Cloud Publications
International Journal of Advanced Packaging Technology
2017, Volume 5, Issue 1, pp. 258-266
ISSN 2349-6665, doi 10.23953/cloud.ijapt.28

Case Study

Open Access

Low Cost Temperature & Humidity Chamber

Erica Thomas, Siripong Malasri, Robert Moats, and Dustin Schrecongost

Healthcare Packaging Consortium, Christian Brothers University, 650 East Parkway South, Memphis, TN, USA

Correspondence should be addressed to Siripong Malasri, pong@cbu.edu

Publication Date: 24 January 2017

DOI: https://doi.org/10.23953/cloud.ijapt.28

Abstract A low-cost 56 ft^3 temperature/humidity chamber was built from 2-inch thick extruded polystyrene (XPS) foam. Commercially available instruments were used, including space heater, humidifier, and temperature/humidity data logger. Due to the inaccurate control of home-use heater and humidifier, environmental condition in the chamber was different from the setting condition. Thus, thirty different combinations of temperature ranging from 70-90$^\circ F$ and relative humidity ranging from 50-75% were used with actual conditions recorded by the data logger. A calibration spreadsheet was then developed using an artificial neural network to instruct the user to set environmental conditions for desired conditions. The neural network spreadsheet predicted temperature and relative humidity within 3% and 2% errors, respectively.

Keywords *Temperature/Humidity Chamber; Environmental Chamber; Artificial Neural Network*

1. Introduction

The CBU packaging test laboratory has had a commercial environmental chamber since 2005 with a controller upgrade in 2016. Since the laboratory has become an ISTA certified packaging test lab in 2009, the chamber has been regularly used for various commercial testing projects. There was a need for the second chamber for R&D projects that do not require the sophistication of an expensive commercial chamber. Thus, a low-cost environmental chamber as described in this article was built from commercially available materials and instruments for under 1,000USD. Table 1 shows a comparison of features between this low-cost chamber and the exiting commercial chamber.

Table 1: *Low-Cost Chamber versus Commercial Chamber*

Feature	Low-Cost Chamber	Commercial Chamber
Cost	Under 1,000USD	Over 30,000USD
Chamber Volume (Interior Dimensions)	56 ft^3 (44"X48"X46")	32 ft^3 (38"X38"X38")
Temperature Range	70 to 90$^\circ F$	-49 to 374$^\circ F$

Relative Humidity Range	50 to 75%	10 to 98%
Profile Setting	One setting	Multiple settings
Temperature Setting Increment	$1^{\circ}F$	$0.1^{\circ}F$
Relative Humidity Setting Increment	5%	0.1%
Footprint	48”X52”	48”X72”
Height	50”	92”
Portability	Yes	No

2. Materials and Methods

Figure 1 shows instruments used in building the low-cost chamber.

- Heater [1]: Automatic temperature control with 750/1500 watts (120USD)
- Humidifier [2]: Ultrasonic warm and cool mist humidifier with auto humidistat and timer (124USD)
- Temperature/Humidity Data Logger [3]: 16,000 humidity and 16,000 temperature readings with a user programmable sample rate and analysis software. Temperate range: -40 to 158 $^{\circ}F$. Relative humidity range: 0 to 100% (128USD)
- Small Fan: For air circulation inside the chamber (12USD)
- Tablet: For calibration spreadsheet (50USD)

Figure 1: *Commercially Available Instruments*

The body of the chamber was built from a 2-inch thick Owens Corning R-10, Foamular 250, energy-saving moisture-resistant extruded polystyrene (XPS) foam [4] as shown in Figure 2. Door was attached to the chamber body via industry-grade Velcro. A 7-inch Android tablet for calibration spreadsheet was mounted next to the chamber. Figure 2 also shows the layout of instruments, with the data logger attached to the right interior side wall.

