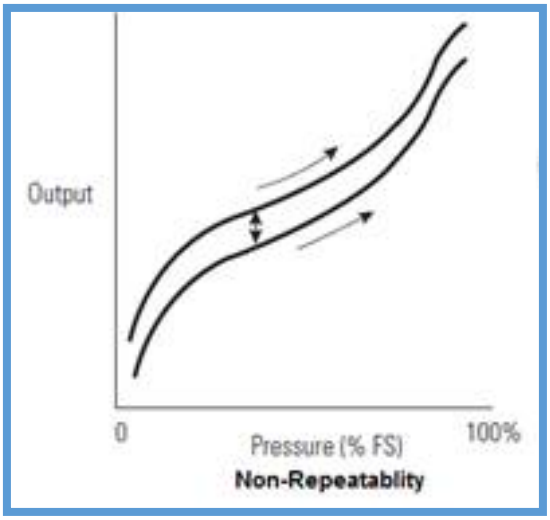


Figure 12-3: Example of non-repeatability measurement (*Understanding Pressure Sensor Accuracy | Building Automation Systems | Industry Articles | Dwyer Instruments*, no date).

Lastly, non-linearity is often defined in a system by one of two things: the Best Fit Straight Line (BFSL) Non-linearity or the Terminal Non-linearity (*Understanding Pressure Sensor Accuracy | Building Automation Systems | Industry Articles | Dwyer Instruments*, no date). These properties are shown in Figure 12-4.

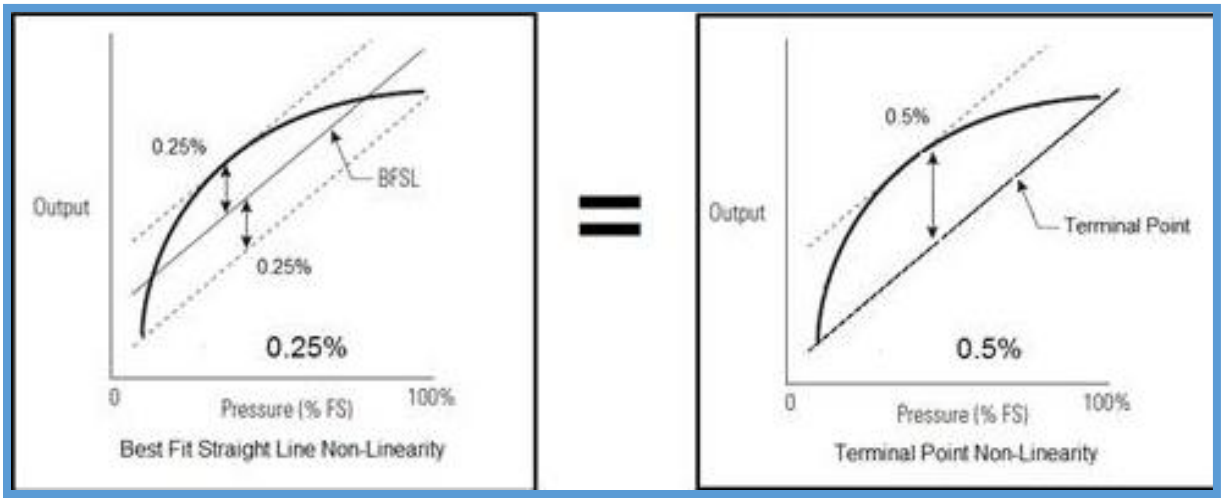


Figure 12-4: Example of non-linearity measurement (*Understanding Pressure Sensor Accuracy | Building Automation Systems | Industry Articles | Dwyer Instruments*, no date).

Figure 12-4 shows that the accuracy can differ based on the method used in the calculation, and this will require further investigation to determine the ideal method for the given application. The final

determination of accuracy can also vary depending on the manufacturer's technique. One of these techniques is the calculation of the Root of the Sum Squared (RSS), as shown below:

$$RSS = \sqrt{(Non - liearity)^2 + (Hysteresis)^2 + (Non - repeatability)^2}$$

The Root of the Mean Squared (RMS) is another technique used to summarise the accuracy of a system, as shown below.

$$RMS = \sqrt{\frac{(Non - linearity)^2 + (Hysteresis)^2 + (Non - repeatability)^2}{3}}$$

Other manufacturers only sum the three variables i.e.

$$Sum = Non - linearity + Hysteresis + Non - repeatability$$

Again, further investigation will ensure that the ideal testing methodology is developed in future studies relating to this neonatal warming blanket design.

13 APPENDIX G (SURVEY: USABILITY OF GUI)

Hi, my name is Morgan Warneford and I recently completed a thesis for my Bachelor of Engineering (Biomedical) (Hons)/Master of Engineering (Biomedical). For this thesis, I designed a neonatal warming blanket with additional sensing capability and temperature compensation functionality. Thank you for taking the time to fill out this survey, which will give me an insight into the usability of the user interface for this device. Please know that you can remain anonymous, and you can withdraw your survey at any time.

1. On a scale of 1 to 10 (where 1 is least comfortable and 10 is most comfortable), how comfortable do you feel accessing the displayed information without assistance?

Least Comfortable									**Most Comfortable**
1	2	3	4	5	6	7	8	9	10

2. On a scale of 1 to 10 (where 1 is least appealing and 10 is most appealing), how appealing do you find the user interface (i.e. your impression based on colours used, font used, graphics, etc.)?

Least Appealing									**Most Appealing**
1	2	3	4	5	6	7	8	9	10

3. On a scale of 1 to 10 (where 1 is very difficult to understand and 10 is very easy to understand), how well are you able to understand the data being shown?

Difficult to Understand									Easy to Understand
1	2	3	4	5	6	7	8	9	10

4. Do you feel as though this interface displays all the information you would like to know? If no, what other information would you like to have included in future versions?

Yes	No

Please comment here.

5. Do you feel as though the information that is shown is mandatory (i.e. do you agree that the current information is necessary to include)?

Yes	No

6. Do you feel as though the displayed data is updated fast enough (i.e. do the values correlate to the real time value, without a delay)?

Yes	**No**

14 INDEX