



QUANSER
INNOVATE. EDUCATE.

Heat Flow Experiment

System identification & Frequency domain design

1 Objective

Perform system identification on the heatflow experiment and design a controller in the frequency domain to control the air temperature at selected points in a duct using a heating element and a blower.

This heat-flow experiment (HFE) system consist of a duct equipped with a heater and a blower at one end and three temperature sensors located along the duct. The power delivered to the heater is controlled using an analog signal. The fan speed can also be controlled using an analog signal. Fast settling platinum temperature transducers are used to measure the temperature. Fan speed is measured using a tachometer and can be used to design speed controllers.

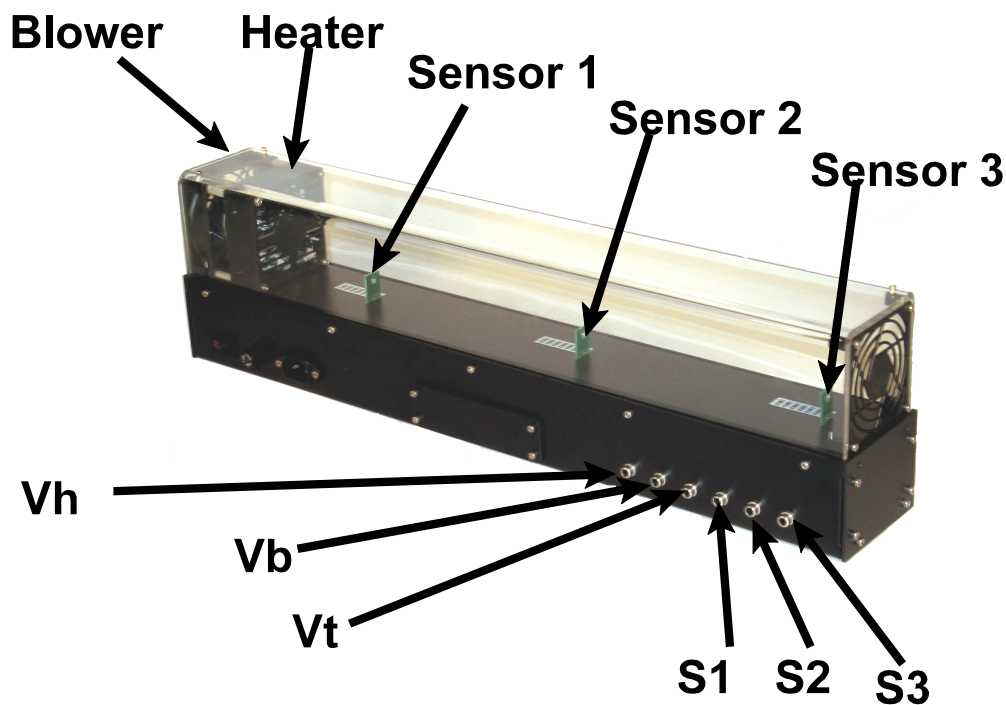


Figure 1 Heatflow apparatus

2 Open loop model

A complete thermodynamic model of the system is difficult to derive. In principle however the following equation applies:

$$\dot{T}_n = F(V_h, V_b, T_a, x_s)$$

Where

T _n	Temperature at Sensor n
V _b	Voltage applied to blower
V _h	Voltage applied to heater
T _a	Ambient temperature
X _n	Distance of sensor n from heater

This experiment is aimed at introducing you to the system in order to gain insight into the salient issues. You will develop a very simple first order model and an associated controller. In subsequent experiments you will investigate system identification techniques and control of systems with time delay.

3 System identification experiment

In the first experiment, we developed an open loop model by examining the step response of the system. In this experiment, we shall investigate the use of system identification techniques to develop a higher order model and use it to design feedback controller. Note that we have 5Hz low pass filters to remove some of the noise in the sensors.

