A Realization of Temperature Monitoring System Based on Real-Time Kernel μC/OS and 1-wire Bus

Haibo Pu, Lijia Xu, and Yanmei Qi

***Abstract*—The traditional temperature monitoring system generally adopt some analog sensors for collecting data and a microcontroller for processing data for the purpose of temperature monitoring. However, this back-fore ground system has the disadvantages that the system has poor real-time property and single function, the amount of sensors is not easy to expand, and the software system has a difficulty in upgrading. Aiming at these disadvantages, the system designed in this paper adopts brand-new hardware and software structures: a digital temperature sensor array is connected to 1-wire bus and communicated with a control core through 1-wire bus protocol, thus a great convenience is provided for the expansion of the sensor; a real-time operating system is introduced into the software, an application program capable of realizing various functions runs on the real-time kernel μC/OS-II platform. The application of the real-time kernel also provides a good lower layer interface for the late-stage software upgrading.**

***Index Terms*—1-wire bus, temperature monitoring system, DS18B20, μC/OS-II, C8051F020.**

Original Research Paper
DOI: 10.7251/ELS1317016P

I. INTRODUCTION

THE traditional temperature monitoring system generally adopts an analog sensor for collecting data and a microcontroller for processing data for the purpose of temperature monitoring. However, this back-fore ground system has the disadvantages that the system has poor real-time property and single function, the amount of sensors is not easy to expand, and the software has a difficulty in upgrading. To overcome the disadvantages of poor real-time property, single function, difficulty in expansion and maintenance of the sensor and difficulty in upgrading of the software system in the traditional real-time temperature monitoring system, this paper provides a real-time temperature monitoring system (temperature Monitoring System, hereinafter called TMPSYS for short) based on a real-time kernel μC/OS and a 1-wire bus. After test, the system can effectively realize multitask scheduling, and accomplish the system function. Compared with the traditional back-fore ground system, especially when adding tasks or sensor nodes, the kernel μC/OS with high real-time property can give a response within a short period time. The response time is mainly determined by both the selected switching frequency of μC/OS-II such as 50 Hz and the performance of CPU. The kernel μC/OS can also enable the C8051F series single-chip microcomputer to realize more complex functions. Another convenience of introducing μC/OS into the system is upgrading of application in the future almost does not affect the system's real-time characteristics. The sensors based on a 1-wire bus is simple and reliable in physical connection, is convenient for expansion and maintenance of a bus, and particularly suitable for monitoring multipoint temperature [1].

Manuscript received 7 April 2013. Received in revised form 24 May 2013 and 9 June 2013. Accepted for publication 11 June 2013.

This work is supported by the Natural Science Project of Sichuan Education Department under Grant 12ZA277.

H. Pu, L. Xu, and Y. Qi are with the School of Information and Engineering Technology, Sichuan Agricultural University, Ya'an, China (Corresponding author is Lijia Xu, e-mail: xulijiasc@163.com).

II. DESIGN OF HARDWARE SYSTEM

The traditional analog sensor lies in that a sensor occupies a channel; however, the limited number of an available channel on the chip of an MCU limits the number of the sensor. Although the number of the sensor can be increased through an expansion port, when a plurality of sensors transmit data in a parallel transmission manner, electromagnetic interference between the channels is easily caused, and an expanded off-chip port cannot be compared with an on-chip port in speed and stability. However, the problem above can be better solved by adopting a digital sensor for transmitting data in a serial transmission manner [2]. Therefore, the 1-wire bus technology is adopted in the TMPSYS, and all the temperature sensors are connected to a 1-wire bus to form a sensor array, as shown in Fig. 1.

The system adopts two kinds of bus expansion technologies. The temperature acquisition parts are connected through a 1-wire bus, a simple 1-wire bus temperature sensor DS18B20 is only required to be accessed on a bus in the actual physical connection, the real-time clock adopts a DS1338 based on an I2C bus protocol, and more peripheral devices also can be connected to the I2C bus. The specific circuit diagram of the system is shown in Fig. 2.

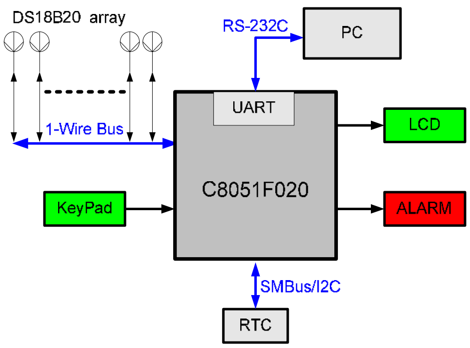

Fig. 1. Minimum system of the TMPSYS.

