Generalized Net Model of a Body Temperature Data Logger Embedded System

Lenko Erbakanov[1], Krassimir Atanassov[1,2], Sotir Sotirov[1]

[1]*Intelligent Systems Laboratory*
Prof. D-r Assen Zlatarov University
Burgas, Bulgaria
E-mails: erbakanov@btu.bg, krat@bas.bg, ssotirov@btu.bg

[2]*Department of Bioinformatics and Mathematical Modelling*
Institute of Biophysics and Biomedical Engineering
Bulgarian Academy of Sciences
Sofia, Bulgaria
E-mail: krat@bas.bg

Received: February 27, 2015 **Accepted: June 18, 2015**

Published: June 30, 2015

Abstract: *This paper represents a prototype embedded system, intended for measuring, gathering and transmitting the obtained data of the temperature of the skin and the ambient air. The system is designed to collect data in relatively short intervals, for fairly long periods of time. The skin temperature measurements are performed using an infrared temperature sensor. Some experimental results are also presented.*

Keywords: *Core body temperature, Generalized nets, Embedded system.*

Introduction

Continuously monitoring of core body temperature suggests measuring device to be placed on, so as not to interfere and not to bother the patient. The classical methods for measuring the core body temperature, in particular the non-invasive methods, would not be adequate for this purpose. Although the skin temperature is not a reliable method [1, 6] to estimate the core body temperature, we decided to attempt to find dependencies between body skin and environment temperatures, and conceivably the core body temperature. Assuming that such a device should be worn constantly, we decided to turn our attention to measuring the temperature of the skin of the arm, as well as the surrounding air temperature. The developed Generalized net model would be helpful in case of increasing the number and the type of the temperature sensors, as well as when using a different microcontroller.

Embedded system overview

The system comprises two separate temperature sensors and a microcontroller. In Fig. 1 and Fig. 2 are shown the simplified structures of the used sensors.

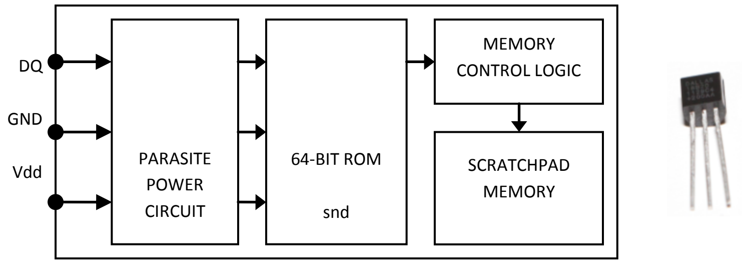


Fig. 1 Simplified block diagram of DS18B20 temperature sensor

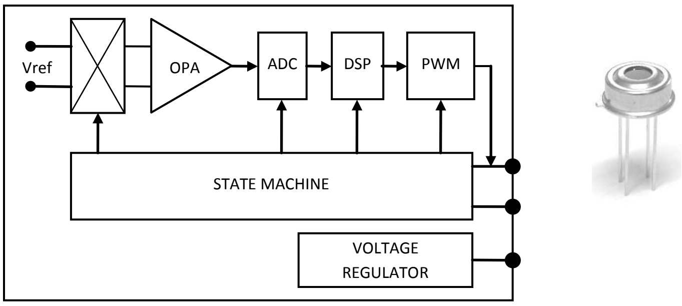


Fig. 2 Simplified block diagram of MLX90614 temperature sensor

The first one, DS18B20 [4] (Fig. 1), is used for measuring the environment temperature. The second one, MLX90614 [5] (Fig. 2) measures the infrared radiation against the sensor (in this case, the skin temperature). The communication between the sensors and the microcontroller is implemented via two separate serial communication busses: 1-Wire and I^2C.

Due to the large number of measurements are expected, the external EEPROM (Electrically Erasable Programmable Read Only Memory) memory is used. This external memory also uses the I2C bus (Fig. 3). The communication between embedded system and the PC is established via serial communication interface (SCI).