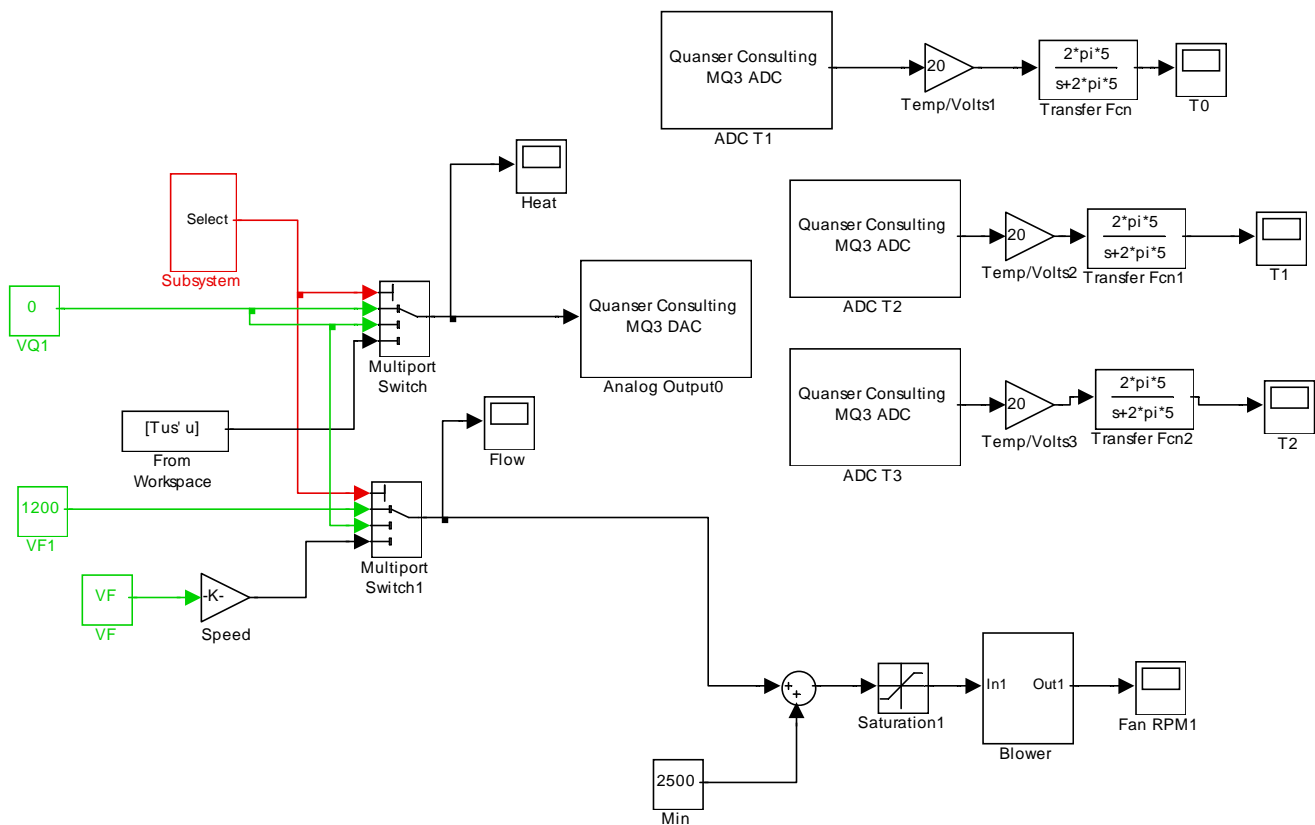


Figure 2 Simulink model to apply random inputs and collect data.

3.1 Data Collection

Follow these steps to collect the data :

- Start Matlab
- Change the directory where you have the FHE models
- In Matlab type **random_input** - This opens the simulink model shown in Figure 2
- In random_input.mdl Click WinCon / Set WinCon Options - This sets the compile for the OS you have.
- In random_input.mdl Click WinCon / Set WinCon Options - This sets the compile for the OS you have.
- In Matlab run **setup_random** - This initializes parameters so you can compile the diagram- It also sets up the random sequence to be applied to the heater

- In the diagram click WinCon/ Build - this will build the realtime code and download to WinCon
- In WinCon click on Plot / New / Scope and select T0, T1 T2
- In the WinCon plot, select Update/ Buffer and enter 60
- Click run - this will run the controller for 240 seconds. (4 minutes!)
- When the run is complete , you will see the last 60 seconds of data in the WinCon plot. Click File/ save and save as a mat file in the file named : **data_rand1.mat**
- Now that the data is saved you can load into Matlab

```
> load dat_rand1
```

Plot the data

```
> plot( plot_time , [random_input_T1,random_input_Heat]);
```

This shows the plot in Figure 3

The blue trace is the temperature at sensor S1 and the green trace is the random input applied to the heater.