Figure 2: *Chamber Details*

Two methods were used to ensure that no moisture was absorbed by the XPS foam. In the first method, several XPS specimens were placed in the commercial chamber (at $73^{\circ}F$ and 90% relative humidity (RH)), as shown in the left image of Figure 3. They were weighed daily with 0.0001 gram accuracy for five consecutive days and no weight change was observed. In the second method, a tube filled with water was secured above a piece of XPS. The bottom of the tube was sealed to prevent leakage, while the top was covered to prevent evaporation, as shown in the right image of Figure 3. The water level was observed daily for five consecutive days. No change was observed. Thus, the XPS moisture-resistance was validated.

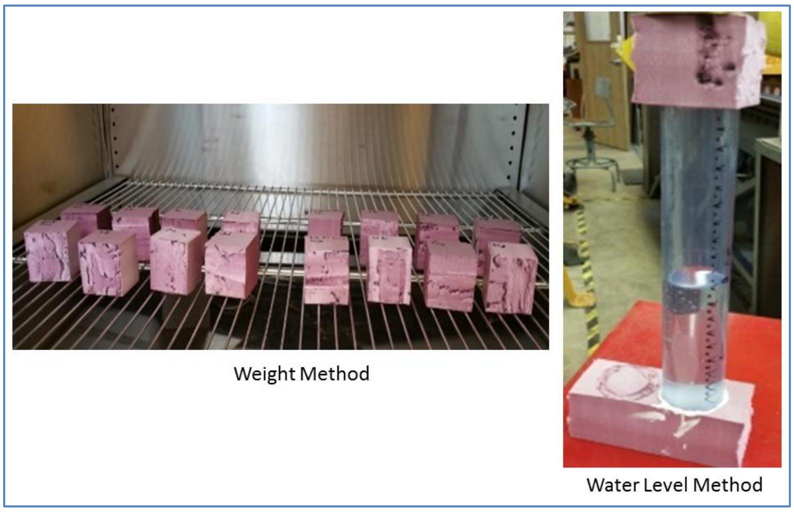

Figure 3: *XPS Moisture-Resistant Verification*

Chamber calibration consisted of collecting 16 to 18 hours of data every one minute for 30 different temperature-humidity combinations ranging from 70-90$^{\circ}F$ and 50-75% RH, in 5$^{\circ}F$ and 5% RH increments, respectively. Tests were run for four days per week over the course of three months. Data was downloaded from the data logger after each test. Ten to 14 hours of the 16 to 18 hours of data collected were used to generate measurements of average chamber operating temperature and relative humidity values given specified heater/ humidifier settings. Only 10 to 14 hours of the data were used because four to six hours were needed for the chamber to reach a steady condition.

Data for each temperature-RH combination was downloaded and averaged, with the maximum, minimum, and range recorded for each testing day. After data for all 30 temperature-RH combinations were collected; scatter plots, trend lines, and trend line equations were produced to establish general chamber temperature and RH calibration charts (see Figure 4 and Figure 5).