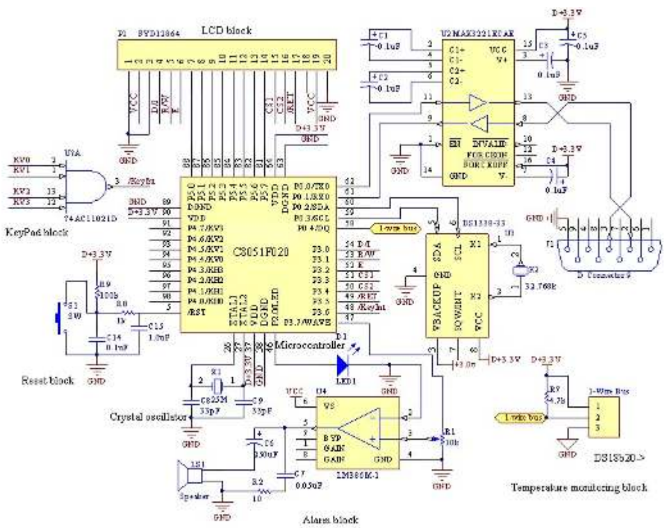

Fig. 2. Circuit diagram of the minimum system of the TMPSYS.

In Fig. 2, alarm block is used to give alarm for high or low temperature and temperature monitoring block is used to measure temperature. Crystal oscillator block is applied to produce crystal frequency for microcontroller C8051F. Reset block can reset signal when the C8051F is powered on and can cancel the reset signal until the power system stability. KeyPad block can set temperature manually. The DS1338 serial real-time clock is a low-power, full binary-coded decimal clock/calendar plus 56 bytes of NV SRAM. Address and data are transferred serially through an I2C interface. The MAX3221ECAE are 3V-poweredEIA/TIA-232 and V.28/V.24 communications interfaces with automatic shutdown/wakeup features, high data-rate capabilities, and enhanced electrostatic discharge protection. FYD12864 is a liquid crystal display module used to show temperature.

A. C8051F Series Microcontroller

The C8051F series microcontroller is configured with a standard JTAG interface, the non-invasive on-chip debugging can be directly performed to the target system through the JTAG interface, the reading/writing operation of Flash is supported, and the JTAG logic also provides boundary scanning function for the system test [3, 4]. Under the programmed control of a boundary register, the weak pull-up functions of all pins, SFR buses and I/O ports of the device can be observed and controlled.

B. 1-wire Bus Temperature Sensor DS18B20

The 1-wire bus adopts a single signal wire for completing the bidirectional data transmission [5], and meanwhile, power supply is provided for the 1-wire bus device through the signal wire. A master-slave data exchange network can be formed conveniently by adopting 1-wire bus interface chips, and the number of slave devices in a network varies from a few to several thousand, and is almost unlimited theoretically. The 1-wire bus system has the characteristics of high networking speed and low cost, most devices depend entirely on power supply obtained from data lines, thus the 1-wire bus device has lower power consumption, and is a new selection of the field bus technology. The DS18B20 is a digital temperature sensor produced by the Maxim Company in USA, and has the advantages of simple structure, flexible operation and no external circuit, and the temperature warning limit can be set by users [6].

III. Design of Software System

A. Introduction of Real-time Kernel μC/OS-II

In a traditional back-fore ground system, the application program is designed into an endless cycle, and the key operation with strong time correlation is executed through an interruption service program, thus the back-fore ground system has poorer real-time property in information processing, and the change of codes decreases the stability in response time of task level.

In the real-time system structure, the interruption service program can directly send a request of immediate execution to task-level codes. The real-time kernel can allow various task codes and the codes relatively urgent in a subprogram to run. The change of task codes does not affect the stability in response time of task level.

To ensure the real-time property of the system and the convenient subsequent software upgrading, the real-time kernel μC/OS-II with good stability and reliability is introduced in the system, and all the task-level codes run on the kernel platform [7].

B. Transplantation of Real-time Kernel μC/OS-II on C8051F020

The μC/OS-II is a hard real-time kernel of an open source code, has good stability and reliability, and meets the standard of RTCA/DO-178B [8, 9]. The selected software build environment in the system is Keil μVision 3.53 for C51, because there is no software interruption in the C8051F020, some programs are required to be designed artificially for transplanting a kernel on an MCU without software interruption, so that the kernel can run normally.