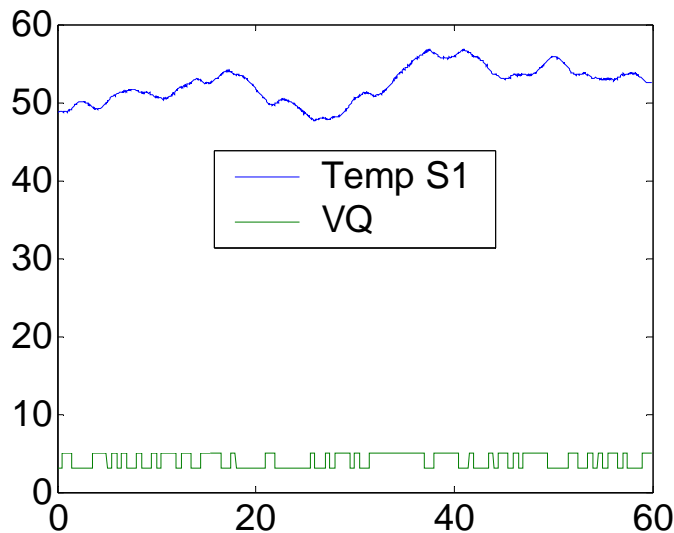


Figure 3 Response of T1 to random input VQ. Note that the data is collected after running the experiment for 4 minutes in order to remove trends.

3.2 Model estimation

Now that the data is saved, you can run the script `id_freq.m` - This script loads the data that was collected and uses pem (Prediction error estimate - See Matlab System Identification toolbox) to create a best fit linear model of the system.

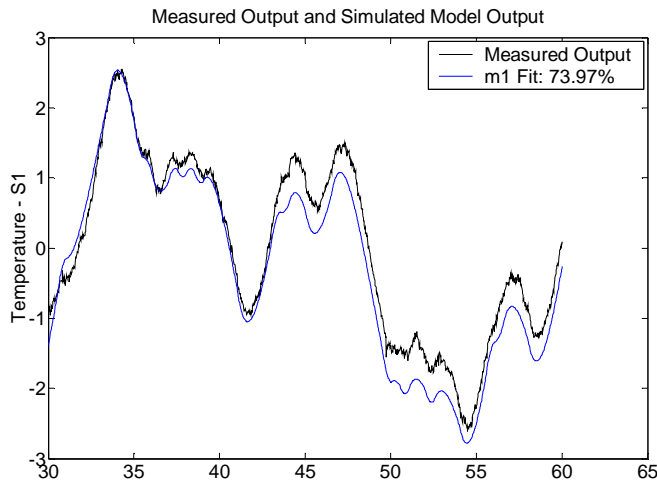
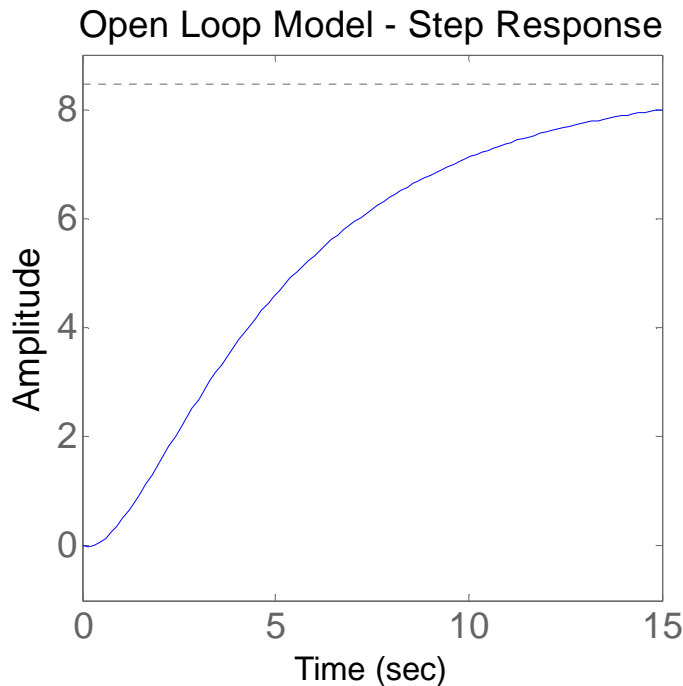


Figure 4 Simulated versus estimated response from random input data

Figure 4 shows the actual data versus data from the estimated model parameters.

