velostream

March 9, 2011

Dr. Andrew Rawicz
School of Engineering Science
Simon Fraser University
8888 University Drive
Burnaby, British Columbia
V5A 1S6

Re: ENSC 440 Design Specifications for a Blood Flow Speed Measurement Probe

Dear Dr. Rawicz,

Our team at VeloStream Technologies Incorporated is designing a blood flow measurement probe, which can be inserted into the atria of a patient to perform blood flow measurements. The enclosed document, *Design Specifications for the VivaceFlow Blood Flow Speed Measurement Probe,* provides the technical details of the design of our system.

The document first provides a detailed description of the design of each of the major components of the system: the *Piccolo*Probe, the *Genio*Box, and the *Unisca*Suite. We then provide a plan for how the individual components will be integrated, the system calibration protocol, and last, a rigorous system test plan.

Our team is comprised of five undergraduate students from Simon Fraser University: Connie Drewbrook, Wyatt Gosling, Kaveh Naziripour, Jedsada Sahachaiwatana, and Elizabeth Steiner. If you have any questions or concerns regarding our design specifications, please do not hesitate to contact us at velostreamtech@googlegroups.com.

Sincerely,

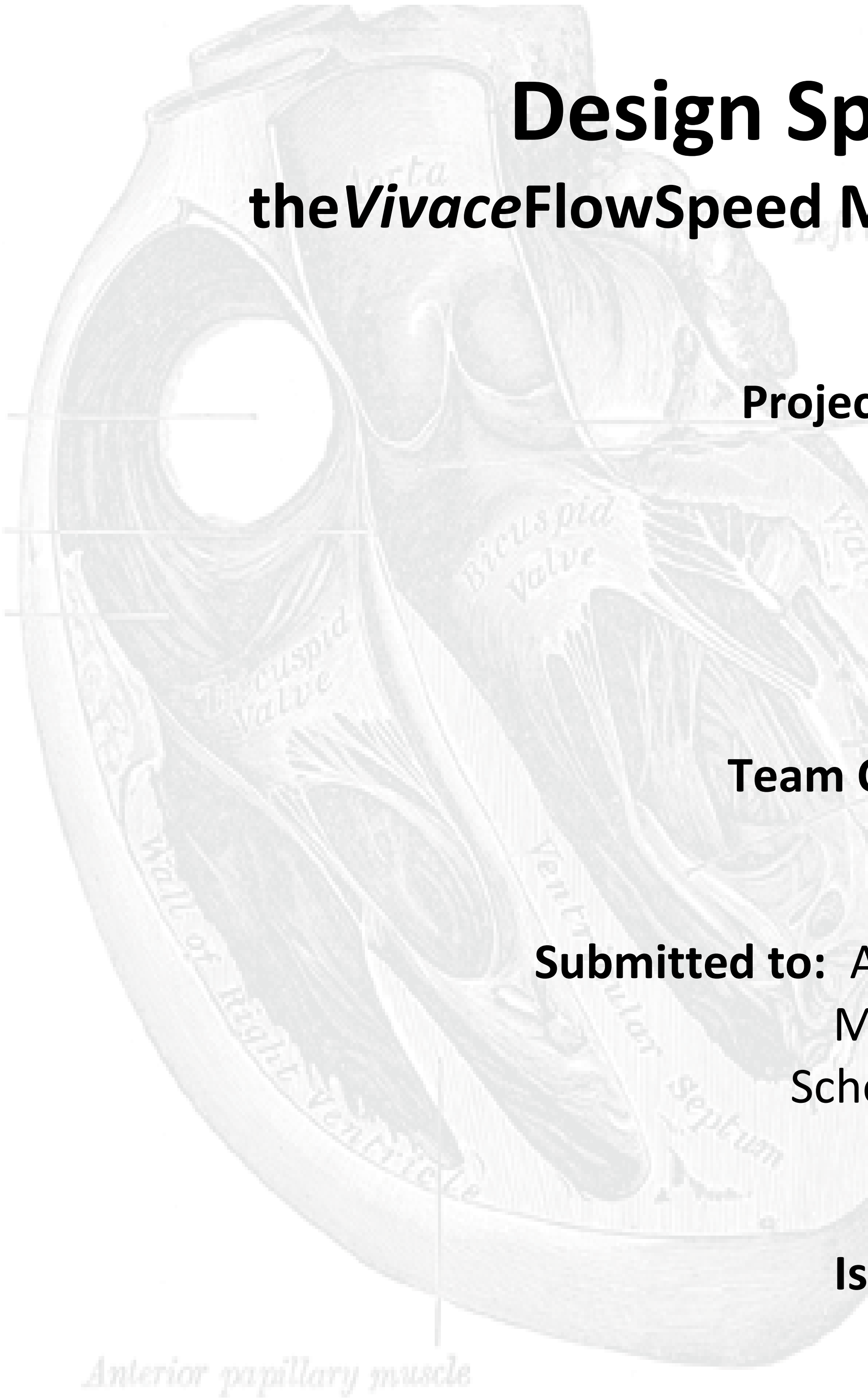
E Steiner

Elizabeth Steiner
Chief Executive Officer
VeloStream Technologies Inc. Suite 2106 - 8888 University Drive
Burnaby, BC V5A 1S6

Enclosure: *Design Specifications for the VivaceFlow Blood Flow Speed Measurement Probe.*

velostream

VeloStream Technologies Incorporated



Design Specification for
the*Vivace*FlowSpeed Measurement Probe

Project Team: Elizabeth Steiner
Wyatt Gosling
KavehNaziripour
Connie Drewbrook
JedsadaSahachaiwatana

Team Contact: Elizabeth Steiner
aes2@sfu.ca

Submitted to: Andrew Rawicz (ENSC 440)
Mike Sjoerdsma (ENSC 305)
School of Engineering Science
Simon Fraser University

Issued Date: March 9, 2011

Executive Summary

The following document is a technical description of the design specifications for the*Vivace*FlowFlow Speed Measurement Probe, and takes into consideration the functionality stated in the functional specifications.

The *Vivace*Flow Flow Speed Measurement Probe is designed to measure blood flow speed within the atria of the heart. The system is divided into three separate components:

- The User Interface: The *Unisca*Suite$^{\text{TM}}$
- The Controller: The *Genio*Box$^{\text{TM}}$
- The Sensor: The *Piccolo*Probe$^{\text{TM}}$

The *Unisca*Suite is a software package programmed in C# which will interface between the user and the GenioBox. Connection of the PC to the *Genio*Boxwill be facilitated by the use of a regular serial port cable provided with the system. The *Genio*Box consists of a microcontroller and printed circuit board (PCB), and will act as the connection between the PC and the *Piccolo*Probe. The *PiccoloProbe* is inserted into the heart, either by direct access or percutaneous navigation by the use of a 17 French guide catheter, for invasive blood flow measurements. This information is then be transmitted back to the *Genio*Boxand then to the software for processing. The results are graphed on the *Unisca*Suite, providing real time flow measurements of the fluid in which the probe has been inserted.

The produced prototype will adhere to the specifications designated with rank ‘A’ in the functional specifications document. Following completion of the prototype, testing will commence to ensure our device meets the stated requirements.

Table of Contents

List of Figures

List of Tables

Glossary

Cardiac catheter	A long catheter designed for passage, usually through a peripheral blood vessel, into the chambers of the heart.
Coagulation	When blood solidifies or clots.
Coronary arteries	Vessels that supply the heart muscle with blood rich in oxygen.
Invasive procedure	A procedure that involves entering the body through the skin or through a cavity or anatomical opening.
Pulmonary venous ostium	The opening to the pulmonary veins which identifies the origin of the vessel
Mutex	An algorithm to prevent concurrent use of a single resource
Percutaneous procedure	A procedure that is done through needle-puncture of the skin, as opposed to an approach using a scalpel.
Pulsatile	Beating, or pertaining to activity that is characterized by rhythmic pulsation.
Sheath	The cover on the opening of the body where the catheter is inserted.

Acronyms

ACK	Acknowledged
ADC	Analog to Digital Converter
CHD	Coronary Heart Disease
EOR	Error Over Head
GUI	Graphical User Interface
NAK	Not Acknowledged
NTC	Negative Temperature Coefficient
PCB	Printed Circuit Board
SOH	Standard Over Head
UART	Universal Asynchronous Receiver/Transmitter

1 Introduction

VeloStream Technologies is developing the *Vivace*Flow Flow Speed Measurement Probe, for biomedical applications. The probe will accurately measure flow rate while being small enough to insert into small enclosures, such as the heart chambers. Flow speed data will be sent back to the *Genio*Box for processing and then to the *Unisca*Suite, where a real-time plot of the data and the flow speed will be displayed.

1.1 Scope

This document describes the design specifications of the *Vivace*Flow Flow Measurement Probe System, and takes into account the requirements described in the functional specifications. The design of each of the components, and the interface between the components is defined. We provide a detailed probe calibration procedure, and test scenarios to ensure that our prototype meets all of the necessary requirements.

1.2 Intended Audience

This document has been prepared for our team at VeloStream Technologies as a guide for the design, integration, calibration, and testing of our system. Each member will be able to refer to the document for design details, to gage their progress to ensure smooth integration, and to carry out test scenarios. Through the planning of these stages, we hope to complete each in a timely fashion.

2 System Requirements

Velostream's *Vicace*Flow Speed Measurement Probe is designed to determine the blood flow speed within the atria of the heart. Typical use involves inserting the device through either a femoral catheter or a sheath to gain access to the chamber of interest. Once inserted, the probe head can be moved throughout the atrium to measure flow speed at various locations.

3 Overall System Design

The *Vivace*Flow system consists of three components: the *Piccolo*Probe, the *Genio*Box, and the *Unisca*Suite. The *Piccolo*Probe will be a small probe that will perform blood flow measurements. The *Piccolo*Probe will contain the minimal amount of electronics necessary to perform blood flow measurements. The *Piccolo*Probe will send flow measurement data to the *Genio*Box for processing. Figure 1 is the expected representation of the *Piccolo*Probe.

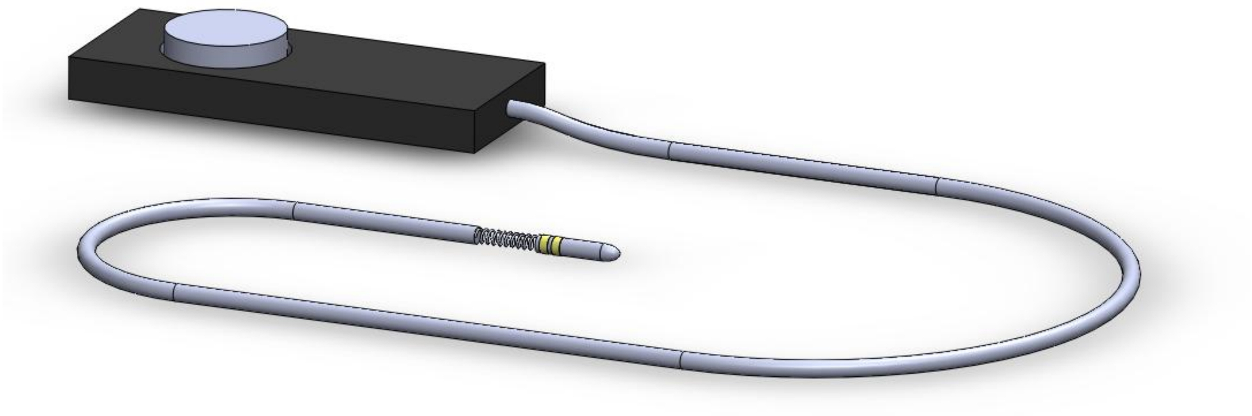


Figure 1 – Image Displaying the Production Model of the *Piccolo*Probe

The *Genio*Box contains all of the electronics needed to process data from the *Piccolo*Probe, as well as regulate the *Piccolo*Probe. It contains a mixture of analog and digital circuits. The *Genio*Box also sends processed probe measurement data to the *Unisca*Suite. Figure 2 shows an early mockup of the *Genio*Box.

Finally, the *Unisca*Suite provides a GUI allowing users to view flow measurements as well as control the operation of the system. The software will display a real time read out of flow measurements, as well as a histogram. Additionally, it will log measurement data to the disk for long term storage, or for use with other programs.

