Home Automation Based on 1-Wire Interface

Marin Marinov[1], Ivan Topalov[2], Todor Djamiykov[3]

Abstract – **In this paper we are presenting a way for automating home processes using a wired network. The network is made using Dallas Semiconductors 1-Wire interface and devices. The control is achieved through a personal computer and a web browser. The automation includes heating and cooling, lighting control, etc.**

Keywords – **1-Wire interface, Home automation, Sensor networks.**

I. INTRODUCTION

The concept of an intelligent house, where devices could be controlled remotely, and would be automated for both comfort and energy efficiency is very appealing. The interface to the devices could be either wireless or wired. In case of a wireless solution there will be a need either of batteries or either of power lines. Our solution is positioned between both strategies. It is a wired one, but uses the power lines also for data transfers. This leads to a total of two wires needed for each sensor or controlling device. In some cases the number of needed wires can be reduced even to one.

1-Wire protocol requires a master device. We have chosen to use Dallas Semiconductors Kit – TINI. It uses DS80C400 microcontroller which is a 1-Wire and Ethernet ready microcontroller.

All the control of the master and 1-Wire devices is achieved through a personal computer and the use of a Web Browser. The use of this program suggests a real easy and user friendly interface.

II. 1-WIRE PROTOCOL

The 1-Wire net is a low-cost bus based on a PC or microcontroller communicating digitally over twisted-pair cable with 1-Wire components. The network is defined with an open-drain master/slave multidrop architecture that uses a resistor pull-up to a nominal 5 V supply at the master. A 1-Wire net-based system consists of three main elements: 1) a bus master with controlling software; 2) wiring and associated connectors; and 3) 1-Wire devices. The 1-Wire protocol uses conventional CMOS/TTL logic levels with operation specified over a supply voltage range of 2,8 V to 6 V.

The 1-Wire net allows tight control because no node is allowed to speak unless requested by the master, and no communication is allowed between slaves, except through the master. Both master and slaves are configured as transceivers permitting bit sequential data to flow in either direction, but only one direction at a time. Technically speaking, data transfers are half-duplex and bit sequential over a single pair of wires, data and return, from which the slaves "steal" power by use of an internal diode and capacitor. Data is read and written least significant bit first. Data on the 1-Wire net is transferred with respect to time slots. For example, to write a logic one to a 1-Wire device, the master pulls the bus low and holds it for 15 microseconds or less. To write a logic zero, the master pulls the bus low and holds it for at least 60 microseconds to provide timing margin for worse case conditions (Fig. 1).

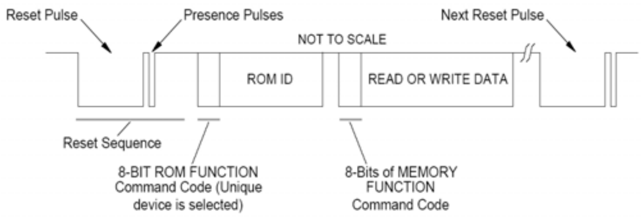

Fig. 1. A typical 1-Wire communication sequence.

A system clock is not required, as each 1-Wire part is self clocked by its own internal oscillator that is synchronized to the falling edge of the master. Power for chip operation is derived from the bus during idle communication periods when the DATA line is at 5V by including a half wave rectifier onboard each slave.

Within each 1-Wire slave is stored a lasered ROM section with its own guaranteed unique, 64-bit serial number that acts as its globally unique address.

The entire collection of 1-Wire devices plus the host form a type of miniature local area network, or MicroLAN; they all communicate over the single common wire. The most significant byte in the ROM code of each device contains a Cyclic Redundancy Check (CRC) value based on the previous 7 bytes of data for that part. When the host system begins communication with a device, the 8-byte ROM is read, LSB first. If the CRC that is calculated by the host agrees with the CRC contained in byte 7 of ROM data, the communication can be considered valid. If this is not the case, an error has occurred and the ROM code should be read again.