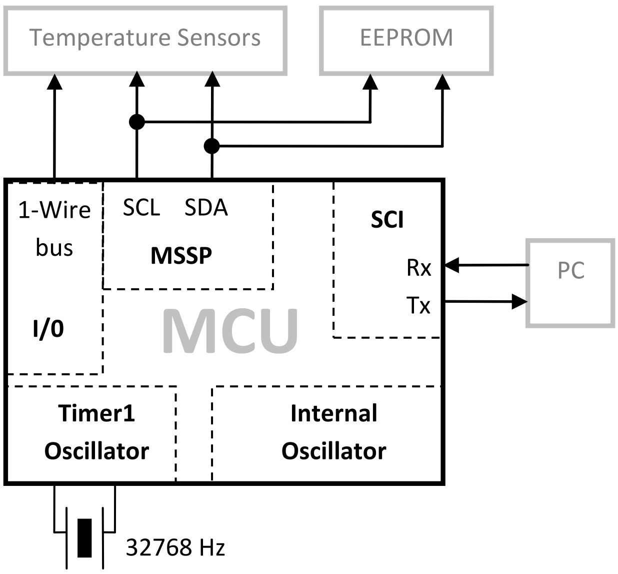


Fig. 3 Block diagram of the embedded system

During most of the time, the system operates in a low-power mode, i.e. the microcontroller unit (MCU) is in a “sleep” mode, the main (internal) oscillator is off. In this mode, only the Timer1 oscillator is active. When the Timer1 counter is overflowed, the interrupt is generated, the main oscillator is turned on and the system wakes up. During the active mode, the temperature data are read from the sensor and stored in the external EEPROM memory. Then, if there is a request from the external device (PC), all data, stored in the external memory, is transmitted via SCI interface. Then, the MCU goes into sleep mode, until the next Timer1 interrupt.

Fig. 4 shows the principle of the data arrangement in the external EEPROM memory. At the first two cells (addresses 00H and 01H) is stored the address of the memory cell, where should be stored the first byte of the next data readings. Since every data reading is summarized in four bytes, after each subsequent reading, the content of the first two cells is increased by 4. After whole memory space write, the oldest data record is overwritten.

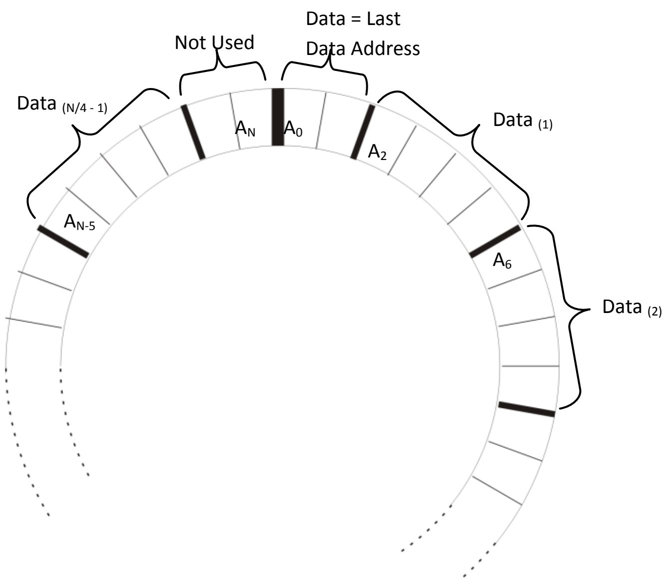


Fig. 4 Temperature data arrangement in the EEPROM memory

When the whole memory data are transmitted to the computer system, it is rearranged, so the first data reading would be the data from the address, previous to the address, written in the first two memory cells. Knowing the time of the transmitting and the period between each temperature measurements, it is easy to draw a time dependent temperature graph. Furthermore, the data are recalculated, according to the manufacturer's specifications [1, 4], as follows:

$$T_{obj} = 0.02\,(256\,t_{objH} + t_{objL}) - 273,$$

where T_{obj} is the temperature obtained from the MLX90614 temperature sensor (skin temperature), t_{objH} and t_{objL} are respectively the Most Significant Bits (MSB) and the Least Significant Bits (LSB) data values.

$$T_{env} = 16\,t_{envH} + (t_{envL} >> 4) + (t_{envL} \,\&\, 0b00001111)/16,$$

where T_{env} is the temperature obtained from the DS18B20 temperature sensor (air temperature), t_{envH} and t_{envL} are respectively the MSB and the LSB data values.

Generalized net model

All definitions related to the concept of Generalized nets are taken from [2, 3]. The net describing the work of the embedded system is shown in Fig. 5.

Initially the following tokens enter the generalized net:

- in place S_{12} – α-token with characteristic "*Timer*1 *value*";
- in place S_{73} – β-token with characteristic "*Array of temperature data*".