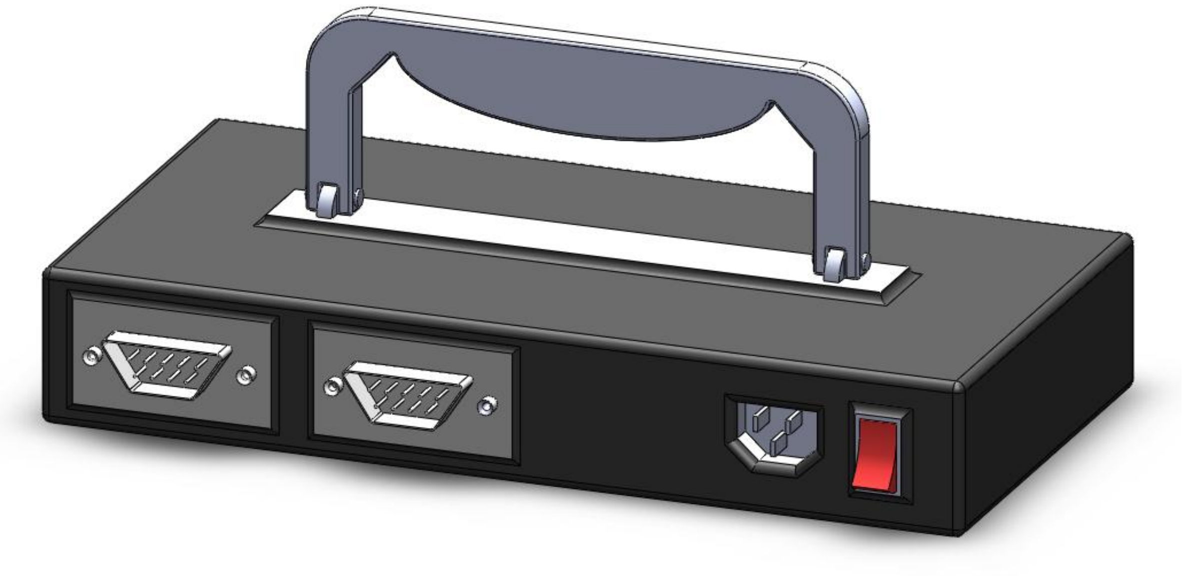


Figure 2 - Image Displaying the Production Model of the *Genio*Box

Figure 3 shows all of the electronics in the system. It contains both analog and digital components. Additionally, it shows how the electronics are distributed between the *Piccolo*Probe, the GenioBox, and the UniscaSuite. The diagram will be explained in detail throughout this document.

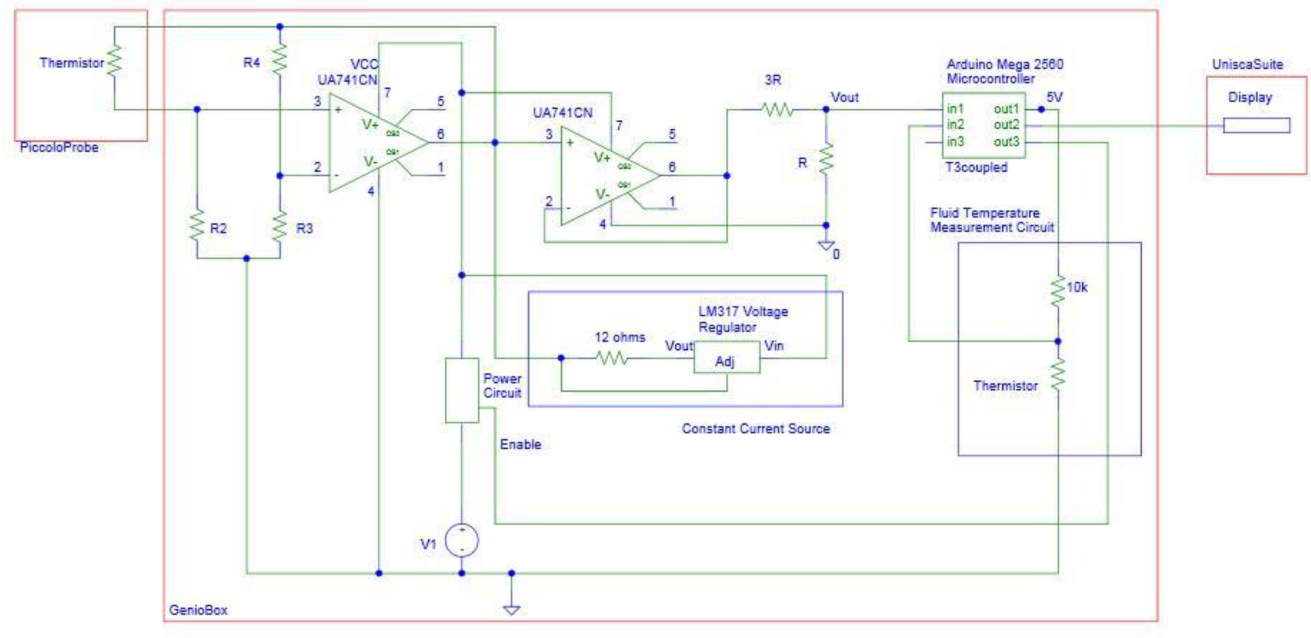


Figure 3 – Complete *Vivace*Flow Electronic Component Schematic

4 *Piccolo*Probe Design

The head of the probe contains a thermistor at its tip which will act as the sensor used to detect flow. The leads of the thermistor will be fed back through the probe connector to the microcontroller, which will contain the circuitry needed to control the thermistor.

4.1 Mechanical Design

The probe consists of a 6 mm diameter plastic tube, 150 mm in length, which at one end contains the sensor, and at the other end contains a nut and a screw to control the angle of the tip of the sensor from the tube. The nut is 7 mm in length and is fixed on the inside of the tube, at the end opposite the tip of the probe. It allows the screw to move laterally with a range of 33 mm along the length of the tube. This is illustrated in Figure 4 below with all the numbers in millimeters scale:

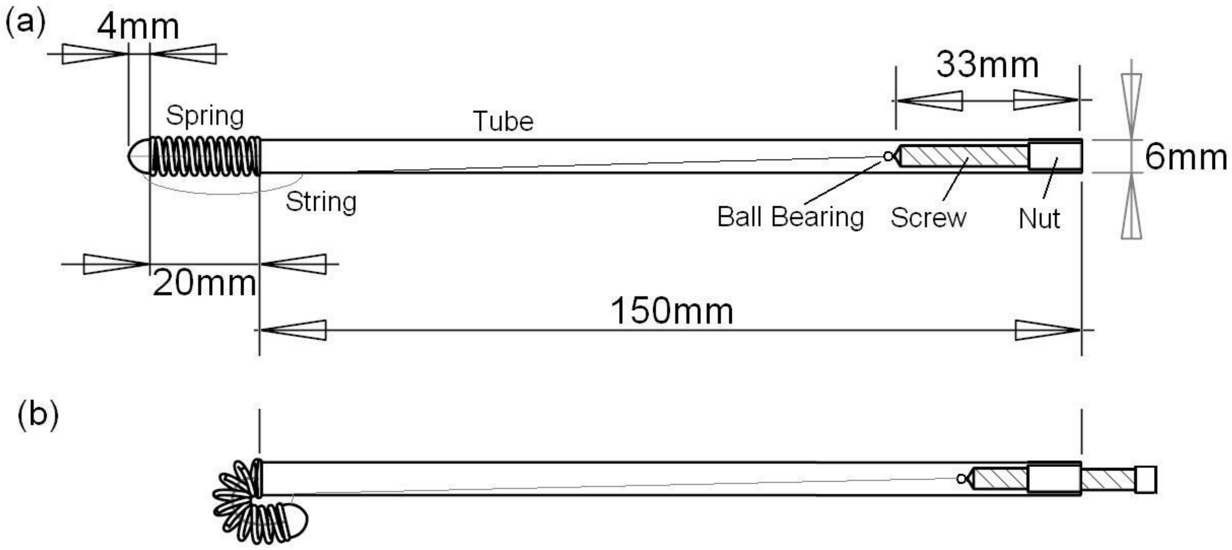


Figure 4 – Schematic of the mechanical design of the *Piccolo*Probe in a) its extended position, and b) when coiled

A string is attached to the screw through the use of a ball bearing, such that when the screw is turned, the string does not twist. The string is fed through the tube and out a small opening to attach to the head of the probe. The screw starts inside the tube as far as it can be inserted. As the screw is twisted out of the tube, it pulls on the string, which pulls on the head of the probe. Between the head of the probe and the far end of the tube is a 20mm long spring. When the string pulls on the probe head, the spring allows the head to bend back toward the tube. This will enable the sensor element to be steered so that it can access a greater range of area within the heart.

The measurement sensor is also attached to the end of the spring opposite the tube. The sensor occupies a 4 mm by 4 mm area of space. The string which is attached to the sensor at the probe's head for steering capability, spring, and the body of the probe are all sealed and will not be in direct contact with the fluid being measured. However, for the prototype model the sensor will be in direct contact with the fluid in order to ensure sufficient accuracy. When the screw is turned, the measurement sensor will be pulled to the side. Via this mechanism, the probe can be steered through nearly 180 degrees. Since the tube can be rotated 360 degrees, and can slide linearly, this means that the sensor can reach everywhere within the atrium except a 1cm radius around the entry orifice.

4.2 Thermistor for Flow Measurement

The probe consists of a negative temperature coefficient (NTC) self-heated thermistor flowmeter. For an NTC thermistor, the thermistor's resistance is proportional to its temperature; as the temperature increases, its resistance decreases. The advantage of using an NTC thermistor for this application compared to other sensors is that it has a reliable performance, high precision, large temperature coefficient of resistance, relatively high dissipation coefficient, and low cost. The thermistor is a B57551 EPCOS glass-encapsulated sensor with insulation. It was chosen since it has a nominal resistance at 25°C of 10kΩ, thermal dissipation constant of 0.8mW/°C, a β value of 3480 K and a heat capacity of 7.2mJ/K. The beta value is the thermal material constant, and a thermistor with a high β value is necessary since this value represents the change in resistance per change in temperature. The resistance of a thermistor at temperature *T* is given by

$$R = R_0 e^{\left(\frac{\beta}{T} - \frac{\beta}{T_0}\right)} \tag{1}$$

where R_0 is the nominal resistance at T_0. We will maintain the thermistor at a temperature of 40.5°C. Using the nominal resistance of the thermistor at 25°C, and the beta value, the resistance of the thermistor at 40.5°C is calculated to be 9.1Ω. When self-heating an NTC thermistor, the applied power,

$$P_{applied} = IV = \delta_{th}(T - T_A) + C_{th}\frac{dT}{dt}$$

where δ_{th} is the thermal dissipation constant, T is the instantaneous temperature of the NTC thermistor, T_A is the ambient temperature, C_{th} is the thermal heat capacity, and $\frac{dT}{dt}$ is the change of temperature with time. Given an ambient temperature of 37^{o}C and a desired operating temperature of 40.5°C, it is evident that 2.4mW needs to be applied in order to start the self-heating process. The regulation of applied power is discussed in section 5.4. The

EPCOS B57551 thermistor was also selected due to its small size, with a width and length of 1.8 mm and a height of 2.3 mm. This will ensure faster response times, and better precision.

4.3 Thermistor for Temperature Measurement

Although human body temperature is usually 37.0°C, it is possible to observe temperatures that deviate slightly from this value. As a result, it is necessary to measure the temperature of the blood before starting blood flow measurements. The circuit illustrated in Figure 5 uses a thermistor to calculate the temperature of the fluid. A potential of 5V is provided to two resistors that act as a voltage divider. The top resistor was chosen to be 10k to limit the current through the thermistor to ensure that it does not self-heat. The resistance of the thermistor will decrease to its resistance at the fluid's temperature. The output voltage will be fed to the microcontroller, which will measure the resistance and calculate the temperature of the fluid. During the calibration of the system, we will determine the coefficients at various fluid temperatures. Based on the fluid's temperature the microcontroller will select the appropriate coefficients for the flow speed calculation.

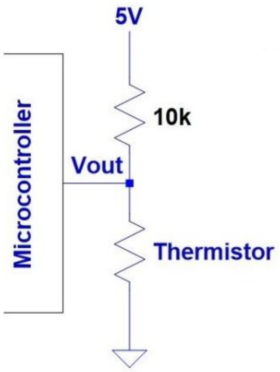


Figure 5 - Temperature Measurement Circuit to be used to Measure Ambient Temperature before Commencing Flow Speed Measurements

5 *Genio*Box Design

The following section will provide details on *Genio*Box, an electronic box that controls and regulates the probe. In addition, it also communicates with *Unisca*Suite, a software package that interacts between operators and regulates overall functions of the system.