The length of the maximum wiring depends on the type of wires used. When using Category 5 twisted pair, controlling

[1]Marin Marinov, Faculty of Electronics, Technical University of Sofia, 8 Kliment Ohridski Blvd., Sofia 1756, Bulgaria, E-mail: mbm@tu-sofia.bg
[2]Ivan Topalov, Faculty of Electronics, Technical University of Sofia, 8 Kliment Ohridski Blvd., Sofia 1756, Bulgaria E-mail: vantop@hotmail.com
[3]Todor Djamiykov, Faculty of Electronics, Technical University of Sofia, 8 Kliment Ohridski Blvd., Sofia 1756, Bulgaria. E-mail: tsd@tu-sofia.bg

slew rates and substituting an active pull-up, reliable communication over 300m of cable with more than 500 assorted 1-Wire devices was demonstrated. Without slew rate control and active pull-up, the limit is about 100 meters with 150 1-Wire devices.

II. BLOCK DIAGRAM

As we know how the 1-Wire protocol works we are ready to present the structure of the whole system. Of course it consists of a master node and slave nodes. All the computation is done in the master node, while the slave nodes represent different sensors and executing devices.

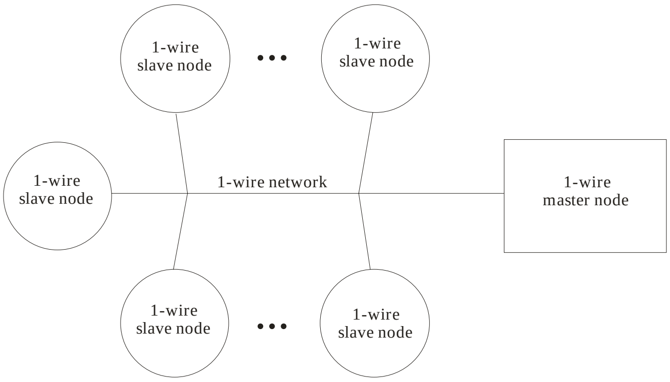

Fig. 2. Block diagram of an automation system

As it is shown on Fig. 2 there is an unlimited number of 1-Wire nodes connected to the network. The 1-Wire network shown on the figure consists only of one wire is it is the real situation. We are not counting the ground, because in some cases the central heating or water pipes can be used for ground wire.

The principle of operation is as the following: The master node asks each sensor slave node constantly for its status. When the master node determines that a given sensor has passed over a predefined limit it tells another slave 1-Wire node to react according to an algorithm defined by the user. A simple example is controlling the temperature in a room. The user sets 20ºC the temperature. Then the master node checks the temperature sensor in the observed room, and in case it sees deviation from the setup temperature it reacts. If the temperature is below the threshold and it is a heating season, the node tells the execution slave node, in our case a relay, to open the heating electro-magnetic valve. When the temperature reaches the threshold, master node tells the execution slave node to shut the valve.

III. SENSOR AND ACTUATOR SLAVE NODES

Dallas Semiconductors offers a variety of 1-Wire devices. The simple of all these devises is the temperature sensor. The biggest advantage of this sensor is that the measured value is ready for calculations – it is coming out of the sensor straight in a two complement value. We have stopped our attention at the use if DS18B20. The core functionality of it is its direct-to-digital temperature sensor. The resolution of the temperature sensor is user-configurable to 9, 10, 11, or 12 bits, corresponding to increments of 0.5ºC, 0.25ºC, 0.125ºC, and 0.0625ºC, respectively. The default resolution at power-up is 12-bit. The DS18B20 powers-up in a low-power idle state; to initiate a temperature measurement and A-to-D conversion, the master must issue an appropriate command. Following the conversion, the resulting thermal data is stored in the 2-byte temperature register in the scratchpad memory and the DS18B20 returns to its idle state.

One of the biggest advantages of this sensor is its size. It is fitted in a TO-92 and has three pins, one pin is common – ground, one is data IO – DQ pin, and a third pin for power, in case the user wants to use a separate power source. Fig. 3. illustrates TO-92 case for DS18B20.

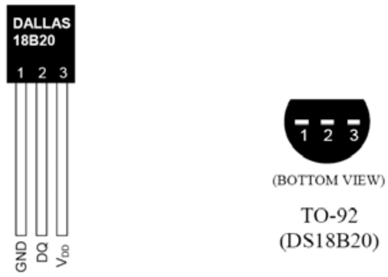

Fig. 3. TO-92 case of DS18B20

The usage of this sensor is very simple. Just plug it to the wires and it is ready for use. The only important thing is the user to know which sensor in which room is placed. The rest is achieved through software and 1-Wire protocol.