The generalized net is presented by the following set of transitions:

$$A = \{Z_1, Z_2, Z_3, Z_4, Z_5, Z_6, Z_7\}.$$

The particular transitions describe the following processes:

- Z_1 – Timer1 counting;
- Z_2 – Sensor data reading;
- Z_3 – Calculating the next EEPROM address for writing;
- Z_4 – SCI routines;
- Z_5 – Calculating the next EEPROM address for reading;
- Z_6 – Read/Write data from/to EEPROM;
- Z_7 – Converting the data and storing in array.

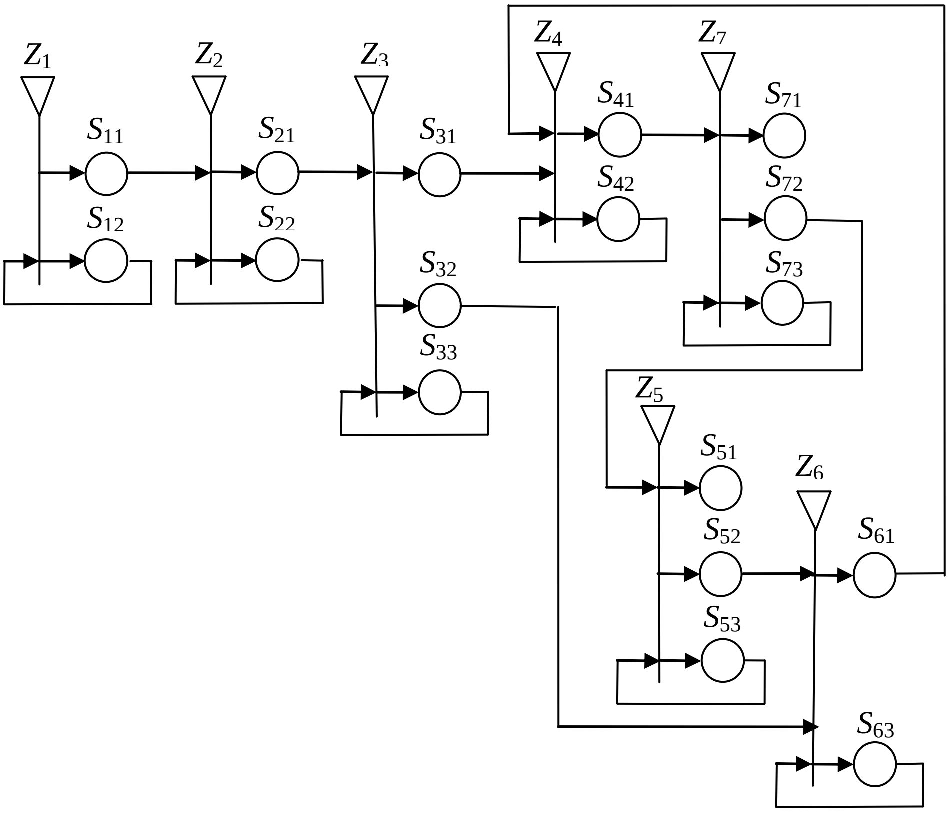


Fig. 5 GN model of the temperature data logger system

The transitions have the following description:

$$Z_1 = \langle \{S_{12}\}, \{S_{11}, S_{12}\}, R_1, \vee(S_{12})\rangle$$

$$R_1 = \begin{array}{c|cc} & S_{11} & S_{12} \\ \hline S_{12} & W_{12,11} & True \end{array}$$

where $W_{12,11}$ = "T1IF = 1" (T1IF – Timer1 Interrupt Flag).

The token that enters place S_{12} obtains characteristic "*Timer*1 *value*". The token that enters place S_{11} obtains characteristic "*Interrupt flag*".

$$Z_2 = \langle\{S_{11}, S_{22}\}, \{S_{21}, S_{22}\}, R_2, \vee(S_{11}, S_{22})\rangle$$

$$R_2 = \begin{array}{c|cc} & S_{21} & S_{22} \\ \hline S_{11} & false & true \\ S_{22} & W_{22,21} & W_{22,22} \end{array}$$

where $W_{22,21}$ = "*The sensor data are read*";
$W_{22,22} = \neg W_{22,21}$.

The token that enters place S_{22}, from place S_{11}, obtains characteristic "*Temperature sensor device address*".

The token that enters place S_{21} obtains characteristic "*Current temperature data*".

$$Z_3 = \langle\{S_{21}, S_{33}\}, \{S_{31}, S_{32}, S_{33}\}, R_3, \vee(S_{21}, S_{33})\rangle$$

$$R_3 = \begin{array}{c|ccc} & S_{31} & S_{32} & S_{33} \\ \hline S_{21} & true & false & true \\ S_{33} & false & W_{33,32} & true \end{array}$$

where $W_{33,32}$ = "*The next address for writing is calculated*".