Figure 5 shows the predicted step response of the system



The estimated open loop transfer function (for S1) is obtained as:

$$\frac{S1}{VQ} = \frac{-0.2405 s + 1.7207}{s^2 + 1.17 s + 0.2}$$

4 Control System Design

We design controller in the frequency domain. Clearly we require an integrator in the loop in order to obtain zero steady state error.

The controller we design is of the form:

$$VQ = \frac{1}{s} K a \frac{s + \frac{wm}{a}}{s + wm a}$$

which consists of a an integrator followed by a lead centred at wm and a gain K .

The design file `d_freq` uses the estimated parameters and prompts the user for gain and phase lead values to design an appropriate compensator. (The values in parentheses are the recommended values to enter)

Using `d_freq` we obtain the following compensator:

$$H(s) = \frac{1.18 s + 0.316}{s + 3.7321}$$

5 Setting up

Wire up the system as described below. Use single RCA cables for each connection.

Inputs to system	Attach to Analog outputs	Range
Vh - Voltage to heater	MultiQ D/A # 0 - Control heat delivered	0 - 5 VDC
Vb - Voltage to blower	MultiQ D/A # 1 - Control fan speed (open loop)	0 - 5 VDC
Measurements	Attach to Analog inputs	
S1	MultiQ A/D #0	0 - 5 VDC
S2	MultiQ A/D #1	0 - 5 VDC
S3	MultiQ A/D #2	0 - 5 VDC
Vt	MultiQ A/D #3	0 - 5 VDC

6 Implementation and results

The controller is implemented using WinCon and Simulink. The Simulink controller is in the file `q_hfe_freq.mdl` The controller measures the voltages from the three sensors and calibrates them to degrees. The feedback controller implements the equation.

$$VQ = \frac{1}{s} \frac{0.6637 s + 0.1778}{s + 3.7321} + V_{ff}$$

Note that we have also added a feedforward component V_{ff} which pre-computes the steady state voltage required for the desired heat. The value of V_{ff} is obtained as follows:

Consider the estimated open loop transfer function:

$$\frac{S1}{VQ} = \frac{-0.2405 s + 1.7207}{s^2 + 1.17 s + 0.2}$$

The steady state gain is $G_{ss} = 1.7/0.2 = 8.5$ deg/volt. That means that in the steady state, in order to maintain 8.5 deg

change from the nominal value we need 1 volt applied to the heater. We can use this gain to predict the steady state value required to maintain the desired temperature.

The feedforward voltage to the heater then is :