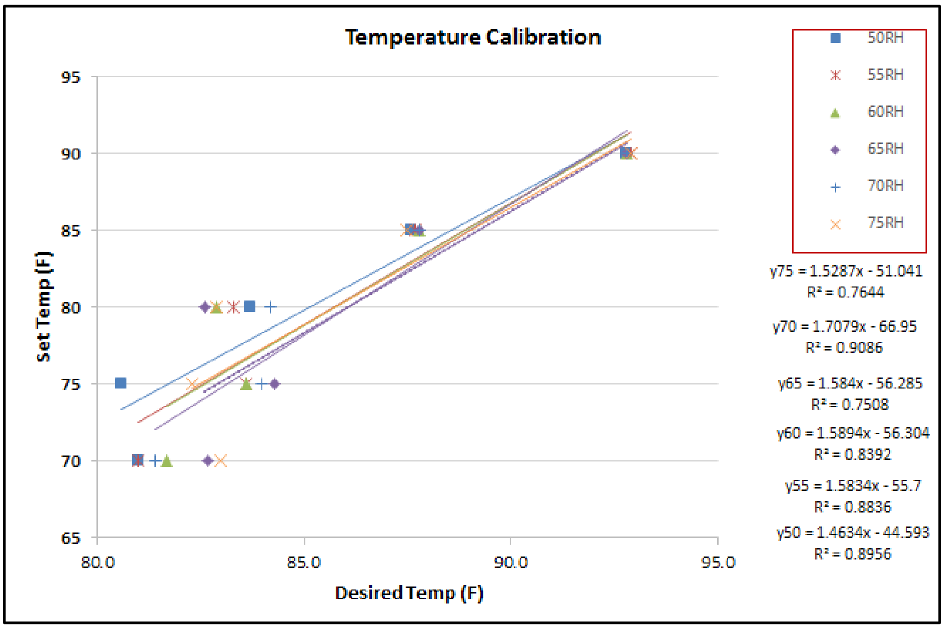

Figure 4: *Temperature Calibration Chart*

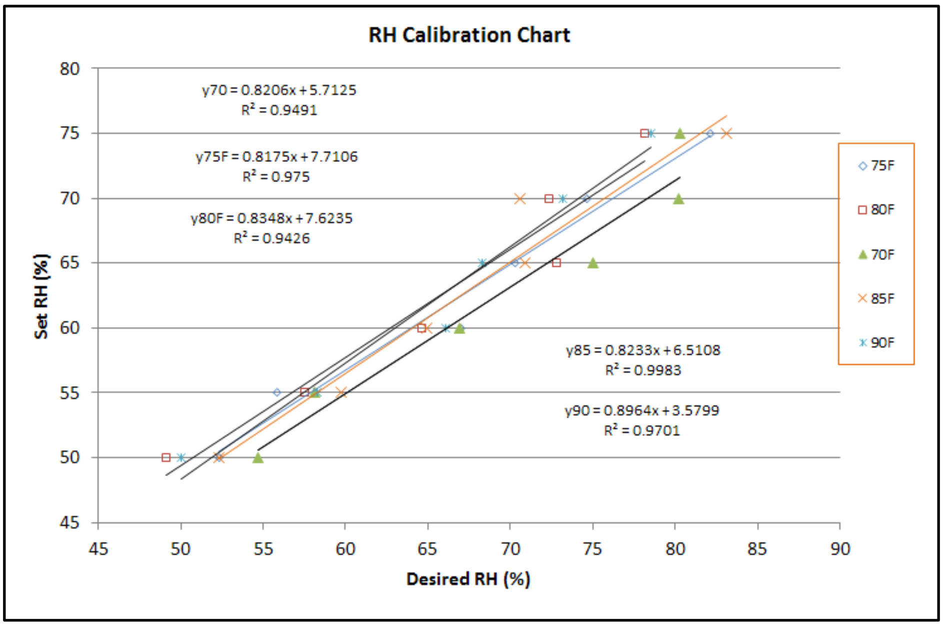

Figure 5: *RH Calibration Chart*

Training data set was then generated from these trend line equations for use with NeuroShell2 neural network software [5]. A feed-forward fully-connected backpropagation neural network shown in Figure 6 was used. The numbers of input and output neurons were controlled by the collected data, i.e., two input parameters (desired temperature and desired RH) and two output parameters (set temperature and set RH). The number of hidden neurons was arbitrary and was chosen as seven in this work.