The token from place S_{21} enters place S_{31}, where does not change its characteristic, and place S_{33}, where obtains characteristic "*EEPROM address*".

The token that enters place S_{32} obtains characteristic "*EEPROM address and temperature data*".

$$Z_4 = \langle\{S_{31}, S_{42}, S_{61}\}, \{S_{41}, S_{42}\}, R_4, \vee(S_{31}, S_{42}, S_{61})\rangle$$

$$R_4 = \begin{array}{c|cc} & S_{41} & S_{42} \\ \hline S_{31} & false & true \\ S_{42} & W_{42,41} & W_{42,42} \\ S_{61} & false & true \end{array}$$

where $W_{42,41}$ = "TRMT = 1" (TRMT – Transmitter Buffer Empty Flag);
$W_{42,42} = \neg W_{42,41}$.

The token from place S_{31} enters place S_{42}, and does not change its characteristic.

The token from place S_{61} enters place S_{42}, and does not change its characteristic.

The token that enters place S_{41} obtains characteristic "*SCI transmitted data*".

$$Z_5 = \langle\{S_{53}, S_{72}\}, \{S_{51}, S_{52}, S_{53}\}, R_5, \vee(S_{53}, S_{72})\rangle$$

$$R_5 = \begin{array}{c|ccc} & S_{51} & S_{52} & S_{53} \\ \hline S_{53} & W_{53,51} & W_{53,52} & true \\ S_{72} & false & false & true \end{array}$$

where $W_{53,51}$ = "*The maximum EEPROM address is reached*";
$W_{53,52}$ = "*The next EEPROM address is calculated*".

The token that enters place S_{51} obtains characteristic "*Sleep instruction*".

The token that enters place S_{52} obtains characteristic "*Next address*".

The token that enters place S_{53}, from place S_{72}, obtains characteristic "*Start address*".

$$Z_6 = \langle\{S_{32}, S_{52}, S_{63}\}, \{S_{61}, S_{62}\}, R_6, \vee(S_{32}, S_{53}, S_{62})\rangle$$

$$R_6 = \begin{array}{c|cc} & S_{61} & S_{62} \\ \hline S_{32} & false & true \\ S_{52} & false & true \\ S_{62} & W_{62,61} & true \end{array}$$

where $W_{62,61}$ = "*The maximum EEPROM address is reached*".

The token from place S_{32} enters place S_{63}, and does not change its characteristic.

The token from place S_{52} enters place S_{63}, and does not change its characteristic.

The token that enters place S_{61} obtains characteristic "*Data read from EEPROM*".

$$Z_7 = \langle\{S_{41}, S_{73}\}, \{S_{71}, S_{72}, S_{73}\}, R_7, \vee(S_{41}, S_{73})\rangle$$

$$R_7 = \begin{array}{c|ccc} & S_{71} & S_{72} & S_{73} \\ \hline S_{41} & false & false & true \\ S_{73} & W_{73,71} & W_{73,72} & true \end{array}$$

where $W_{73,71}$ = "*The temperature data are recalculated, rearranged and stored in the array*";
$W_{73,72}$ = "*There is a request for a whole EEPROM data reading*".

The token that enters place S_{71} obtains characteristic "*Array with recalculated temperature data*".

The token that enters place S_{72} obtains characteristic "*EEPROM read request*".

The token from place S_{41} enters place S_{73}, and does not change its characteristic.

Results and discussion

We conducted a number of measurements, placing the sensors at various locations on the arm, at different ambient air temperatures, and for different time durations. In Fig. 6 is shown a graph of one of those measurements. These are raw data (there is no additional data processing), obtained by continuous measurements, within about five hours. The graph represents the variations of the skin temperature, given that the infrared sensor was placed just above the elbow, touching the skin. The air temperature sensor was placed about an inch above the skin. The measurements until around 220th minute are made in clothed state of the arm, while the remaining measurements are made with direct exposure to the ambient air.