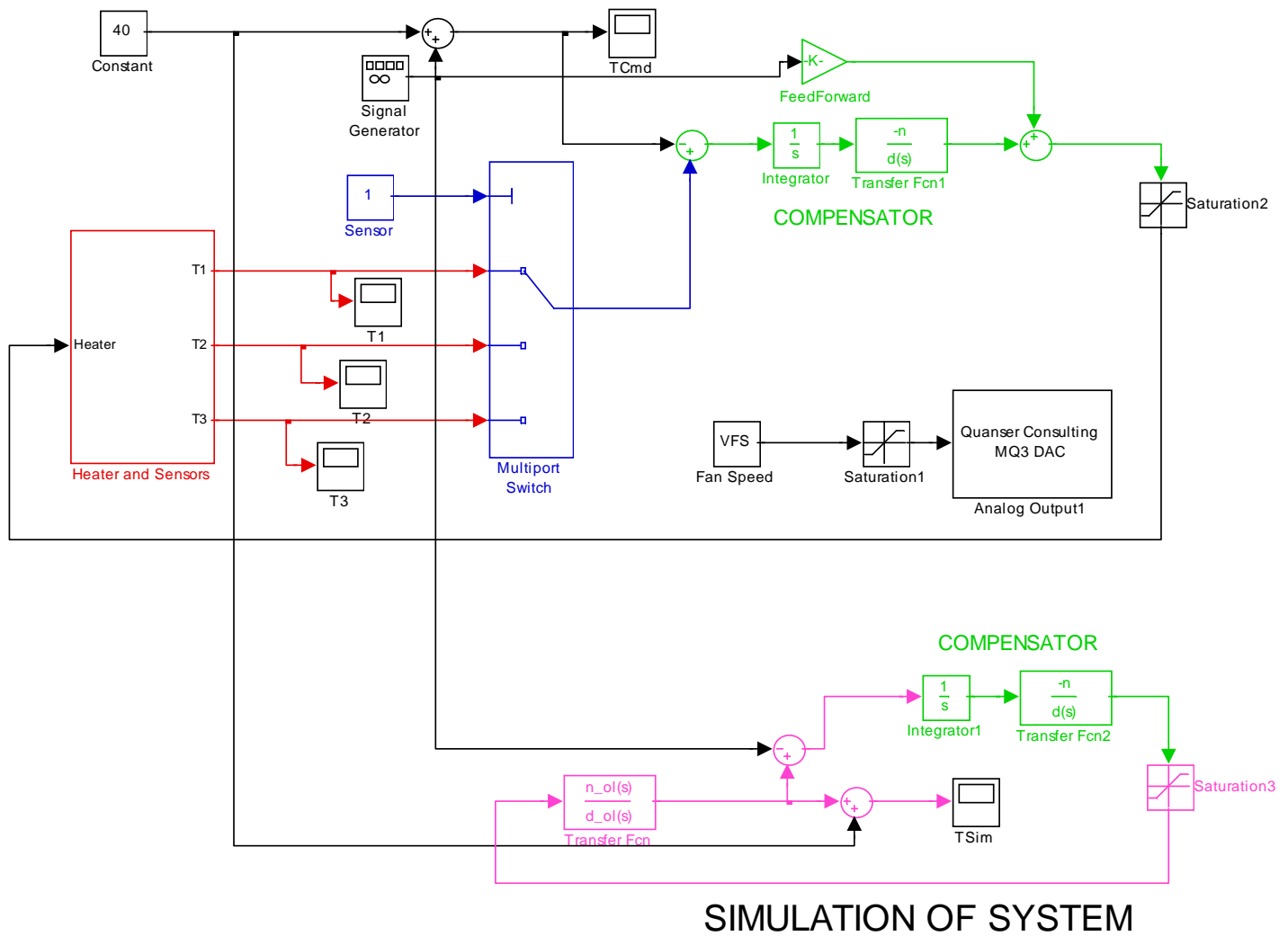
$$V_{ff} = \frac{1}{G_{ss}}(V_{desired} - V_{nominal})$$

Depending on the value of the selector (Constant named Sensor in the diagram), we will control one of the three points along the duct. Note that we are also running a realtime simulation of the system modeled as obtained from the PEM estimator.

By setting Sensor = 1, we control T1 and make it track a square wave of 5 deg p-p centered about 40 degrees. The response is shown in Figure 7.

Examining the response we note the following:

- 1 The output tracks the desired temperature command with zero steady state error
- 2 The transient response matches the simulation closely



SIMULATION OF SYSTEM

Figure 6 Feedback controller with frequency domain designed compensator. Note a simulation of the system is running in tandem.