The practical use of the temperature sensor is very popular because of its function to monitor a predefined temperature. In both occasions – heating and cooling, the control of the room temperature is achieved through measuring the temperature in the particular room. For better performance we use several temperature sensors in one room and then average the temperature reading from all sensors. Thus we achieve better control, because the temperature close to the source of heat or cold is different from the one in the other corner of the room. Also the temperature close to outer walls and windows is different. We also use a temperature sensor placed on top of the source of heat or cold to observe and control the device itself. This is used like a precaution safety backup switch in case of failure in the device producing heat or cold.

All the information from the temperature sensors goes to the master 1-Wire device. It reads the temperature and reacts accordingly the predefined algorithm.

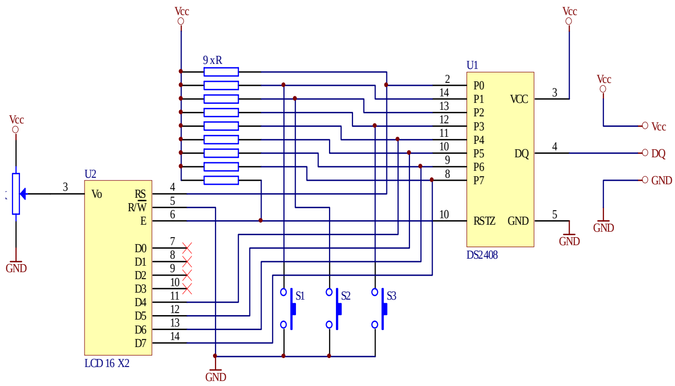

Fig. 4. Display and buttons module

The temperature thresholds are stored in the 1-Wire master device and are entered through a personal computer and a web browser.

In case the user wants to set the temperature without using a computer, and this setup to be done in each room, he can use a separate module, placed in each room for setting the temperature. In this module there are several buttons and a LCD display. All the data entered through this module and displayed on the LCD are communicated again through 1-Wire protocol. In other words, this is a monitoring and control module connected to the 1-Wire master device. There are no control functions in this module, it is only an interface for the user. The main part of this module, as shown on fig. 4., is the 1-Wire 8-channel addressable switch.

The DS2408 is an 8-channel, programmable I/O 1-Wire chip. PIO outputs are configured as open-drain and provide an on resistance of 100 Ω max. A robust PIO channel-access communication protocol ensures that PIO output-setting changes occur error-free. A data-valid strobe output can be used to latch PIO logic states into external circuitry such as a D/A converter (DAC) or microcontroller data bus.

DS2408 operation is controlled over the single-conductor 1-Wire bus. Device communication follows the standard Dallas Semiconductor 1-Wire protocol. Each DS2408 has its own unalterable and unique 64-bit ROM registration number that is factory lasered into the chip. The registration number guarantees unique identification and is used to address the device in a multidrop 1-Wire net environment. Multiple DS2408 devices can reside on a common 1-Wire bus and can operate independently of each other. The DS2408 also supports 1-Wire conditional search capability based on PIO conditions or power-on-reset activity; the conditions to cause participation in the conditional search are programmable. The DS2408 has an optional V_{CC} supply connection. When an external supply is absent, device power is supplied parasitically from the 1-Wire bus. When an external supply is present, PIO states are maintained in the absence of the 1-Wire bus power source. The RSTZ signal is configurable to serve as either a hard-wired reset for the PIO output or as a strobe for external circuitry to indicate that a PIO write or PIO read has completed.

Devices described up to now were for sensing and setting the temperature thresholds. We need also devices for controlling the heating or cooling devices. The simplest way is to put a relay which is controlled over 1-Wire interface.