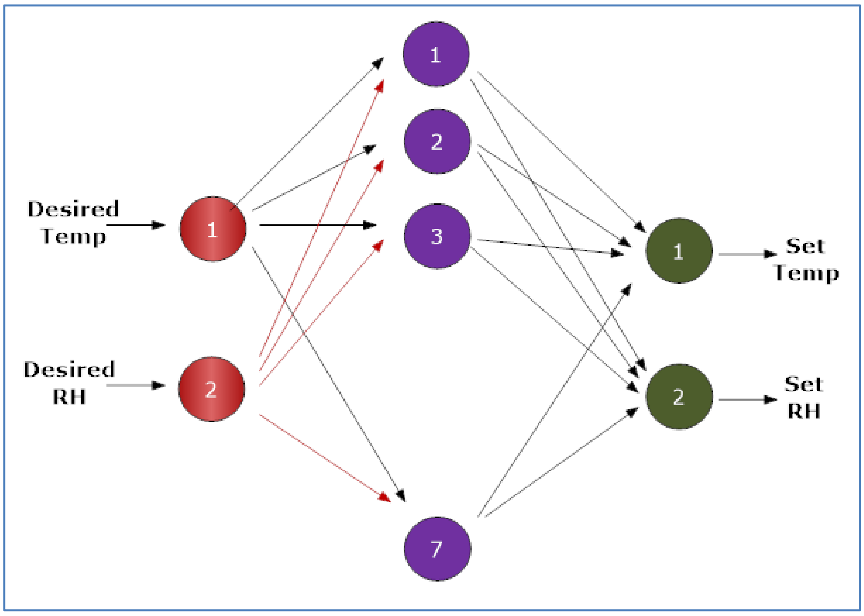

Figure 6: *Calibration Neural Network*

3. Results and Discussion

Table 2 summarizes training data, neural network predicted value, and % errors.

- Column 1 **Case**: Thirty cases were used in developing the neural network calibration software.
- Column 2 **Desired Temp**: This is the desired temperature in the chamber.
- Column 3 **Desired RH**: This is the desired RH in the chamber
- Column 4 **Set Temp**: This is the set temperature on the heater. It should be noted that this is not the raw data collected. It is the value generated from a trend line equation shown in Figure 4.
- Column 5 **Set RH**: This is the set RH on the humidifier. It should be noted that this is not the raw data collected. It is the value generated from a trend line equation shown in Figure 5.
- Column 6 **Mark**: The code 'T' indicates that the data is for training.
- Columns 7 & 8 **NN Set Temp** & **NN Set RH**: These are the set temperature and set RH predicted by the neural network.
- Column 9 & 10 Temp Error & **RH Error**: These are the errors for temperature and RH predictions.

The neural network predicts temperature with an error range of 0 – 2.70% and average error of 0.75%. It predicts relative humidity with an error range of 0.05 – 1.26% and average error of 0.49%.

***Table 2:** Performance of Neural Network Calibration Software*

1	2	3	4	5	6	7	8	9	10
Case	**Desired Temp ($^\circ F$)**	**Desired RH (%)**	**Set Temp ($^\circ F$)**	**Set RH (%)**	**Mark**	**NN Set Temp ($^\circ F$)**	**NN Set RH (%)**	**Temp Error (%)**	**RH Error (%)**
1	70	50	58	47	'T'	57	48	2.51	1.26
2	70	55	55	51	'T'	56	51	1.66	0.47
3	70	60	55	55	'T'	55	55	0.52	0.16
4	70	65	55	59	'T'	55	59	0.66	0.72
5	70	70	53	63	'T'	54	63	2.70	0.54
6	70	75	56	67	'T'	55	68	1.88	0.75
7	75	50	65	49	'T'	65	49	0.62	0.67
8	75	55	63	53	'T'	64	53	1.33	0.68
9	75	60	63	57	'T'	63	57	0.10	0.55
10	75	65	63	61	'T'	62	61	1.26	0.05
11	75	70	61	65	'T'	62	65	1.88	0.17
12	75	75	64	69	'T'	63	69	0.94	0.05
13	80	50	72	49	'T'	72	49	0.51	0.32
14	80	55	71	54	'T'	71	54	0.70	0.85
15	80	60	71	58	'T'	71	58	0.51	0.70
16	80	65	70	62	'T'	70	62	0.35	0.13
17	80	70	70	66	'T'	70	66	0.16	0.28
18	80	75	71	70	'T'	72	70	0.78	0.51
19	85	50	80	48	'T'	80	48	0.13	0.39
20	85	55	79	52	'T'	79	52	0.53	0.08
21	85	60	79	56	'T'	79	56	0.44	0.63
22	85	65	78	60	'T'	78	60	0.23	0.19
23	85	70	78	64	'T'	78	64	0.11	0.20
24	85	75	79	68	'T'	79	68	0.39	0.49
25	90	50	87	48	'T'	87	48	0.01	0.27
26	90	55	87	53	'T'	87	52	0.11	1.16
27	90	60	87	57	'T'	87	58	0.30	0.97
28	90	65	86	62	'T'	86	62	0.56	0.21
29	90	70	87	66	'T'	87	66	0.46	0.54
30	90	75	87	71	'T'	87	70	0.00	0.83
							Min =	*0.00*	*0.05*
							Max =	*2.70*	*1.26*
							Avg =	*0.75*	*0.49*