5.1 Overview

*Genio*Box will consist of an enclosure, a microcontroller, a printed circuit board, and a power regulator. All parts will be located inside an enclosure. The enclosure will only have minimal connectors, switches, and light indicators. The material that is made up an enclosure will be stainless steel as it does not stain, corrode, or rust as easy as the other metal counterparts. Those characteristics will make the *Genio*Box easy to be cleaned. Addition shielding might be needed to shield electromagnetic interference that could affect its function. Although stainless steel is considered metal, it has low magnetic permeability and thus, cannot effectively block magnetic interference.

5.2 Microcontroller Consideration

Microcontroller inside *Genio*Box must be able to perform as follows: First, it must have serial communication in order to communicate with the *Unisca*Suite software that is being installed inside a computer. Second, it must have an analog to digital converter, in order to read value off the *Picolo*Probe and process those data. Third, it must be able to run at a speed greater than 100 Hz according to function specification.

From all of these criteria, we picked a microcontroller board called Arduino Mega 2560, which contains ATmega2560 8-bit microcontroller from Atmel. The reason for this choice is that the microcontroller contains all the features required by the system: a universal asynchronous receiver/transmitter (UART) for serial connection, an analog to digital converter, a pulse width modulation controller, and a microcontroller speed of 16 MHz.

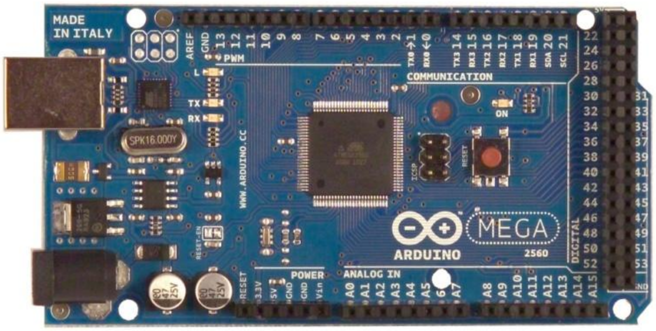


Figure 6 – Arduino Mega 2560

5.3 Power Supply Consideration

The *Genio*Box will be supplied by a 24 volts wall-wart adapter. This power supply will be shared between the *Piccolo*Probe, the current source, and the microcontroller. The *Piccolo*Probe and the current source need 20 volts power supply; therefore, a buck converter is needed to transform 24 volts DC to 20 volts DC. The Arduino microcontroller can be powered by a range of voltage from 7 volts up to 12 volts. We will develop another buck converter (see Figure 7) to transform 24 volts DC to 9 volts DC.

The circuits have been designed using National WEBENCH® tool from National Semiconductor website. WEBBENCH gives you a wide range of solutions; we picked LM2678 as the buck power converter because it has an extremely high efficiency of 92%. Also, the total cost of all the components is cheaper than most converters available, and the output can easily be adjusted by changing resistor values.

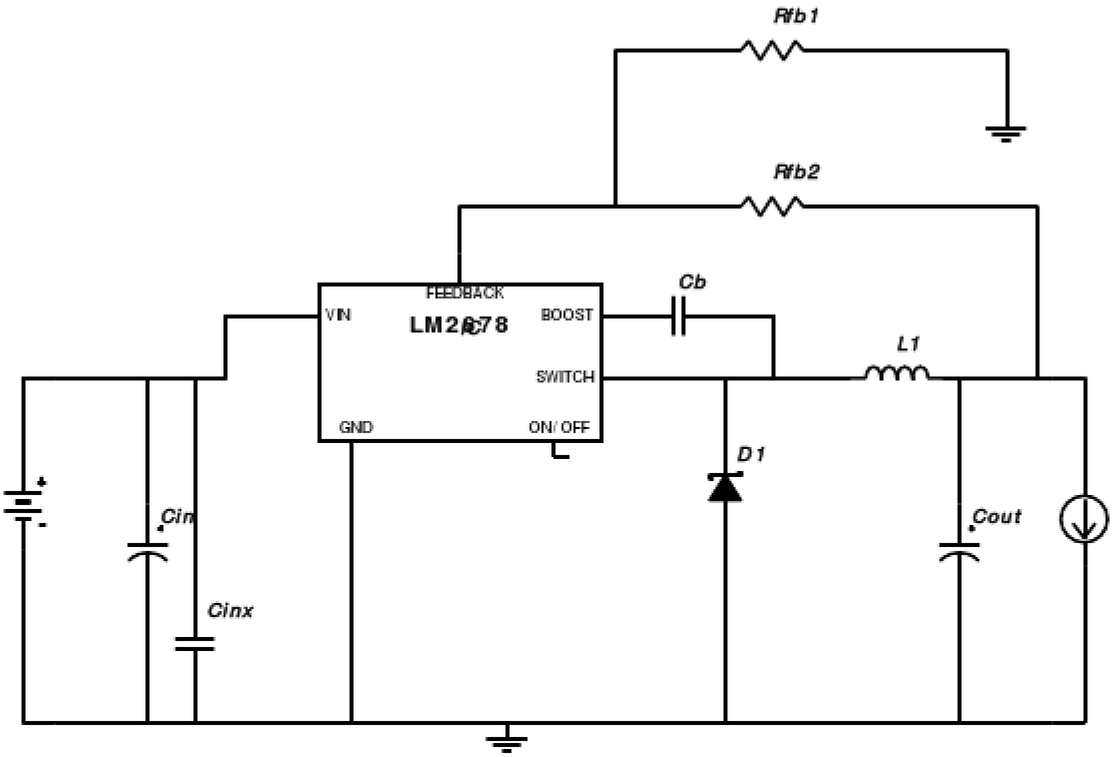


Figure 7 – Schematic of Power Supply Electronics Including a LM2678 Buck Power Converter

The specific values of resistors, capacitors, diodes, and inductors for 9 volts and 20 volts power regulators are located in Appendix A.

5.4 Flow meter circuit

The thermistor flow-meter circuit is illustrated in Figure 8. In the Wheatstone bridge configuration, R3 and R4 are known, and R1 is adjustable, such that it sets the operation temperature of the thermistor. The bridge is balanced (i.e., the error voltage is zero) when

$$\frac{R_3}{R_{thermistor}} = \frac{R_4}{R_1} \tag{2}$$

We will be operating the thermistor at 40.5°C, and at this temperature the thermistor will have a resistance of 9.1Ω. We chose the resistor R_3 to be 10Ω, which is slightly greater that the resistance of the thermistor. To maximize out circuit efficiency, we want to pump the majority of the current through the leg of the bridge containing the thermistor. As a result, R_1 was chosen to be 9.1kΩ, and R_4 was chosen to be 10kΩ to ensure the bridge will be balanced at the operation temperature. The nodes for measurement of the error voltage are the inputs to the UA741CN operational amplifier. The op-amp will operate on 18V VCC, and grounded VEE to ensure a positive output voltage. The current source, I, provides the necessary current to heat up the thermistor such that its resistance decreases to the value at which the bridge is balanced. At this point, the error voltage and the feedback current provided from the op-amp will be zero.

To calibrate the circuit, the probe will be heated to the operational temperature of 41°C, and R1 will be adjusted such that the error voltage is zero. When fluid flows past the thermistor, the temperature will decrease, its resistance will increase, and there will be a non-zero error voltage. The op-amp will produce a feedback current to balance the bridge. The output voltage will be measured and used to calculate fluid flow.

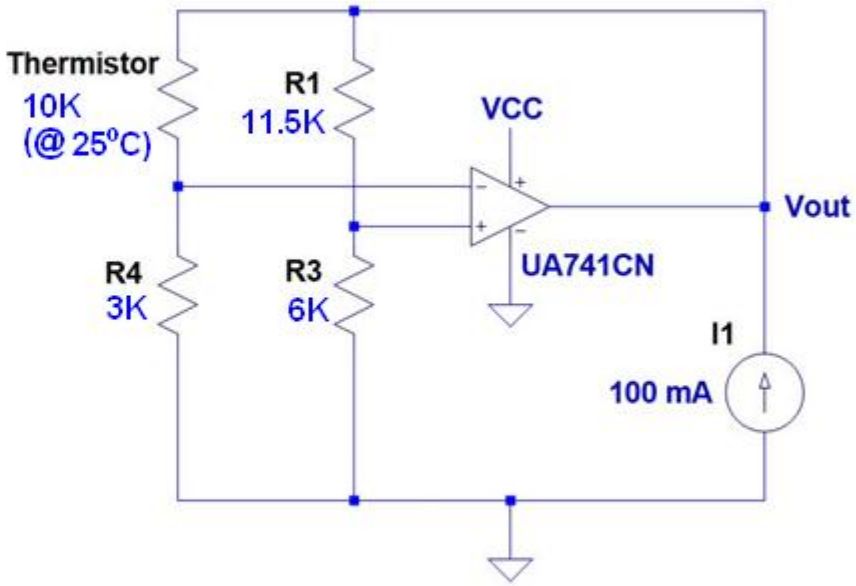


Figure 8 – Wheatstone Bridge Circuit For Maintaining a Constant Temperature During Flow Speed Measurements

The constant current source will be provided by a voltage regulator, which provides a 100mA output current to the load. Figure 9 illustrates the constant current circuit.

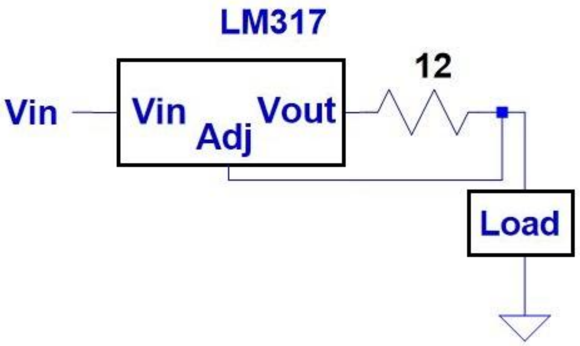


Figure 9 – Constant Current Circuit for Application to the Output of the Wheatstone Bridge Circuit

5.5 Using Voltage to Determine Fluid Speed

Assuming that the thermistor is in thermal equilibrium with its environment, the electrical power input is equal to the heat transferred by convection,

$$\frac{V^2}{R_t} = \hbar A_t \left(T_t - T_f\right) \qquad (3)$$

where V is the voltage across the thermistor, R_t is the resistance of the thermistor, h is the heat transfer coefficient of the thermistor, A_t *is* thermistor surface area, T_t is the temperature of the thermistor, and T_f is the temperature of the fluid. The difference between the thermistor temperature and the fluid temperature is

$$(T_t - T_f) = \left(\frac{R_t - R_0}{R_0 C}\right)$$

whereR_0 is the thermistor's resistance at the fluid's temperature, and C is the temperature coefficient of resistance of the thermistor. Therefore the heat transferred is

$$\frac{V^2}{R_t} = \hbar A_t\left(\frac{R_t - R_0}{R_0 C}\right) = (R_t - R_0)\big(X + Y\sqrt{U}\big)$$

Where U is the speed of the flow, the constant X is given by,

$$X = \frac{0.42kA_t}{R_0 Cd}\left(\frac{mC_p}{k}\right)^{0.2}$$

and the constant Y is given by,

$$Y = \frac{0.57kA_t}{R_0 Cd}\left(\frac{mC_p}{k}\right)^{0.33}\left(\frac{rd}{m}\right)^{0.5}$$

In these equations, m is the dynamic viscosity of the fluid, k is the thermal conductivity of the fluid, r is the gas density, C_p is the specific heat of the fluid, and d is characteristic length. Therefore,

$$V^2 = R_t(R_t - R_0)\big(X + Y\sqrt{U}\big).$$

Since the voltage across the thermistor is proportional to the output voltage of the op-amp, and by absorbing the resistances into the constants, this expression can be simplified to give

$$V_{out}{}^2 = A + B\sqrt{U} \qquad (4)$$

where V_{out} is the voltage measured at the output of the op-amp as shown in Figure 1. The constants are then determined experimentally during calibration.

5.6 Buffer Circuit

The Arduino analog to digital converter (ADC) can accept a maximum voltage of 5 V, but the wheatstone bridge circuit, shown in Figure 8, can output a maximum voltage of 20 V. In order to protect the microcontroller from excessive voltage input a voltage divider with a 4:1 ratio is implemented. A buffer is required before the voltage divider in order to ensure it will not interfere with the current source and cause instability.

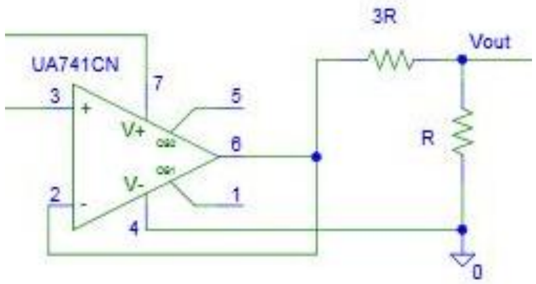


Figure 10 – Voltage Divider and Buffer between the Wheatstone Bridge Output and Microcontroller ADC

5.7 Safety Considerations

In order to connect the electronic components inside the *Genio*Box, the microcontroller has to perform additional duties. For safety reasons, the Arduino must be able to shut off the power to the probe and current source in the case where the probe is disconnected. Two digital pins will be used to signal those components to shut down.