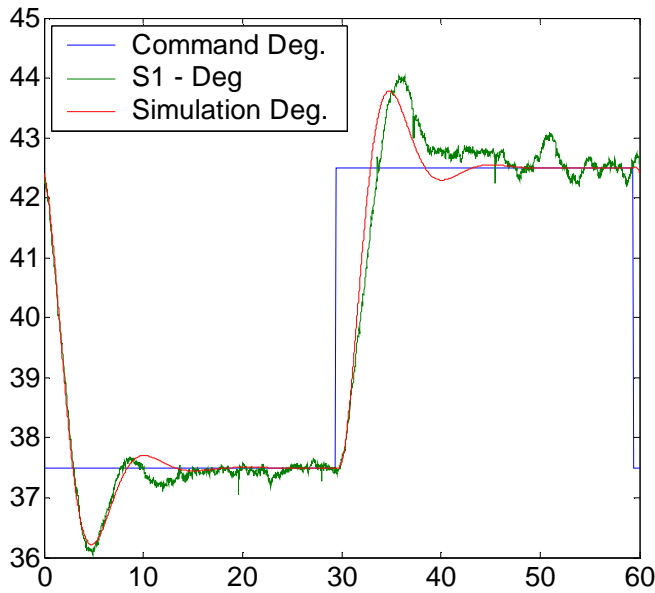


Figure 7 Response at S1 using compensator designed with estimated parameters- Green trace is actual system response while red trace is the simulated response using estimated model.

We can easily examine the system response if we try to control the temperature at S3. We do this by setting Sensor = 3 in the diagram and running it again. In this case the response is shown in Figure 8. We note that the performance deteriorates, primarily due to the transport delay in the system.

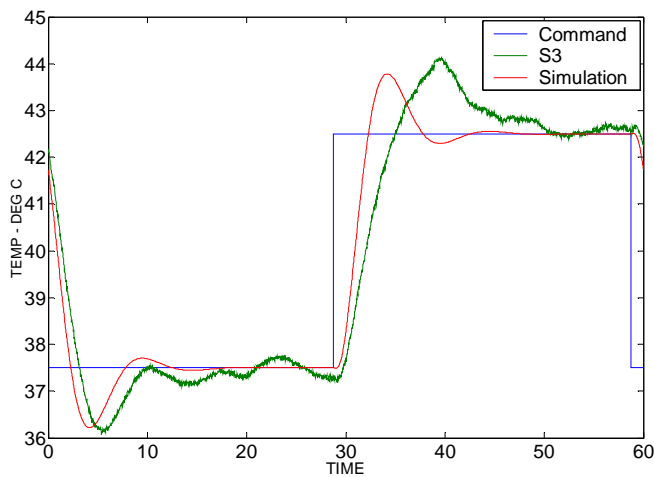


Figure 8 Response at S3 when using controller designed for S1

3 Conclusions

The system identification technique results in a good model that can be used to design a frequency domain compensator. The actual system response closely matches the predicted response from the estimated model. Closing the loop at S3 using the controller designed for S1, although inferior, still has acceptable performance.

4 System Parameters

Parameter	Symbol	Value	Units
Heater maximum power (5 V Input)	P	400	Watt
Blower nominal airflow (4 V Input)	B	32.1	CFM
Blower nominal airflow (4 V Input)	B _{SI}	0.91	m ³ / min
Maximum blower speed (5 V Input)	Fmax	2580	RPM
Minimum blower speed (0 V Input)	Fmin	1580	RPM
Tachometer calibration	Ct	1063	RPM / V
Temperature Sensor calibration	Cs	20	Deg C / V
Duct cross section area	A	0.0064	m ²
Max wind speed	Ws	142	m/ min
Temp sensor settling time at max wind speed	Ts	4	seconds
Dimensions		Power Requirements	
50 x 15 x 10	cm	VAC	100 - 240
0.5	Kg	Amps Max	5
Data Acquisition Requirements			
Analog inputs	4	0 - 5 V DC	
Analog outputs	2	0 - 5 V DC	

5 Calibration

Occasionally you may need to calibrate the sensor measurements. You will need a measurement of ambient temperature and a DC Voltmeter. (Or you can use WinCon)

Power up the system

Open the service panel as shown

Measure S1 and adjust potentiometer P1 such that the voltage is proportional to room temperature

Measure S2 and adjust potentiometer P2 such that the voltage is proportional to room temperature

Measure S3 and adjust potentiometer P3 such that the voltage is proportional to room temperature

The table below shows the voltages you should obtain at various temperatures:

Temp	19	20	21	22	23	24	25	26	27	28	29	30
Vs	0.95	1.00	1.05	1.10	1.15	1.20	1.25	1.30	1.35	1.40	1.45	1.50