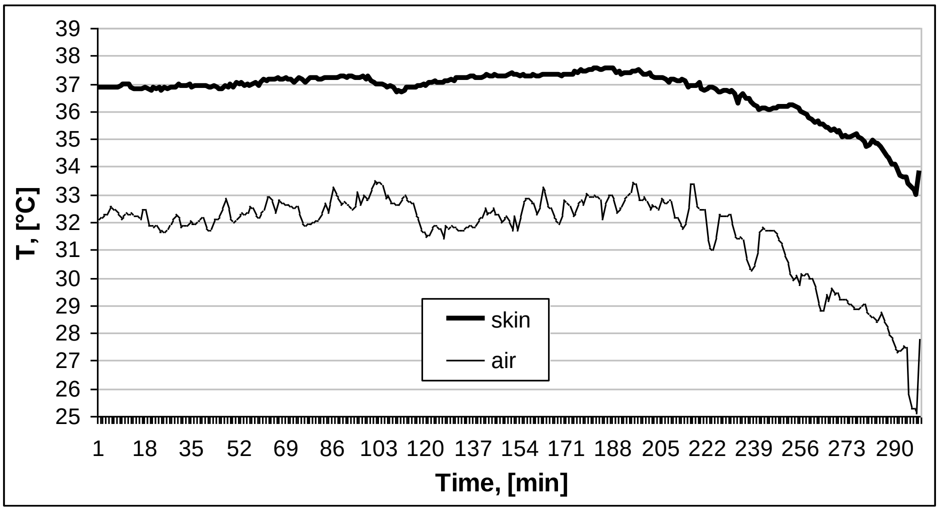


Fig. 6 The temperature of the skin and the temperature near the skin (around an inch)

Conclusion

The prototype of the system is realized and experimental results are obtained. To establish reliable dependencies between temperature of the particular place on the skin and core body temperature, it would be necessary to perform additional measurements. The measurement of the temperature of the ear canal would be the most convenient method to obtain these reference data.

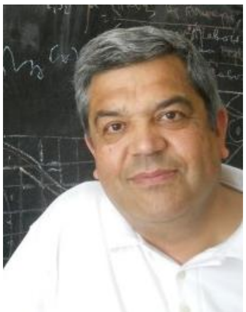
References

1. Andonov V., D. Stephanova, M. Esenturk, M. Angelova, K. Atanassov (2013). Generalized Net Model of Telemedicine Based on Body Temperature Sensors, 14th Int. Workshop on Generalized Nets, Burgas, November 29-30, 78-89.
2. Atanassov K. (1991). Generalized Nets, World Scientific, Singapore, New Jersey, London.
3. Atanassov K. (2007). On Generalized Nets Theory, Prof. M. Drinov Academic Publishing House, Sofia.
4. Maxim Integrated (2008). DS18B20 Programmable Resolution 1-Wire Digital Thermometer, http://datasheets.maximintegrated.com/en/ds/DS18B20.pdf (Last accessed 15 June 2015).
5. Melexis MIS (2013). MLX90614 Family: Single and Dual Zone Infra Red Thermometer in TO-39, http://www.adafruit.com/datasheets/MLX90614.pdf (Last accessed 15 June 2015).
6. Moran D. S., L. Mendal (2002). Core Temperature Measurement: Methods and Current Insights, Sports Medicine, 32(14), 879-885.

Lenko Erbakanov, M.Sc.
E-mail: erbakanov@btu.bg

Lenko Erbakanov received his Master degree (2012) from Department of Electronics, Prof. D-r Assen Zlatarov University, Burgas, Bulgaria. His current research interests include embedded systems design, fuzzy logic, and neural networks.

Prof. Krassimir Atanassov, D.Sc., D.Sc.
E-mail: krat@bas.bg



Krassimir Atanassov graduated Mathematics in the Sofia University, Sofia, in 1978, and defended his Ph.D. in 1986. He became Doctor of Technical (Computer) Sciences in 1997, with a thesis on Generalized Nets, Full Professor in 1998 and Doctor of Mathematical Sciences in 2000, with a thesis on Intuitionistic Fuzzy Sets. He is a Corresponding Member of the Bulgarian Academy of Sciences since 2012 and Fellow of IFSA since 2013. Since 1994 he works in the Institute of Biophysics and Biomedical Engineering at the Bulgarian Academy of Sciences, being a head of Department of Bioinformatics and Mathematical Modelling.

Assoc. Prof. Sotir Sotirov, Ph.D.
E-mail: ssotirov@btu.bg

Sotir Sotirov graduated from Department of Electronics, Technical University in Sofia, Bulgaria. He received his Ph.D. degree from Prof. D-r Assen Zlatarov University, Burgas in 2005. His current research interests are in the field of modeling of neural networks.