The algorithm of the calibration software was generated from NeuroShell2 software (Appendix A) and programmed into an Excel spreadsheet as shown in Figure 7. In order to test the generalizability of the neural network algorithm the spreadsheet was used to generate data for an RH of 52.5%, which was not used in training. Figure 8 shows a temperature calibration chart with RH of 52.5%, generated by the calibration spreadsheet, plotting between an RH of 50% and 55%. Similarly, the generalizability was shown on RH calibration chart using a temperature of 72.5 $^\circ F$ (Figure 9).

	A	B	C	D
7	netsum			
8	feature2(7)			
9				
10	Note - the following are names of inputs and outputs:			
11	Note - inp(1) is AT	Desired Temperature (F)	80	Input
12	Note - inp(2) is ARH	Desired RH (%)	55	Input
13	Note - outp(1) is ST	Set Temperature (F)	71	Output
14	Note - outp(2) is SRH	Set RH (%)	54	Output
15				
16	if (inp(1)<70) then inp(1) = 70			
17	if (inp(1)>90) then inp(1) = 90			
18	inp(1) = (inp(1) - 70) /20	inp(1)	0.5	
19				
20	if (inp(2)<50) then inp(2) = 50			

Figure 7: *Calibration Excel Spreadsheet*

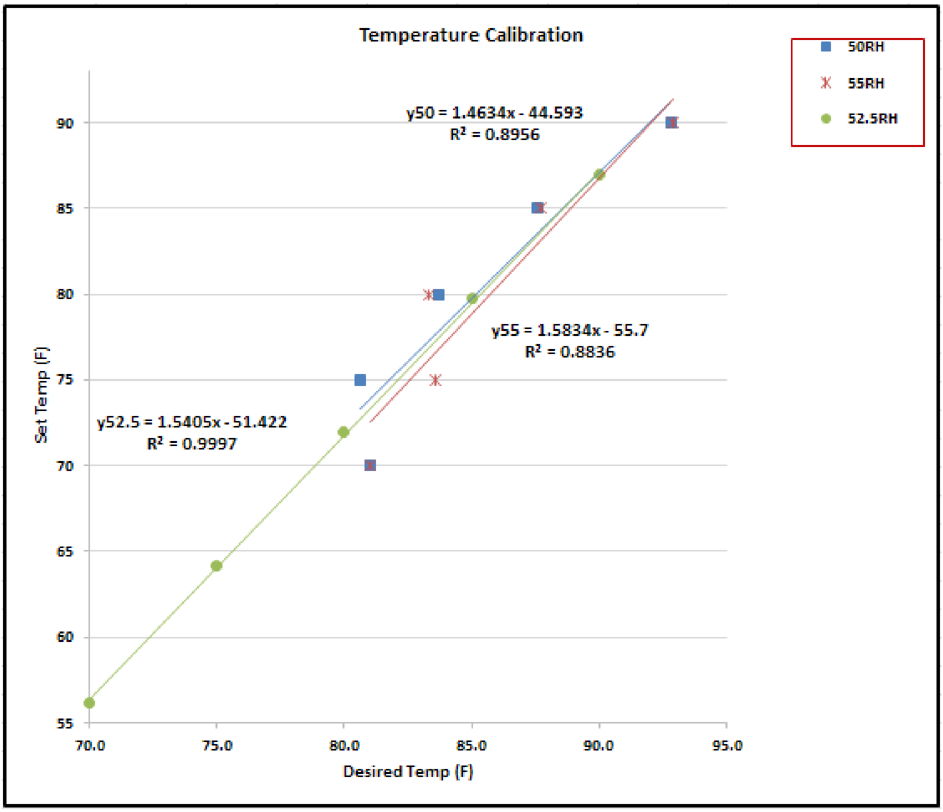

Figure 8: *Generalization of the Calibration Software Shown on Temperature Calibration Chart*