5.8 Accuracy

The microcontroller operates on 0-5V and the ADC pin is 10 bits. Therefore, the resolution of the microcontroller can be calculated as 4.88 mV. Uncertainty in the thermistor will result in an uncertainty in the output voltage from the op-amp, and therefore an uncertainty in the final calculated flow speed. This error is attributed to the uncertainty in the β value, and the given resistance for the thermistor at 25°C, R_{25}.

Table 1 – Accumulated Uncertainties in Flow Speed Calculation

Parameter	Uncertainty
ADC resolution	4.88 mV
β value	3040 ± 3%
R_{25}	15 ± 2%

6 UniscaSuite Design

The following section describes the design choices and characteristics for the *Unisca*Suite software package. The *Unisca*Suite package provides a GUI allowing the user to control the operation of the *Vivace*Flow system using a Personal Computer. It will provide facilities to set device parameters, monitor device readings, and log device readings.

6.1 Overview

The *Unisca*Suite will be programmed in C#. Using C# provides us with the .NET library which provides use with classes which simplify of our main design challenges: graphing the data, building a user interface, and communicating via the COM port. An advantage of .NET solving all these issues is we do not have to mix and match libraries for each which avoids possible conflicts between libraries. Additionally, .NET is quite commonly already installed on Windows systems, reducing the footprint of our software. We still should be prepared to install .NET in case it is missing.

In the production version, we plan to port *Unisca*Suite to OS X and Linux. The Mono project provides .NET support for OS X and Linux. This would simplify porting to these environments. We would merely have to re-test our software package and adjust our user interface based on the conventions of each system. To contrast, if we had used different libraries, we would need to find new libraries on each system and due to this maintain three separate applications.

The software uses an event based design. The class managing the serial port connection will create events as it reads data off the serial port, and these events will propagate the data to the user interface and the log file.

6.2 Main Classes

The following subsections examine the main classes of the suite. We follow the C# convention that all classes appear on their own in a file of the same name with the extension .cs, unless otherwise specified. If we had a class named Example, it would be in the file Example.cs with no other classes.

6.2.1 Program

The Program class is responsible for initializing the program. It will provide the Main method. Main's first task is to verify only one instance of the application is being used. To achieve this, Program will employ a named mutex. If the program is able to get a lock on the mutex, then the application is unique. If it is unable, then another instance is running and main will simply quit. Using a named mutex will make the uniqueness lock robust. Since it is being handled by the system, it should not break due to application error nor should an application crash cause the lock to persist. Main will instantiate the main window (class: MainForm), the log file backend (class: Logger), and the serial port interface (Class: SerialBackend). It will then connect the event generators and handlers of these classes. Then Main will run the main program loop.

The class itself will not be instantiated nor will it have any other methods.

6.2.2 SerialBackend

SerialBackend is a class that will manage all communication through the serial port. Every time data is received from the COM port, SerialBackend will unpack the data and interpret it. When data needs to be sent to the COM port, the backend will pack the data and send it. The data format used is described in Section 7.1.1. For reference, data packets will contain a header, a value, and the inverse of the value.

The data received will be voltage measurements from the circuitry contained within the *Genio*Box. Serial-Backend will convert these voltage measurements into flow speed readings using Equation 4. The constants A and B will be stored one per line in the text file with extension: ~/My Documents/VivaceFlow/Liquids. An example of how these constants will be stored is as follows:

```
Liquid Name: Liquid_temperatureA_valueB_value
```

To choose the right calibration constants, the software will need to know the temperature of the liquid in which the flow is being measured. The *Genio*Box will have to measure the temperature every time the fluid is changed. Serial-Backend will take care of requesting the current temperature.

The class will provide functions to allow data to be sent by from the interface to the probe. Since the other classes should be clueless to the packet format, the functions contained in this class should be quite specific such as:Calibrate(Fluid), StartProbing(), and StopProbing(). The methods themselves will have to manage the details of creating a packet with the appropriate data.

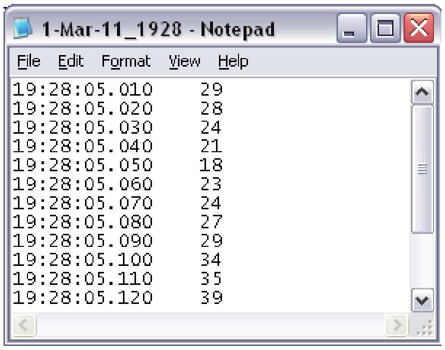
This class will use events to notify the other classes of data from the COM port, and it will prepossess all the data from the COM port. Separate events will be used to notify that new data has arrived at the serial port, such as speed data, temperature, commands and the various error states of the probe.

One final duty of Serial-Backend is to occasionally ping the *Genio*Box. This ping is necessary to allow the *Genio*Box to know it still has an active connection to the *Unisca*Suite, and thus does not have to perform an emergency disabling of the *Piccolo*Probe. Since the *Genio*Box should always be sending data at 100Hz, it is not necessary for the *Genio*Box to ping the software suite.

6.2.3 Logger

The Logger class provides functionality for creating and writing to log files. For each session that is run, Logger will create a unique log file. For uniqueness, each log file will be named Date_Time.log (eg. 1-Mar-11_1840.log). Since only one instance of the application can exist at a time, using a time and date stamp will provide unique file names for 100 years. The log files will be stored in the user's My Documents folder in a subfolder named VivaceFlow\Logs.

The each line of the log file will contain a timestamp and a flow measurement reading, separated by a tab. The time stamp will be in 24-hour format, and have a resolution to the millisecond. The speed reading will be in centimetres per second. An example log file is shown in Figure 11.



```
19:28:05.010	29
19:28:05.020	28
19:28:05.030	24
19:28:05.040	21
19:28:05.050	18
19:28:05.060	23
19:28:05.070	24
19:28:05.080	27
19:28:05.090	29
19:28:05.100	34
19:28:05.110	35
19:28:05.120	39
```

Figure 11 – Example Log File produced by the *Unisca*Suite

Logger will respond only to flow speed events from Serial-Backend. Each event it receives will cause it to write a line to the log file. To reduce the dangers of keeping a file open, Logger will only have the log file kept open in an open state for as long as needed to perform a write; any pre-processing and formatting will occur before.

6.2.4 MainForm

MainForm will provide the main GUI window for the application, and will use Windows Forms. Figure 12 shows a user interface mock-up.

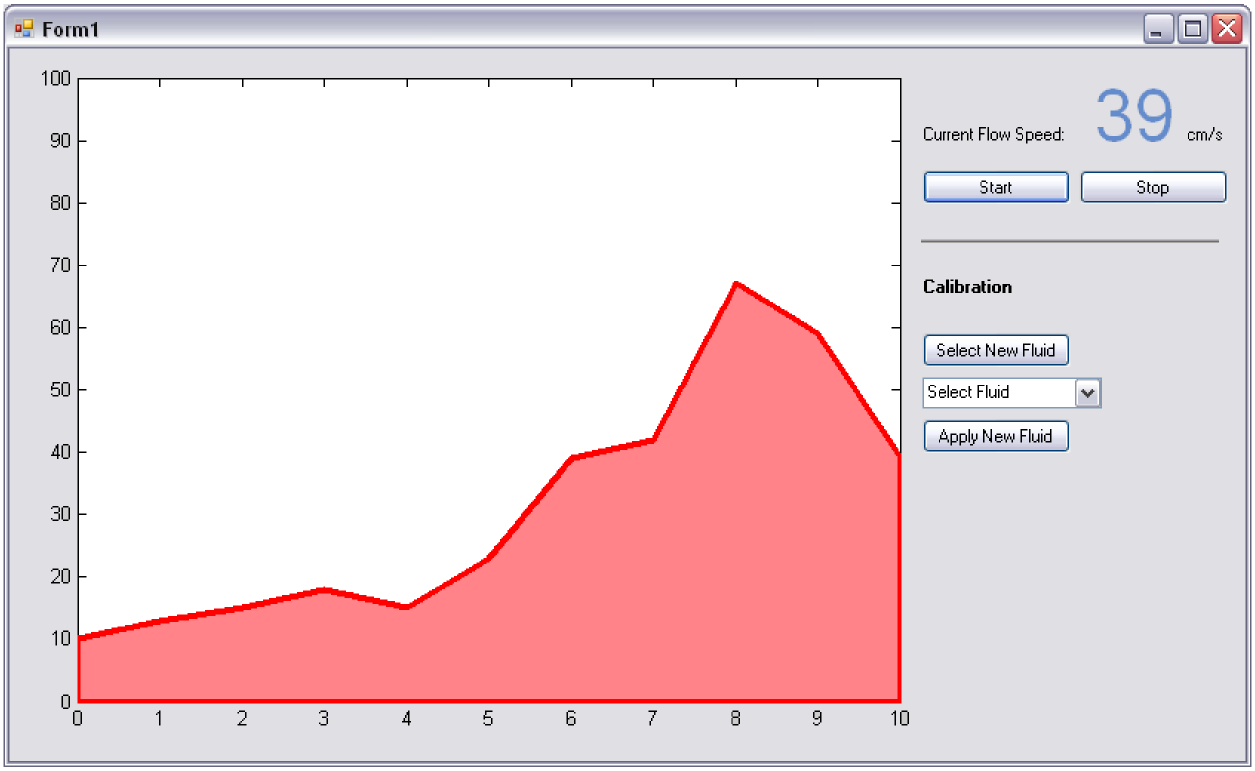


Figure 12 – User Interface for the *Unisca*Suite

The buttons marked Start and Stop will start and stop, respectively, flow speed measurements from the probe. When pressed, Serial-Backend will send a message to the *Genio*Box to start or stop flow speed measurements. To change the fluid, the user uses the controls under the “Calibration” header. The user presses “Select New Fluid” to enable the drop down box, selects a fluid with the drop down box, and then finally clicks “Apply New Fluid”. While not shown in Figure 12 for clarity purposes, the buttons and dropdown box will be disabled when appropriate to prevent the user from clicking buttons out of sequence. The “Start” button will only be enabled when the probe is stopped, and there is a fluid selected. The “Stop” button will only be enabled when the probe is gathering data. The “Calibration” controls will only be available when the probe is stopped. The “Select Fluid” drop box will be disabled until the user presses the “Select New Fluid” button. Pressing the “Select New Fluid” button will disable the “Start” button, as a fluid will not be selected anymore. The user can select a fluid with the drop down box, and then press “Apply New Fluid”. The system will select the new fluid and enable the “Start” button. The intent is to make the GUI such that a user cannot try to perform flow measurements without a fluid selected, and to prevent the GUI from displaying that a fluid is selected without it actually being selected.

MainForm will respond to events from Serial-Backend of all types. For flow speed events, it will update the “Current Speed Reading” and the flow graph. For error message type events (i.e., the *Piccolo*Probe is disconnected from the *Genio*Box), an error message will be displayed to the user. While a message box seems fitting, it would interfere with the user's ability to quickly disable the probe, which may very well be necessary. Instead, a warning sound will be played and an error message panel will be displayed across the bottom of the application window. With this system, error messages should give enough of a warning to catch user attention while not preventing use of the interface.

6.2.5 Properties

In version 4 of the .NET Framework, the framework itself is able to automatically perform per application settings. As such, there will be no Properties.cs; the framework will provide this for us. The framework provides a class, Properties::Settings, that will manage application settings in a standard way. Any settings that should need to persist through multiple sessions of the program will use this to store the setting.

7 System Integration

This section outlines design decisions regarding the integration of separate parts into a complete system, and will discuss the interfaces between components and the reasoning behind them.

7.1 UniscaSuite and GenioBox

The *Unisca*Suite and *Genio*Box will communicate via a COM port. The COM port interface is rather simple, but sufficient for our purposes. COM ports are commonly integrated onto microcontrollers via UART chips, which allow us to easily select a microcontroller. Furthermore, the simplicity of programming the COM port will simplify development of the software for the microcontroller, and communication programming will be much simpler for the *Unisca*Suite, making it a secondary concern.

The *Genio*Box will send probe data to the *Unisca*Suite to be logged every 0.01 seconds to satisfy requirements [FS-I-09-A] and [FS-I-10-A]. To satisfy requirement [FS-S-21-A], the *Unisca*Suite will ping the *Genio*Box once per second. If the *Genio*Box does not receive a ping after 2 seconds, it should assume that the *Unisca*Suite has somehow been disconnected, and will shut off power. Other command and error packets will be sent as necessary. Since each component is periodically checking for data, error and commands should be dealt with during this period (100Hz for *Unisca*Suite and 1Hz for *Genio*Box).