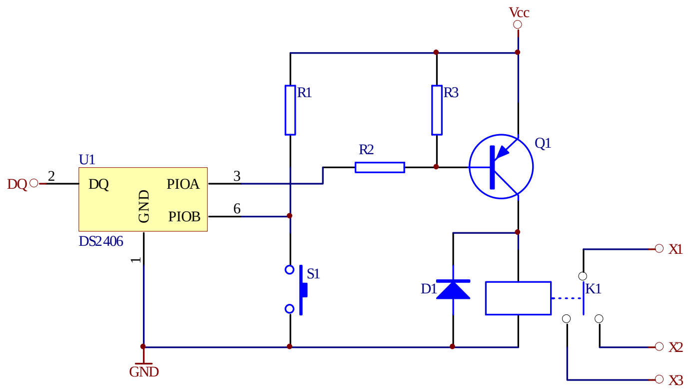

Fig. 5. 1-Wire relay and button

The schematic for this solution is shown on Fig. 5.

It is a simple solution based on DS2406 – dual addressable switch plus 1kbit memory. The DS2406 offers a simple way to remotely control a pair of open drain transistors and to monitor the logic level at each transistor's output via the 1-Wire bus for closed loop control. Each DS2406 has its own 64-bit ROM registration number that is factory lasered into the chip to provide a guaranteed unique identity for absolute traceability.

The device's 1024 bits of EPROM can be used as electronic label to store information such as switch function, physical location, and installation date. Communication with the DS2406 follows the standard Dallas Semiconductor 1-Wire protocol and can be accomplished with minimal hardware such as a single port pin of a microcontroller.

Multiple DS2406 devices can reside on a common 1-Wire network and be operated independently of each other. Individual devices will respond to a Conditional Search command if they qualify for certain user-specified conditions, which include the state of the output transistor, the static logic level or a voltage transition at the transistor's output.

In our case, one of the PIO pins is used for controlling the relay and the other is for reading the status of a button. The button might be used for manual control of given process or device. Except controlling heating and cooling devices, it can be used also for controlling lights in the home.

Such board can be fitted also in the wall lights switch. And by pressing the button it will turn on and off the lights in the room and also gives the opportunity all the lights in the home to be controlled by one single place.

When there is no need for a button the solution shown on Fig. 6. can be used. It uses DS2405 instead DS2406 as on Fig. 5.

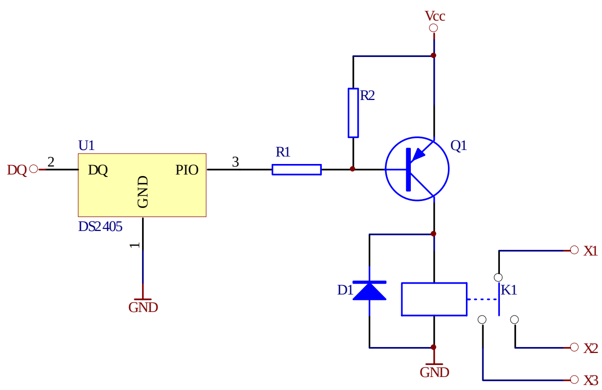

Fig. 6. Single 1-Wire relay

The DS2405 Addressable Switch is an open drain N-channel transistor that can be turned on or off by matching the 64-bit factory-lasered registration number within each part. If the relay is replaced with an LED we receive a remote 1-Wire indicator. Fig. 7 shows the proposed solution for 1-Wire indicator.

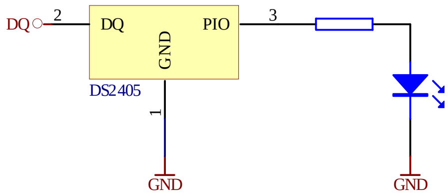

Fig. 7. 1-Wire LED indicator

All these intelligent slave nodes are connected to the 1-Wire master device. The master node in our case is doing all the computation and storing of data. It can be a simple microcontroller, a personal computer or a high performance microcontroller.

IV. 1-Wire Master Node

Our solution for the master 1-Wire node is based on a high performance microcontroller from Dallas Semiconductor DS80C400. The DS80C400 network microcontroller offers the highest integration available in an 8051 device. Peripherals include a 10/100 Ethernet MAC, three serial ports, a CAN 2.0B controller, 1-Wire® Master, and 64 I/O pins.