1) Configuration of System Stack and Task Stack

As shown in Fig. 3, the OSTCBStkPtr in the OS_STK structure in the μC/OS-II is used for protecting a stack pointer of a task stack, and the pointer points at a contiguous region in an external data memory. A simulated stack is used for preserving formal parameters and local variables of a reentrant

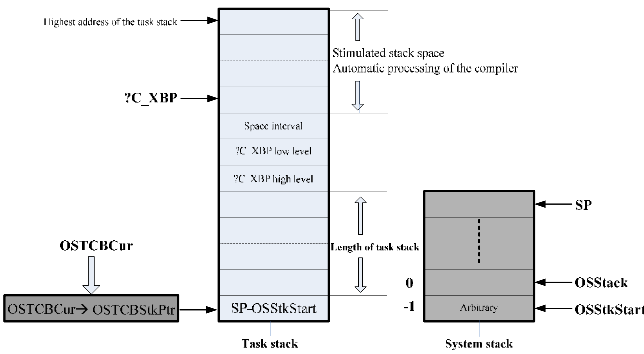

Fig. 3. Task stack frame customized for the C8051F020.

function, and grows downwards. A system stack occupies the IDATA area in a memory, and the area is the part where the data storage speed is the fastest.

During the system transplantation, to hold the variables in each task stack and the application program, the whole system is compiled in a large mode, most of the data structures are assigned in an external data memory XRAM through a compiler, and the external space has a 16-bit address (The addressing space of the C8051F020 is 0x0000-0x0FFF).

2) Overwriting of Portable File

The data type supported by the C8051F020 is compatible with the standard MCS-51 [10], and the two files are only required to be overwritten for use. The OS_CPU.H includes constants and macro definition related to the processor, data type related to the compiler and codes related to the processor. The OSTaskStkInit() is only required to be corrected for initializing the task stack in the OS_CPU_C.C, the structure of the task stack must be determined before overwriting, and all registers at the task site are preserved in the task stack.

Because many assembly language functions in the OS_CPU_A.ASM are invoked by C language functions in the μC/OS-II, thus they should be described in a principal function. And in an assembly program, a pseudo instruction "SEGMENT CODE" should be declared as a relocatable segment type, the function names are translated according to different circumstances; the symbols used in other modules should be declared with a pseudo instruction "PUBLIC", and the foreign symbols should be declared with a pseudo instruction "EXTRN".

During the late-stage application programming process, if interrupt processing is involved, the assembly codes at an interrupt entry should be written into this file for convenient unified management.

3) Calculation of Clock Frequency

The timing interruption of a timer 0 brings extra load of 10%-20% to the MCU, thus the extra load caused by the kernel should be reduced as possible on the premise of ensuring the actual demand, and the higher its frequency is, the greater the system overhead is, thus the frequency of the system is 50 Hz (the commonly-used frequency range in the μC/OS-II is 0-100 Hz).

In the TMPSYS, the C8051F020 adopts an external crystal oscillator (25 MHz) as a clock source. According to the initial value of the timer, the calculating formula is as follows:

$$(2^{16} - X) \times F = \frac{F_{OSC}}{12} \tag{1}$$

Where F represents clock frequency TICK, $Fosc$ represents crystal frequency, and X represents the initial value of a counter. The equation $F = 50$ Hz is substituted into the formula to obtain $X = 5D3D_H$, that is, $TH0 = 5D_H$, and $TL0 = 3D_H$.

C. Design of Application Program

1) Analysis of Application Program

The application program is developed on a microcontroller successfully transplanted with a real-time kernel, the system service provided by the kernel can be invoked conveniently for programming, and the real-time property, stability and reliability are all greatly enhanced.

The system can realize three basic functions to reduce power consumption: single-machine multi-point temperature measuring, single-machine multi-point temperature monitoring and online multi-point temperature monitoring, and the task created in these modes is different for reducing power consumption and increasing performance of the system. In the single-machine multi-point temperature measuring mode, the system is only used as a temperature measuring instrument, and data acquisition and task processing are only required; In the single-machine multi-point temperature monitoring mode, it must have warning function under the premise of temperature measurement, and an exception processing task is required for recording the time of occurrence of temperature anomaly; In the online multi-point temperature monitoring mode, it must be communicated with a computer, a serial communication task is required [11], the system is just used as a terminal machine at the moment, a user monitors the computer, and data processing on the terminal machine is not required [12]. Particularly, as the keyboard is an important channel for interaction between a user and the system, it can work in any working mode.