7.1.1 Packet Format

For the *Unisca*Suite and *Genio*Box to communicate, a packet format must be agreed upon. Our packet format will consist of three characters per packet. The software sending and receiving a packet should consider each three characters as a single, discrete packet: information should not be split across two packets.

The first character of the packet will serve as a header. This serves two roles. The header is it will prevent noise from being interpreted as the beginning of a packet. Since we will have two different headers, this would mean there is 2 in 256 chance of a noise character being considered as the beginning of a packet. The second purpose of the header is to indicate how the data should be interpreted. The first header, SOH (0x15), will indicate flow speed readings from the probe. The other header, EOR (0xEA), will indicate the data is a command or an error code. It is worth noting that SOH is the inverse of EOR. This will minimize the chance of a SOH corrupting into an EOR, and vice versa.

The next character will be the actual data being sent. In the case of flow speed readings, it will simply be the flow speed cast to a character. Since there are 256 characters, this would support data values of up to 256 cm/s. For commands and error messages, Table 2 shows the meaning of each value. Generally low values indicate error messages, while high values indicate commands.

The last character will be the inverse of the previous character, which will be used to ensure data integrity. Upon receipt, the receiver should check that the second and third characters are inverses of each other, and packets that do not pass this test should be discarded. For data and command packets, successful receipt should be indicated by sending an ACK, and unsuccessful packets should be indicated with a NAK. The sender should be prepared to resend the last command or data packet if it receives a NAK within a three second timeout. ACKs and NAKs will not be sent for flow speed measurements since the implementation cost outweighs the benefits.

Table 2 –Error and Command Codes in Packet Format for Serial Communication

Hex Code	**Meaning**
0x01	Error: The probe is not attached to the *Genio*Box
0x02	Error: The probe is in a liquid outside its safety range
0xA0	PING: *Unisca*Suite periodically pings *Genio*Box to show an active connection.
0xA1	ACK: Acknowledges the last command or error message.
0xA2	NAK: Not-acknowledge. Indicates the last command or error message was not received.
0xE0	Calibrate the probe for blood
0xE1	Calibrate the probe for saline
0xF0	Start taking flow readings from the probe.
0xF1	Stop taking flow readings from the probe.

7.2 GenioBox and PiccoloProbe

In order for the *Piccolo*Probe to be at all operable, it will need to be supplied power from the *Genio*Box. The *Genio*Box will have to supply the probe with +18 Volts, and a connection to ground. The *Genio*Box will also be required to supply a current to the probe. The current is constant, but currently unknown. The *Genio*Box will have to be able to measure a current from the probe. As shown in detail in equation 3 (Section5.5), A and B will have to be determined experimentally. They will have a strong dependence on the shell of the head, the resistors in the circuit and the wires. Microcontroller programming will have to be done in a way such that the actual values for A and B can be programmed in later.

The wires connecting the probe to the *Genio*Box will be concealed in a singular casing. The wires will be fed into the box via a small hole with a grommet.

7.3 PiccoloProbe Head and Body

As the probe head and body will be developed by two separate teams, it is necessary to discuss the connection between the two.

The body of the probe will connect to the head via a flexible and hollow tube. Wires from the tube would be able to be thread through the tube, allowing us to meet requirement [FS-P-05-A]. The head of the probe will have a maximum diameter of 5mm. Thus we will need the hollow tube to be less than 5mm in diameter so we can seal the two pieces together.

8 System Test Plan

Our system test plan will begin with individual unit testing of our main subsystems: the *Piccolo*Probe, the *Genio*Box, and the *Unisca*Suite. These tests will target unit functional requirements which have no correlation with other components of the device. The procedure for each of these unit tests are identified in Section 8.1 below, along with the functional requirement being verified and their pre- and post-condition.

Due to the integration of the system, and the reliance of each component on a subsequent components output, the testing of the entire V*ivace*Flow device will be performed over stages of increasing complexity. These tests will occur after unit testing is complete, and will be monitored by the use of a testing device which is presented in Section 8.2. This testing device will be rigged with a flow speed measurement device which has higher measurement accuracy than required by the *Vivace*Flow system. This will be used to both calibrate the device by identifying the unknown coefficients from Equation 3, and to confirm the valid integration and measurement data made by our device to within the required accuracy.

Benchtop testing, as described above, will be performed in two types of solutions. The first is an isotonic saline solution, used for basic flow property testing. The second will be a mock blood solution, referred to as “blood saline”, which will have a similar specific heat capacity and viscosity to blood. Synthesis of this solution is described in Section 8.3.

The remainder of this section outlines how we have determined adequate testing to be performed for validating the specifications outlined in the functional requirements. We also describe the drawback of benchtop testing, and the need to continue to *in vivo* testing for further validation.

8.1 Unit Testing

The following section identifies the test cases that will be performed in order to verify the functionality of each individual component. The Tests names are identified by the following code:

[DS-T-##]

where ## identifies the test number within the design specification document. For each test case the functional requirements that are being validated by the completion of the test are identified. The preconditions, procedure which will be following and post conditions that must be satisfied are also identified.

8.1.1 *Piccolo*Probe

Test Name:	[DS-T-01] - Sheath Insertion
Requirements:	[FS-P-02-B] [FS-P-09-A] [FS-P-19-B]

Preconditions:	None
Procedure:	1. Insert the probe through the end of the sheath. 2. Close the homeostatic seal as required
Post Conditions:	The probe fits comfortably into the sheath and produces a seal against the inner wall of the sheath which is impermeable to fluid transfer. The probe is free to move forward and back in the sheath without breaking the seal

Test Name:	[DS-T-02] - Imaging
Requirements:	[FS-P-08-A]
Preconditions:	None
Procedure:	1. Insert probe head a body into gelatinous phantom used to mimic the density qualities of the body. 2. Use ultrasound machine the image through the gelatinous material to the probe 3. Use a fluoroscopy machine to image through the gelatinous material to the probe
Post Conditions:	The probe and body must be easily identifiable on the fluoroscopy and ultrasound image by the naked eye of an untrained user

Test Name:	[DS-T-03] - Steering Control
Requirements:	[FS-P-07-A] [FS-P-10-C] [FS-P-11-C]
Preconditions:	The probe head must be in a unbent position
Procedure:	1. Turn knob of steering controls to bend the head of the probe to its maximum position 2. Remove your hand from the steering control 3. Put your hand back on the steering control and turn the knob in the other direction until it cannot be turned any further 4. Repeat the steps 1 to 3 in saline, and again in the mock blood solution
Post Conditions:	The head must bend back to within a distance of 10 mm from the body. The head must remain in its bent position when the knob is released in all solution types. The head must return to its original position in a controlled manner by the knob. The controls must keep the user from continuing to turn the knob once the head is back to its fully straight position.

Test Name:	[DS-T-04] - Voltage location sensing system
Requirements:	[FS-P-14-A] [FS-P-15-A]
Preconditions:	None

Procedure:	1. Kardium to take the *Piccolo*Probe and determine the location of the probe on a 3-D voltage sensing system
Post Conditions:	The location of the probe must be determinable and the probes exterior must not interfere with the signal.

Test Name:	[DS-T-05] - Liquid Leakage
Requirements:	[FS-P-17-A] [FS-P-22-B]
Preconditions:	None
Procedure:	1. Insert the probe head and body casing into an isotonic saline solution 2. Leave the probe in the solution for one hour 3. Remove the probe from the solution 4. Repeat steps 1 to 3 with the mock blood solution and isopropyl alcohol
Post Conditions:	The interior of the probe must remain completely dry after one hour of exposure

8.1.2 GenioBox

Test Name:	[DS-T-07] - Collection and Distribution of data
Requirements:	[FS-B-07-A] [FS-B-08-A]
Preconditions:	None
Procedure:	1. Hook up the input of the *Genio*Box to a function generator 2. Hook up the output of the *Genio*Box to an oscilloscope 3. Set the generator to a frequency of 110Hz 4. Turn the GenioBox on 5. Monitor the output of the *Genio*Box on the oscilloscope
Post Conditions:	The collection and distribution of data must be successful at frequencies above 110Hz

Test Name:	[DS-T-08] - Downtime
Requirements:	[FS-B-10-A]
Preconditions:	The *Genio*Box is off
Procedure:	1. Turn the *Genio*Box on 2. Leave idle for 24 hours 3. Hook up a function generator to the output and oscilloscope to the input 4. Turn both the devices on and leave on for 24 hours
Post Conditions:	The *Genio*Box must remain active for the entire 48 hours. The output signal integrity must remain the same as at the beginning of the 24 hours.

8.1.3 *Unisca*Suite

Test Name:	[DS-T-09] - Installation
Requirements:	[FS-I-07-B]
Preconditions:	A computer with a 'fresh install' of Windows XP.
Procedure:	1. Insert USB key containing installer. Use file manager to locate setup.exe on the USB key. Run setup.exe. 2. Follow instructions provided in installation Wizard. Remove USB after the wizard completes.
Post Conditions:	The user is able to find icons for the software on the desktop and the start menu. The software packages will run when these icons are clicked.

Test Name:	[DS-T-10] - Multiple Instances of Software
Requirements:	[FS-I-04-A]
Preconditions:	None
Procedure:	1. Run the software suite on a Windows computer 2. Attempt to run the suite again, while the original is still open
Post Conditions:	The original suite must stay open, and a second instance must not open

Test Name:	[DS-T-11] - Collection and Distribution of data
Requirements:	[FS-I-09-A] [FS-I-10-A] [FS-I-11-A]
Preconditions:	The software suite is running on a windows computer
Procedure:	1. Use null terminal program to send data to the serial port 2. Commence flow data measurements 3. Track the collection of data in the log file 4. Track the displaying of data on the screen
Post Conditions:	The data must be able to collect at a frequency of at least 100Hz and the display screen must be refreshed at 10Hz

Test Name:	[DS-T-12] - Log File Corruption
Requirements:	[FS-I-17-A]
Preconditions:	The software suite is running on a windows computer
Procedure:	Commence flow data measurements Use task manager to force quit *Unisca*Suite Verify log file contents Re-run *Unisca*Suite software
Post Conditions:	The data from the suit before the crash must be retained, and

	contain all the information up to the point just before the crash. The data from the new suit instance must start logging normally.

Test Name:	[DS-T-13] - Simulation of Error Reception
Requirements:	N/A
Preconditions:	The software suite is running on a Windows computer. A null terminal program is running on the same computer.
Procedure:	Use the null terminal program to send an error code. Observe post conditions. Repeat test for all error codes in Table Porn
Post Conditions:	The UniscaSuite displays an appropriate error message on screen for each error code sent. The null terminal program receives an ACK for each error message sent.

Test Name:	[DS-T-14] - Simulation of Error Reception II
Requirements:	N/A
Preconditions:	The software suite is running on a Windows computer. A null terminal program is running on the same computer.
Procedure:	Use null terminal to send numerous error codes in a serial fashion.
Post Conditions:	The UniscaSuite displays one error message for each error code sent by the null terminal program. The null terminal program has received one ACK for each error code sent.

Test Name:	[DS-T-15] - Simulation of Sending Commands with Response
Requirements:	N/A
Preconditions:	The software suite is running on a Windows computer. A null terminal program is running on the same computer.
Procedure:	1. Press Select New Fluid. 2. Select Blood. 3. Press Set New Fluid 4. Use null terminal program to send an ACK. 5. Press Start. 6. Use null terminal program to send an ACK. 7. Press Stop. 8. Use null terminal program to send an ACK. 9. Press Select New Fluid. 10. Select Saline. 11. Press Set New Fluid. 12. Use null terminal program to send an ACK.
Post Conditions:	Observe null terminal program has received 0xE0, 0xF0, 0xF1, and 0xE1. Each code should be received one time.

Test Name:	[DS-T-16] - Simulation of Sending Commands without a Response
Requirements:	N/A
Preconditions:	The software suite is running on a Windows computer. The suite has a fluid selected. A null terminal program is running on the same computer.
Procedure:	1. Press Start.
Post Conditions:	Observe null terminal program receives 0xF0 twice. UniscaSuite displays error message indicating it has lost a connection to the GenioBox.

Test Name:	[DS-T-17] - Simulation of Sending Commands with a NAK Response
Requirements:	N/A
Preconditions:	The software suite is running on a Windows computer. The suite has a fluid selected. A null terminal program is running on the same computer.
Procedure:	1. Press Start. 2. Use null terminal program to send a NAK. 3. Use null terminal program to send an ACK.
Post Conditions:	Observe null terminal program receives 0xF0 twice. The second 0xF0 should come after the null terminal sends the NAK. Observe no error is displayed in UniscaSuite.