To enable access to the network, a full application-accessible TCP IPv4/6 network stack and OS are provided in ROM. The network stack supports up to 32 simultaneous TCP connections and can transfer up to 5 Mbps through the Ethernet MAC. Its maximum system-clock frequency of 75 MHz results in a minimum instruction cycle time of 54ns. Access to large program or data memory areas is simplified with a 24-bit addressing scheme that supports up to 16MB of contiguous memory.

To accelerate data transfers between the microcontroller and memory, the DS80C400 provides four data pointers, each of which can be configured to automatically increment or decrement upon execution of certain data pointer-related instructions. The DS80C400's hardware math accelerator further increases the speed of 32-bit and 16-bit multiply and divide operations as well as high-speed shift, normalization, and accumulate functions.

DS80C400 is the core of the evaluation module DSTINIm400. In addition to the DS80C400, the DSTINIm400 includes a real-time clock, 1MB flash, 1MB static RAM, and support for an external Ethernet PHY for connecting to a wide variety of networks. The circuit board is designed as a module to be plugged into a 144-pin SODIMM connector. For evaluation, the DSTINIm400 can be inserted into the DSTINIs400 socket board (for future use, both boards are referred to as a TINI module). The combination of the two boards allows full evaluation of the features of the DS80C400 using an Ethernet network.

As described before, TINI provides all the necessary to run a smooth 1-Wire network. It has memory to store the firmware, 1-Wire outputs for the network, and Ethernet connectivity for personal computer operation. The operation is simple. All devices, sensors, buttons, and relays are 1-Wire devices and all are connected to the TINI. TINI from its side is doing all the computation, monitors the sensors and controls the relays. Besides this, TINI is serving a web server, which server is the door to a user interface. TINI is connected to an Ethernet Local Area Network (LAN) where the HTTP server is accessible. A personal computer is also connected to the LAN and accesses the web server through a Web Browser. The purpose of this solution is to give even to non technical people the ability to use and setup the home automation system. This is the main reason to use a web browser.

The options that can be set up through the web browser are a lot. First and important option is to designate a name for each slave node. This means to give a name to the unique 64-bit ROM registration number in each 1-Wire device. The name should be also unique and should hint the type of node – sensor or execution device, and the room in which it is placed. After the names are setup comes the time to define the function for each execution device that should do. If it is a heating device, the user makes a relation with the sensors placed in the same room and sets the temperatures for thresholds.

So, we have a heating occasion, the execution device when turned on turns the power on for the heating device. The TINI checks the temperature in the room in regular intervals predefined by the user. When the temperature goes over the threshold for the room it turns off the heating system. But the algorithm continues to work and continues to measure the temperature.

The web browser is the interface for the user to setup the algorithms. All the settings are stored in the TINI module, which means that the personal computer can be turned off, while the home automation system can continue to work. There is also a battery on the TINI module which can keep the data in case of power failures. Another option is to use the real time clock on the TINI. This gives the option to set up some devices to turn on and off at particular time.

In some occasions the setup of the temperatures for the rooms can be done through the use of the 1-Wire IO module with LCD display and buttons. This module is placed in each room and is accessible for the user.

V. Conclusion

With the help of the variety of 1-Wire devices a cheap and easy to design and implement home automation system can be achieved. The control and monitoring includes individual room heating and cooling, lights, open door checking, etc. The master device is a module based on DS80C400. The module is doing all the computation and is accessed via a web browser on a personal computer. The web browser reveals the potential and simplicity for use.

References

[1] Awtrey, D., Transmitting Data and Power over the 1-Wire Bus," Sensors: 48-51, Feb. 1997.
[2] Awtrey, D., The 1-Wire Weather Station, Sensors: 34-40, June 1998.
[3] Dallas Semiconductor. MicroLAN Design Guide, Sep. 2003.
[4] Hill, J., R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister. System architecture directions for networked sensors. In Proceedings of the 9th International Conference on

Architectural Support for Programming Languages and Operating Systems ASPLOS-IX, pages 93–104, Cambridge MA, USA, November 2000.
[5] Howarth, D.: Performing Data Acquisition over the Internet. Sensors Magazine, January 1998.
[6] Yamasaki, H. K. Takahashi.: Advanced intelligent sensing system using sensor fusion. Proceedings of the 1992 International Conference on Industrial and Electronic Instrumentation and Automation. Vol. 1, 1993.