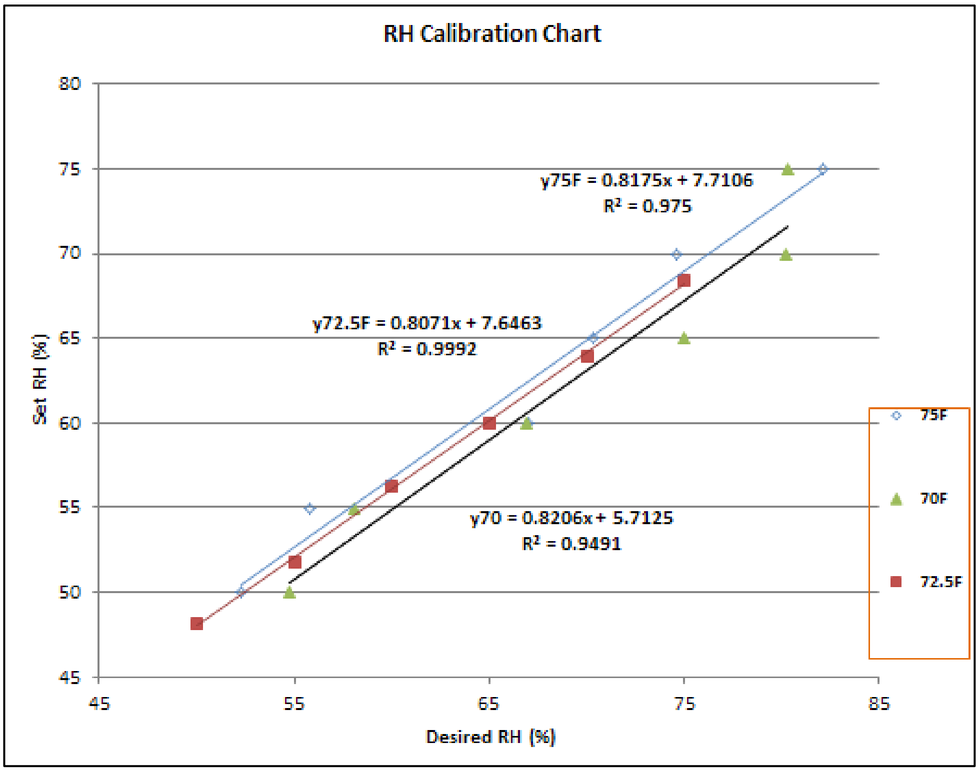

Figure 9: *Generalization of the Calibration Software Shown on Relative Humidity Calibration Chart*

4. Conclusion

This study demonstrates that the construction of a reliable and inexpensive temperature/humidity chamber using readily available materials is possible. Commercial chambers, depending on size, can have a starting price tag of 30,000USD. The chamber constructed for this study cost less than 1,000USD and can simulate temperature and relative humidity conditions that fall within the temperature-RH ranges tested with a percent error of less than 5%. Further testing of additional temperature-RH combinations would increase the reliability of setting temperatures and RH values for the heater and humidifier used in this study. However, it should be noted that temperature and humidity results do not necessarily transfer, even if a second chamber were to be built in exactly the same way, using the same components. All chambers require calibration.

Additionally, because residential grade heater and humidifier devices were used, temperature and humidity levels are constrained by the range and setting mode of the device. In this study, the heating range tested was from 70-90$^\circ F$ using only 5$^\circ F$ increments. The heater could have run using 1$^\circ F$ increments, and as a result, can provide additional parameters for testing. The humidifier chosen for this study was tested using a relative humidity range of 50-75% RH. Unlike the heater, the operational increments for the humidifier were limited to 5% increments. Consequently, if an experiment were to require a relative humidity setting between the 5% increments, the value would require rounding, which could influence the outcome of chamber interior condition.

Given the functional limitations of residential grade heating and humidifying devices, it is important to consider the level of accuracy required. However, use of the artificial neural network proved invaluable in its ability to interpolate - providing setting temperature and RH values, for temperature and humidity parameters not previously tested. Thus, while the choice of heater or humidifier may limit setting values, the use of an artificial neural network can advance the range of settings. Finally, if the percentage of error presented in this study is within reason, constructing and calibrating a low-cost temperature/ humidity chamber might be a reliable alternative.