8.2 Testing Device

In order to perform both calibration and testing of the *Vivace*Flow device, a system needs to be built which has the ability to produce both constant and pulsatile flow, to accurately measure the flow speed, to provide easy access for the probe into the flow and to control temperature. As a result, a testing device was designed which uses a high performance pumping system and simple and accurate flow measuring method to satisfy our requirements. The following section outlines the requirements of the testing system followed by the overall device design and justification for each component choice.

8.2.1 Requirements

The range of fluid flow speed detectable by the VivaceFlow system is required by [FS-S-25-A] and [FS-S-26-A] to have an accuracy of the largest of 10% and 2.5cm/s for flow speeds between 0 and 1m/s (Steiner, Drewbrook, Naziripour, Gosling, & Sahachaiwatana, 2011). This means that flow from 25cm/s to 1m/s must have an accuracy of at least 10% and flow speeds from 0m/s to 25cm/s must have an accuracy of 2.5cm/s. To determine the specifications of the testing device, the use of the fluid flow formula is required. The formula states that the flow rate, Q, is defined as

$$Q = AS$$

where S is the speed of flow and A is the area cross-sectional area within which the liquid is moving (Holland & Bragg, 1995). Given a pipe with a 1/2 inch diameter (0.00635 m radius), the cross sectional area of the pipe is found to be 0.0001267m^2. From a maximum speed of 1m/s it is possible to calculate the maximum flow rate through the defined tube as,

$$Q_{max} = (0.0001267\ m^2)\left(\frac{1.0\ m}{s}\right) = \frac{0.00013\ m^3}{s} = \boxed{7.8\,\frac{\mathrm{L}}{\mathrm{min}}.}$$

Also, given a minimum of 25cm/s at 10% accuracy, it is possible to calculate the minimum flow rate as,

$$Q_{min} = (0.0001267\ m^2)\left(\frac{0.25\mathrm{m}}{s}\right) = \frac{0.000032\ m^3}{s} = \boxed{1.92\,\frac{\mathrm{L}}{\mathrm{min}}.}$$

Based on these calculations, the requirements for the flow measuring device are presented in Table 3 below. For each required parameter, the functional requirement of the device, to which the testing device must meet or exceed, is also provided.

Table 3 – Required Parameters of Flow Sensor for Testing Device

Flow Sensor Parameter	Required Values	Functional Requirement
Flow range	1.92 to 7.8 L/min	[FS-S-25-A]
Accuracy for the given flow range	Less than ± 10%	[FS-S-26-A]
Accuracy for flow less than 1.92 L/min	Less than ± 100%	[FS-S-26-A]
Output	Electronic output of flow rate with a temporal resolution of at least 10ms	[FS-S-29-A]
Maximum working temperature	At least 40°C	[FS-P-16-A]
Minimum working temperature	At least 35°C	[FS-P-16-A]
Acceptable Fluids	Water and Saline having low to medium viscosity	[FS-S-27-A]

8.2.2 Design Overview for Constant Flow Testing

Presented in Figure 13, below, is a diagram showing the complete testing device design. The system consists of four components: an inflow bath, a fluid pumping system, an outflow bath and the tubing to connect them. The fluid pumping system provides the forced outflow of liquid which travels through the tubing to the outflow bath. The flow rate of the pump can be controlled manually by the use of a rotameter. A rotameter does not provide the accuracy required by [FS-S-26-A] to validate the device performance, so an alternate method was devised. This is described in greater detail in Section 8.4 below. The pumping system also has the ability to regulate the temperature of the fluid. As the fluid is sucked into the pump system from the inflow bath, it gathers in a basin where it is warmed. Saline from this basin is pushed through the tubing, which the *Piccolo*Probe is placed inside for flow measurements.

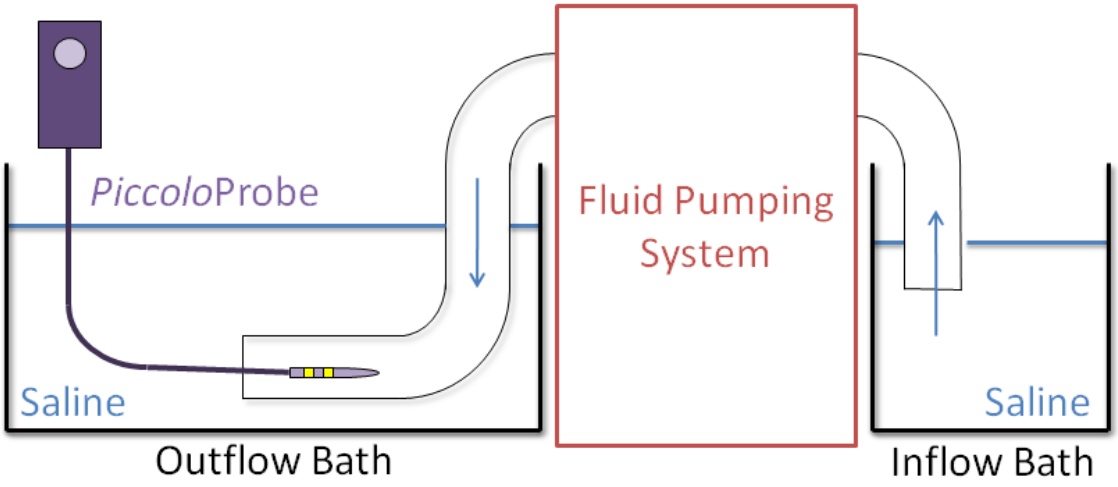


Figure 13 – Schematic of the Constant Flow Testing and Calibration System

8.2.3 Tube Diameters

Given the intended use of the product, the testing system will be designed to have similar setup to its intended use within the vessels of the heart. Within the heart, the blood is supplied to the left and right atria via the pulmonary veins and vena cava, respectively. Flow within these vessels is fairly laminar, and can be approximated with the flow through a tube of similar diameter. To determine the appropriate size of tubing to be used in the design, the approximate cross sectional diameter of the pulmonary veins will be used. Research done by Kim, Marom, Herndon, & McAdams(2005) found that the average cross sectional diameter at the pulmonary venous ostium of the Right and Left Superior Pulmonary Vein and Right and Left Inferior Pulmonary Vein is 11.9mm, 10.0mm, 12.7mm and 9.4mm, respectively. Given these results, it is determined that an appropriate pipe size for use is one with an inner diameter of half an inch (12.7mm) as it is a commonly used and easily accessible pipe size, and it represents average diameter of the largest pulmonary vein.

8.2.4 Pulsatile Flow Testing

The constant flow test cases are sufficient for the calibrations of the device, and testing its accuracy. Due to the environment in which our device will be used, it is necessary to test the device under similar flow conditions. In these conditions, due to fast changes in flow speed response time is of vital importance. As a result, we will measure the speed of pulsatile flow to ensure our device is responsive enough and that the temporal resolution is less than 10 ms. We have access to use a pulsatile flow machine through our investors, which will satisfy these requirements.

8.3 Benchtop Testing Liquids

The major problem presented during testing is the inability to test the *Vivace*Flow device in the media for which it is intended - blood. Instead, we will be testing the device in a fluid with similar physical properties. This fluid should have a similar specific heat capacity and viscosity to blood to ensure that the properties of blood which affect the flow speed measuring process are adequately represented. The specific heat capacity of blood is 3.78 J/g·K, and its viscosity is approximately 3.5 mPa·s.

The following table lists the ratio of the specific heat capacity of saline to the specific heat capacity of pure water, r, at the same temperature, for various concentrations of saline, s. The units for s are grams of salt per kilogram of water. According to Dorsey (1940) the specific heat of saline can be calculated by the following equation:

$$C = rC_o$$

where C_o is the specific heat of fresh water. Given a specific heat capacity for pure water of 4.2kJ/Kg^{o}C, and a desired heat capacity of 3.78 J/g·K for matching that of blood, the calculated ratio, r, is found to be 0.9.

Table 4 – Salinity versus ratio of Specific Heat between Salt Water and Pure Water at 17.5^{o}C(Dorsey, 1940)

s (g salt/ kg water)	r
0	1.00
5	0.982
10	0.968
15	0.958
20	0.951
25	0.945
30	0.939
35	0.932
40	0.926

In order to determine the concentration of saline for this requirement we plotted the above values and extrapolated to find the concentration of salt when r=0.9. We found this value to be 60.7g of salt per kg of pure water. As a result, the testing will be performed in two types of saline:“Isotonic saline”, having a concentration of 9g of salt per kg of water, and “blood saline”, having a concentration of 60.7g of salt per kg of water.

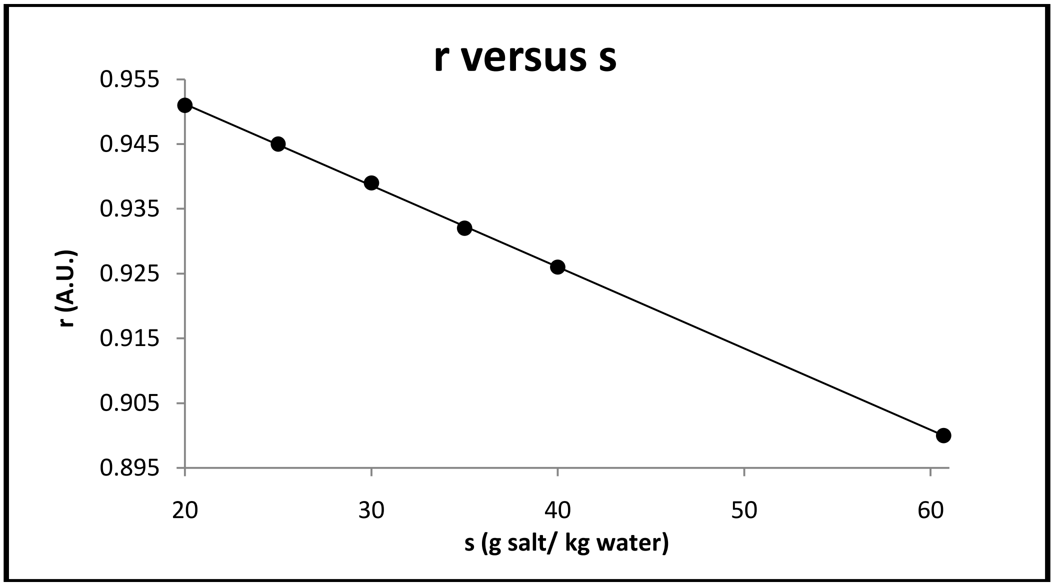


Figure 14 – Plot of ratio of Specific Heat between Salt Water and Pure Water (r) versus Salinity (s)

8.4 *PiccoloProbeCallibration*

Before testing of the integrated system can begin, the *Piccolo*Probe must be calibrated in order to obtain values for A and B in Equation 3. The calibration process will be performed with the testing system presented in Section 8.2. Before calibration starts the thermistor must be connected to the probe head and sealed. The thermistor is connected to the constant-temperature thermistor flow meter circuit in Figure 8 and the op-amp and current source are powered by a transformer from an electrical outlet. The output of the constant-temperature thermistor flow meter circuit is measured with an oscilloscope and the digital values are recorded using Matlab with a connecting PC.

The constant flow testing system will be setup with isotonic saline at 35.5°C. The head of the *Piccolo*Probe will be inserted into the end of the 12.7 mm cross-sectional diameter tubing. The pump will be set to produce flow at 10 cm/s. At the same pump is started, we will start a stopwatch. The pump will run until the inflow container is full, or for 100 seconds, whichever occurs first. We will measure the volume of saline in the container, and use this measurement along with the time measurement to calculate the volumetric flow. With the volumetric flow, we can calculate the linear flow speed simply be dividing by the cross-sectional area of a 12.7 mm tube. This process is repeated at 5.0cm/s increments up to 90cm/s. From these values, we will plot V^2 versus the square root of the flow speed, and a linear regression will be fit to the data to determine A and B from Equation 3. These steps will be repeated for saline at temperatures of 37^{o}C and 38.5^{o}C. Finally, this entire procedure will be completed again with "blood saline" to obtain three sets of calibration constants for use in blood.