References

[1] *Vornado Heater Owner's Guide*. https://images-na.ssl-images-amazon.com/images/I/818PuYfw0PS.pdf. *Access date: December 28, 2016.*

[2] *PureGuardian Ultrasonic Humidifier Use & Care Instructions*. https://www.guardiantechnologies.com/uploads/H8000_manual.pdf. *Access date: December 28, 2016.*

[3] *EXTECH Humidity/Temperature Datalogger Product Datasheet*. http://www.extech.com/resources/RHT20_DS-en.pdf. *Access date: December 28, 2016.*

[4] *Owens Corning Foamular Energy-Saving, Moisture-Resistant XPS Insulation*. http://www.owenscorning.com/NetworkShare/EIS/10010242-FOAMULAR-XPS-101-Commercial-Brochure.pdf. *Access date: December 28, 2016.*

[5] *Ward Systems Group, Inc. NeuroShell2*. http:www.wardsystems.com/neuroshell2.asp. *Access date: December 25, 2015.*

Appendix A Generic Source Code Generated by NeuroShell2

```
netsum
 feature2(7)
Note - the following are names of inputs and
outputs:
Note - inp(1) is AT
Note - inp(2) is ARH
Note - outp(1) is ST
Note - outp(2) is SRH
if (inp(1)<70) then inp(1) = 70
if (inp(1)>90) then inp(1) = 90
inp(1) = (inp(1) - 70) /20
if (inp(2)<50) then inp(2) = 50
if (inp(2)>75) then inp(2) = 75
inp(2) = (inp(2) - 50) /25
netsum = -3.070898
netsum = netsum + inp(1) * 4.10027
netsum = netsum + inp(2) * -1.408727
feature2(1) = 1 / (1 + exp(-netsum))
netsum = 0.2558365
netsum = netsum + inp(1) * -0.4038733
netsum = netsum + inp(2) * -3.089265
feature2(2) = 1 / (1 + exp(-netsum))
netsum = -0.6637989
netsum = netsum + inp(1) * -3.811205
netsum = netsum + inp(2) * 0.6173124
feature2(3) = 1 / (1 + exp(-netsum))
netsum = -22.38877
netsum = netsum + inp(1) * 9.987292E-02
netsum = netsum + inp(2) * 26.78505
feature2(4) = 1 / (1 + exp(-netsum))
netsum = 12.07675
netsum = netsum + inp(1) * -14.85412
netsum = netsum + inp(2) * -0.6312295
feature2(5) = 1 / (1 + exp(-netsum))
netsum = -14.60969
netsum = netsum + inp(1) * 13.47738
netsum = netsum + inp(2) * 0.8668328
feature2(6) = 1 / (1 + exp(-netsum))
netsum = -2.918397
netsum = netsum + inp(1) * -4.432046
netsum = netsum + inp(2) * 0.7326113
feature2(7) = 1 / (1 + exp(-netsum))
netsum = 0.7137885
netsum = netsum + feature2(1) * 1.945597
netsum = netsum + feature2(2) * -0.3679988
netsum = netsum + feature2(3) * -4.390033
netsum = netsum + feature2(4) * 0.3530397
netsum = netsum + feature2(5) * -0.4985445
netsum = netsum + feature2(6) * 1.220905
netsum = netsum + feature2(7) * -1.870898
outp(1) = 1 / (1 + exp(-netsum))
netsum = 0.3879333
netsum = netsum + feature2(1) * -0.7902336
netsum = netsum + feature2(2) * -5.173065
netsum = netsum + feature2(3) * -1.050403
netsum = netsum + feature2(4) * 0.8617824
netsum = netsum + feature2(5) * 0.9966981
netsum = netsum + feature2(6) * 2.90973
netsum = netsum + feature2(7) * -1.400668
outp(2) = 1 / (1 + exp(-netsum))
outp(1) = 34 *  (outp(1) - .1) / .8  + 53
outp(2) = 24 *  (outp(2) - .1) / .8  + 47
```