As previously mentioned in the functional specifications, it is vital that these measurements are more accurate than our device, since they are being used for calibration and validation. All the measurements we make following this procedure will produce some error. The error in time measurements is typically taken to be 0.3s. This error occurs twice, resulting in a 0.6s error. Over a 100 second interval, this amounts to a 0.6% error. The measurements of the rectangular inflow bucket have an error of 0.05cm per end, for a total of 0.1 cm per measurement. Given the sides of the inflow bucket are roughly 30 cm, this is a 0.0333% error per side or 0.1% for the volume. Since measurement of the cross-sectional radius of the tube would be too erroneous if measured with a ruler, we will use a micrometer with an accuracy of 100 nm. This would give an estimated area measurement error of $1.6 \cdot 10^{-3}$ %. Since the measurements are all multiplied, we can simply add the percentage errors, for a total of 0.7016% error in the flow measurement.

8.5 Integration Testing Scheme

The integrated testing scheme is designed due to the reliance of each component on other component outputs of the device. The testing will therefore be performed over stages of increasing complexity, starting with the *Piccolo*Probe electronics. In order to achieve submersion in water for the integrated testing, testing and integration of the *Piccolo*Probe hardware must already be complete such that the thermistor can be introduced into the probe head and connected to the *Genio*Box. Below is each of the testing schemes, presented in chronological order ending in system completion.

Test Name:	[DS-IT-01]
Preconditions:	All tests from [DS-T-01] to [DS-T-06] are complete. The thermistor has been connected to the probe head and sealed. Calibration of the probe, as defined in Section 8.4, has been complete for both saline and blood. The thermistor is connected to the constant-temperature thermistor flow meter circuit in Figure 8. The Op-Amp and Current source are powered by a transformer from an electrical outlet.
Procedure:	1. Setup the Constant Flow Testing System with Isotonic Saline at 37^{o}C. 2. Set the flow rate to 7 L/min. 3. Measure the flow speed with the methods described in section 8.4. 4. Place the PiccoloProbe at the output of the pipe. 5. Measure the output voltage of the wheatstone bridge with

	the *Piccolo*Probe. Use Equation 3 and the calibration parameters to determine the flow speed. 6. Repeat steps 3 to 5 for flow speeds of 25cm/s, 50cm/s, and 75cm/s. 7. Start from 7L/min and repeat steps 3 to 6 five times 8. Determine the variability for each measurement
Post Conditions:	Repeatability for each of the flow measurements (25cm/s, 50cm/s and 75cm/s) is less than 10%. Difference from measured flow is less than 16%.

Test Name:	[DS-IT-02]
Preconditions:	Test from [DS-IT-01] is complete and results are within acceptable tolerance. The output of the constant-temperature thermistor flow meter circuit in Figure 8 is connected to the Buffer and voltage divider system in Figure 10, which in turn is connected to the ADC input of the Arduino Mega2560 Microcontroller. The Op-Amp and Current source are powered by a transformer from an electrical outlet. The output of the microcontroller is connected to the serial port of a PC.
Procedure:	1. Setup the Constant Flow Testing System with Isotonic Saline at 37°C. 2. Set the flow rate to 7 L/min. 3. Measure the flow speed with the methods described in section 8.4. 4. Place the PiccoloProbe at the output of the pipe. 5. Monitor the raw output flow speeds of the microcontroller on the PC. 6. Repeat steps 3 to 5 for flow speeds of 25cm/s, 50cm/s, and 75cm/s. 7. Start from 7L/min and repeat steps 3 to 6 five times 8. Determine the variability for each measurement
Post Conditions:	Repeatability for each of the flow measurements (25cm/s, 50cm/s and 75cm/s) is less than 10%. Difference measured values is less than 16%.

Test Name:	[DS-IT-03]
Preconditions:	Test from [DS-IT-02] is complete and results are within acceptable tolerance. The output of the constant-temperature thermistor flow meter circuit in Figure 8 is connected to the Buffer and voltage divider system in Figure 10, which in turn is connected to the ADC input of the Arduino Mega 2560 Microcontroller. The Op-

	Amp and Current source are powered by a transformer from an electrical outlet. The microcontroller is connected to the serial port of a PC and the *Unisca*Suite is running on the PC. The Suite controls of the power supply are disabled.
Procedure:	1. Setup the Constant Flow Testing System with Isotonic Saline at 37^{o}C. 2. Set the flow rate to 7 L/min. 3. Measure the flow speed with the methods described in section 8.4. 4. Place the PiccoloProbe at the output of the pipe. 5. Monitor the raw output flow speeds of the microcontroller on the PC. 6. Repeat steps 3 to 5 for flow speeds of 25cm/s, 50cm/s, and 75cm/s. 7. Start from 7L/min and repeat steps 3 to 6 five times 8. Determine the variability for each measurement
Post Conditions:	Repeatability for each of the flow measurements (25cm/s, 50cm/s and 75cm/s) is less than 10%. Difference from measured flow is less than 16%.

8.6 Typical Usage Testing

8.6.1 Constant Flow

Once integration of the *Piccolo*Probe, *Genio*Box and *Unisca*Suite is complete, typical usage testing will begin, consisting of both constant value flow testing and varying flow testing with the systems described in Section 8.2. The typical usage scenario is designed to mimic the use of the *Vivace*Flow so that we can test the overall functionality of the system.

1. Setup the Constant Flow Testing System with Isotonic Saline at 37^{o}C.
2. Start the *Unisca*Suite software on the connecting PC.
3. Power up the *Genio*Box.
4. *Genio*Box and *Unisca*Suite indicate a ready to use signal.
5. The user sets the type of fluid being used to be "Isotonic Saline".
6. The user inserts the *Piccolo*Probe at the output of the 12.7mm cross-sectional diameter tubing.
7. The user presses start to commence temperature readings for calibration parameters

8. Once temperature reading is complete, the flow speed measurements commence. Speed of the flow is plotted in real-time, and logged and saved automatically by the software.
9. Set the flow rate of the liquid pump to 7 L/min. Hold the flow rate constant for 1 minute. Change the flow rate to 6 L/min, and hold the flow rate constant for 1min. Continue to change the flow in this manner until reaching 1L/min.
10. The user stops flow speed collection by hitting the stop button.
11. The user removes the *Piccolo*Probe.
12. The user turns off the *Genio*Box
13. The user terminates the *Unisca*Suite.

To ensure the accuracy of our system, the measured flow with the methods described in section 8.4will be compared to the flow speed values logged by the Unisca*Suite*. The difference between them must be less than 16%. The values collected over the minute of constant flow speed will be compared for variability. Once complete the above 13 steps will be repeated for ‘”blood saline”.

8.6.2 Pulsatile Flow

The pulsatile flow test is the scenario that will truly test the performance of the *Vivace*Flow system for *in vivo* conditions. The above thirteen steps must therefore be repeated with the pulsatile flow testing system and isotonic and blood saline at 37°C.

8.7 Improper Use Testing

As with all medical equipment, safety is of paramount importance. As such it is necessary that we have sufficiently tested improper usage and failures to ensure that the system does not produce potentially dangerous outputs. This section should be considered living: it will be added to and refined as we discover new potential failures.

Test Name:	[DS-IT-04] – *Piccolo*Probe is disconnected
Requirements:	N/A
Preconditions:	The system is running under normal operation
Procedure:	1. The *Genio*Box is disconnected from the *Piccolo*Probe
Post Conditions:	The *Genio*Box detects that the *Piccolo*Probe has been disconnected and the warning LED on the *Genio*Box lights up. *Unisca*Suite then shows a warning that the *Piccolo*Probe has been disconnected.

Test Name:	[DS-IT-05] – GenioBox loses connection to the UniscaSuite
Requirements:	N/A
Preconditions:	The system is running under normal operation

Procedure:	1. The GenioBox is disconnected from the UniscaSuite by a disconnection of cable, a software crash, or the user closes the software
Post Conditions:	The GenioBox detects the connection loss and its warning LED lights up. The GenioBox stops sending power to the PiccoloProbe. If the UniscaSuite is still running, it will display a warning, which indicates that there is no connection to the GenioBox.

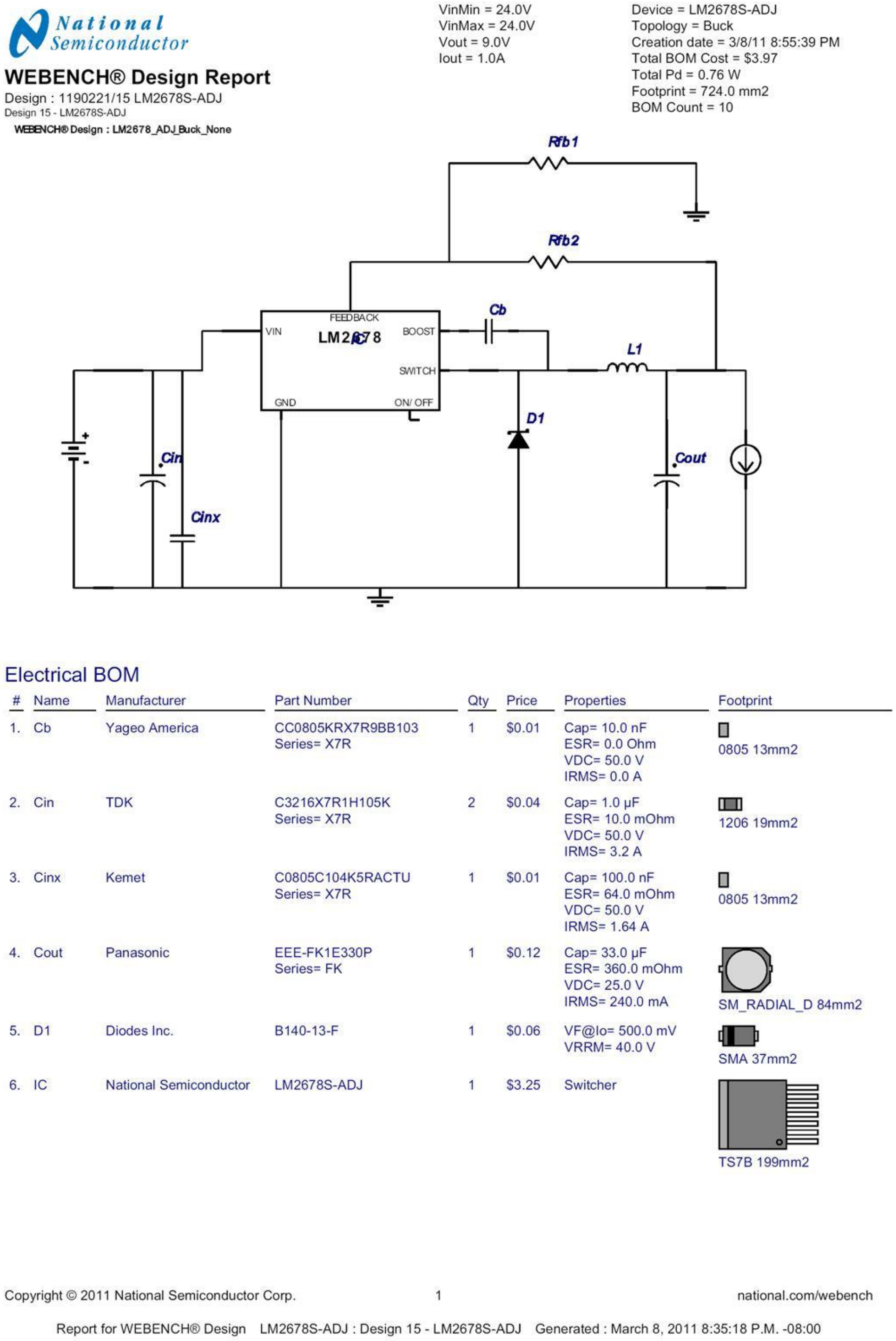
Test Name:	[DS-IT-06] – The GenioBox loses power
Requirements:	N/A
Preconditions:	The system is running under normal operation
Procedure:	1. The *Genio*Box loses power
Post Conditions:	The *Piccolo*Probe will lose power as well. The *Unisca*Suite will display a warning indicating that there is no connection to the *Genio*Box.

Test Name:	[DS-IT-07] – The PiccoloProbe is placed in a fluid at or above 46 degrees Celsius.
Requirements:	N/A
Preconditions:	The VivaceFlow system is running but the Probe has not been inserted into the liquid media yet
Procedure:	1. The *Piccolo*Probe is placed into a liquid with a temperature above 46 degrees Celsius
Post Conditions:	The system will assume the *Piccolo*Probe is at risk of causing bodily harm and will stop providing power to the *Piccolo*Probe. The *Unisca*Suite will display a warning to the user to alert them of their mistake.

9 Conclusion

Our fantastic team at VeloStream Technologies is dedicated to the development our *Vivace*Flow Blood Flow Speed Measurement Probe. This document has provided the technical details for the design of the individual components, integration, calibration, and system test plan. Due to our strong work ethic, we will be able to accomplish this project by mid-April.

Appendix A – WEBENCH Design Report for Power Supply



National Semiconductor

WEBENCH® Design Report

Design : 1190221/15 LM2678S-ADJ
Design 15 - LM2678S-ADJ
WEBENCH® Design : LM2678_ADJ_Buck_None

VinMin = 24.0V
VinMax = 24.0V
Vout = 9.0V
Iout = 1.0A

Device = LM2678S-ADJ
Topology = Buck
Creation date = 3/8/11 8:55:39 PM
Total BOM Cost = $3.97
Total Pd = 0.76 W
Footprint = 724.0 mm2
BOM Count = 10

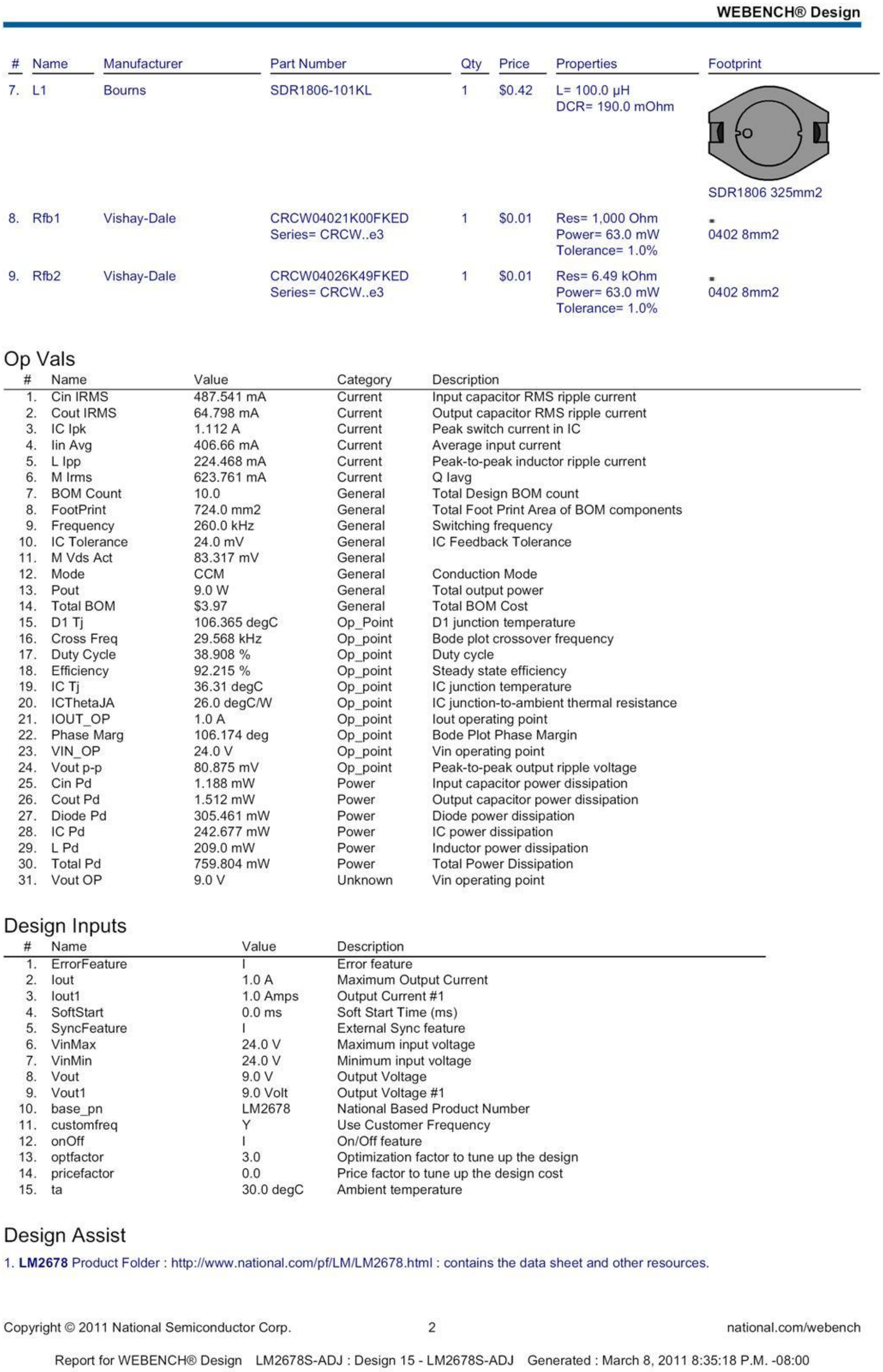
Electrical BOM

#	Name	Manufacturer	Part Number	Qty	Price	Properties	Footprint
1.	Cb	Yageo America	CC0805KRX7R9BB103 Series= X7R	1	$0.01	Cap= 10.0 nF ESR= 0.0 Ohm VDC= 50.0 V IRMS= 0.0 A	0805 13mm2
2.	Cin	TDK	C3216X7R1H105K Series= X7R	2	$0.04	Cap= 1.0 µF ESR= 10.0 mOhm VDC= 50.0 V IRMS= 3.2 A	1206 19mm2
3.	Cinx	Kemet	C0805C104K5RACTU Series= X7R	1	$0.01	Cap= 100.0 nF ESR= 64.0 mOhm VDC= 50.0 V IRMS= 1.64 A	0805 13mm2
4.	Cout	Panasonic	EEE-FK1E330P Series= FK	1	$0.12	Cap= 33.0 µF ESR= 360.0 mOhm VDC= 25.0 V IRMS= 240.0 mA	SM_RADIAL_D 84mm2
5.	D1	Diodes Inc.	B140-13-F	1	$0.06	VF@Io= 500.0 mV VRRM= 40.0 V	SMA 37mm2
6.	IC	National Semiconductor	LM2678S-ADJ	1	$3.25	Switcher	TS7B 199mm2

1

national.com/webench

Report for WEBENCH® Design LM2678S-ADJ : Design 15 - LM2678S-ADJ Generated : March 8, 2011 8:35:18 P.M. -08:00



WEBENCH® Design

#	Name	Manufacturer	Part Number	Qty	Price	Properties	Footprint
7.	L1	Bourns	SDR1806-101KL	1	$0.42	L= 100.0 µH DCR= 190.0 mOhm	SDR1806 325mm2
8.	Rfb1	Vishay-Dale	CRCW04021K00FKED Series= CRCW..e3	1	$0.01	Res= 1,000 Ohm Power= 63.0 mW Tolerance= 1.0%	0402 8mm2
9.	Rfb2	Vishay-Dale	CRCW04026K49FKED Series= CRCW..e3	1	$0.01	Res= 6.49 kOhm Power= 63.0 mW Tolerance= 1.0%	0402 8mm2

Op Vals

#	Name	Value	Category	Description
1.	Cin IRMS	487.541 mA	Current	Input capacitor RMS ripple current
2.	Cout IRMS	64.798 mA	Current	Output capacitor RMS ripple current
3.	IC Ipk	1.112 A	Current	Peak switch current in IC
4.	Iin Avg	406.66 mA	Current	Average input current
5.	L Ipp	224.468 mA	Current	Peak-to-peak inductor ripple current
6.	M Irms	623.761 mA	Current	Q Iavg
7.	BOM Count	10.0	General	Total Design BOM count
8.	FootPrint	724.0 mm2	General	Total Foot Print Area of BOM components
9.	Frequency	260.0 kHz	General	Switching frequency
10.	IC Tolerance	24.0 mV	General	IC Feedback Tolerance
11.	M Vds Act	83.317 mV	General	
12.	Mode	CCM	General	Conduction Mode
13.	Pout	9.0 W	General	Total output power
14.	Total BOM	$3.97	General	Total BOM Cost
15.	D1 Tj	106.365 degC	Op_Point	D1 junction temperature
16.	Cross Freq	29.568 kHz	Op_point	Bode plot crossover frequency
17.	Duty Cycle	38.908 %	Op_point	Duty cycle
18.	Efficiency	92.215 %	Op_point	Steady state efficiency
19.	IC Tj	36.31 degC	Op_point	IC junction temperature
20.	ICThetaJA	26.0 degC/W	Op_point	IC junction-to-ambient thermal resistance
21.	IOUT_OP	1.0 A	Op_point	Iout operating point
22.	Phase Marg	106.174 deg	Op_point	Bode Plot Phase Margin
23.	VIN_OP	24.0 V	Op_point	Vin operating point
24.	Vout p-p	80.875 mV	Op_point	Peak-to-peak output ripple voltage
25.	Cin Pd	1.188 mW	Power	Input capacitor power dissipation
26.	Cout Pd	1.512 mW	Power	Output capacitor power dissipation
27.	Diode Pd	305.461 mW	Power	Diode power dissipation
28.	IC Pd	242.677 mW	Power	IC power dissipation
29.	L Pd	209.0 mW	Power	Inductor power dissipation
30.	Total Pd	759.804 mW	Power	Total Power Dissipation
31.	Vout OP	9.0 V	Unknown	Vin operating point

Design Inputs

#	Name	Value	Description
1.	ErrorFeature	I	Error feature
2.	Iout	1.0 A	Maximum Output Current
3.	Iout1	1.0 Amps	Output Current #1
4.	SoftStart	0.0 ms	Soft Start Time (ms)
5.	SyncFeature	I	External Sync feature
6.	VinMax	24.0 V	Maximum input voltage
7.	VinMin	24.0 V	Minimum input voltage
8.	Vout	9.0 V	Output Voltage
9.	Vout1	9.0 Volt	Output Voltage #1
10.	base_pn	LM2678	National Based Product Number
11.	customfreq	Y	Use Customer Frequency
12.	onOff	I	On/Off feature
13.	optfactor	3.0	Optimization factor to tune up the design
14.	pricefactor	0.0	Price factor to tune up the design cost
15.	ta	30.0 degC	Ambient temperature

Design Assist

1. **LM2678** Product Folder : http://www.national.com/pf/LM/LM2678.html : contains the data sheet and other resources.

2 national.com/webench

Report for WEBENCH® Design LM2678S-ADJ : Design 15 - LM2678S-ADJ Generated : March 8, 2011 8:35:18 P.M. -08:00

References

AHA. (2011). *Cardiovascular Disease Statistics*. Retrieved January 25, 2011, from American Heart Association: http://www.americanheart.org

Charles Riva, B. R. (1972). Laser Doppler measurements of blood flow in capillary tubes and retinal arteries. *Investigative Opthalmology and Visual Science* , 936-944.

Delaunois, A. L. (1973). Thermal method for continuous blood-velocity measurements in large blood vessels, and cardiac-output determinaion. *Medical and Biological Engineering* , 201-205.

Dorsey, N. (1940). *Properties of Ordinary Water-Substance.* New York: Hafner Publishing.

Holland, F. A., & Bragg, R. (1995). *Fluid Flow for Chemical Engineers* (Vol. 2). Oxford: Elsevier.

Inc., F. (n.d.). Retrieved January 25, 2011, from Medical Dictionary: http://medical-dictionary.thefreedictionary.com/

Kim, Y.-H., Marom, E. M., Herndon, J. E., & McAdams, H. P. (2005). Pulmonary Vein Diameter, Cross-sectional Area, and Shape: CT Analysis. *Radiology* (235), 43-50.

Mace. (2008). *Doppler Technologies*. Retrieved January 25, 2011, from Mace Meters: http://macemeters.com/products/technologies-doppler/

Medicine, U. N. (2010, December 15). Retrieved January 25, 2011, from MedlinePlus Medical Encyclopedia: http://www.nlm.nih.gov/medlineplus/ency/article/000184.htm

MedicineNet, I. (1988, October 28). *Coronary arteries definition*. Retrieved January 25, 2011, from Medical Terms: http://www.medterms.com/script/main/art.asp?articlekey=7250

Medtronic. (2010, September 22). *Bio Probe Flow Transducer*. Retrieved January 25, 2011, from Medtronic: http://www.medtronic.com/

NHLBI. (2009, May). *Diseases and Conditions Index*. Retrieved January 25, 2011, from National Heart Blood and Lunk Institute: http://www.nhlbi.nih.gov

Pim A.L. Tonino, M. B. (2009). Fractional Flow Reserve versus Angiography. *The New England Journal of Medicine , 360* (3), 213-224.

Shercliff, J. A. (1962). *The theory of electromagnetic flow-measurement.* Cambridge: Cambridge University Press.

Steiner, E., Drewbrook, C., Naziripour, K., Gosling, W., & Sahachaiwatana, J. (2011). *Functional Specifications for the VivaceFlow Speed Measurement Probe.* Burnaby: Simon Fraser University.

University, P. (2010, December 20). Retrieved January 25, 2011, from Wordnet: http://wordnet.princeton.edu/