2) Task Design

Task design is mainly used for partitioning tasks, allocating the task priority and determining a communication mechanism between tasks. The sensor cannot work normally before receiving the command, a task TMPStart must be created for periodically sending a temperature acquisition command to the sensor array by using a timing function OSTimeDlyHMSM() of the system, and the frequency of sending commands must be set within the scope of working frequency of the sensor. A binary semaphore SemTMPRead is used for synchronizing data reading tasks TMPRead. The periodicity of the task TMPStart cannot be destroyed according to the requirement of real-time property, thus the priority of the task TMPStart is set highest to ensure the integrity of data acquisition.

The TMPRead stores the read data in a buffer area TMPQueue, the space of the buffer area is dynamically allocated via system service, and the data is processed through a follow-up task in a first-in first-out queuing way.

The data processing task TMPProcess can run only when there is data in the buffer area TMPQueue, and it's used for displaying the formatted temperature data through an LCD, judging the excessive temperature point(out of the set threshold of temperature), and starting the exception processing task TMPExpt through a mailbox MboxExpt. A straight selection sorting algorithm is used for searching the excessive temperature point during the process of judging the excessive monitoring point.

In the straight selection sorting process, the selection and exchange of n-1 times are carried out, the comparison of n-i times is required in every time of selection, wherein 1≤i≤n-1, and the number of movements is at most three in exchange per time, then the total number of comparisons:

$$C = \sum_{i=1}^{n-1}(n-i) = \frac{1}{2}(n^2 - n) \qquad (2)$$

Because of the number of movements is at most three in exchange per time, then the maximum number of movements is $3(n-1)$.

The time complexity of straight selection sorting is $O(n^2)$, the straight selection sorting speed is higher than the straight insertion sorting speed, and the storage space is less used, thus the straight selection sorting algorithm is suitable for a system with small data storage space.

The monitoring point exception processing task TMPExpt controls the specific hardware to give an alarm response, and searches and records the time of occurrence of anomaly, and its priority should be higher than that of the data processing task. To avoid the loss of input information, a queue buffer area KeyQueue is used to store the keyboard scan code, the keyboard interrupt service program generates a scan code and writes it into the KeyQueue. If there is data in the buffer area, the keyboard task KeyTask executes the related operation, and is triggered by an external event, and its priority is set lowest.

From the analysis above, six tasks and the corresponding priorities (the highest priority is 2) can be obtained, as shown in Table I, and the operating mechanism of the whole software system is shown in Fig. 4.

D. Search of 1-wire Bus Device

The purpose of search is to obtain values (as shown in Table II) stored in an ROM in each device, because the MCU controls the corresponding devices by using the serial number.

1) Search Algorithm

The search algorithm adopts a binary tree structure, and the search process is carried out along each partial node until the leaf nodes of an ROM code of the device are found [13]. The subsequent search is carried out along other paths on the node,

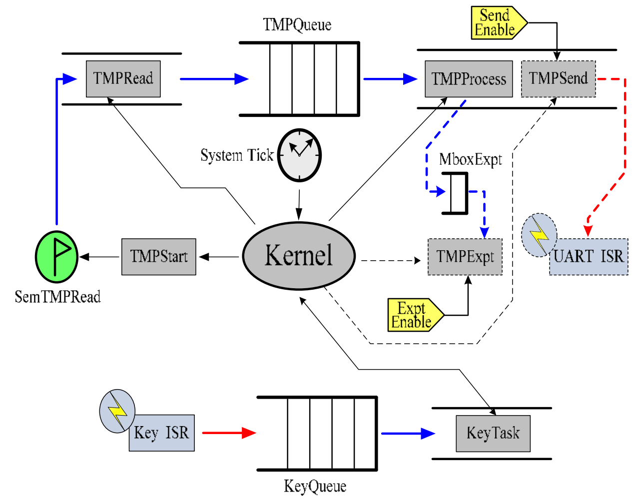

Fig. 4. Operating mechanism of the TMPSYS.

until the registration codes of all the devices on the bus are found in the same way.

2) Search Process

After the bus reset, the host sends a search command for starting the actual search process. First, all the sensors on the bus send the first bit (least significant bit) in the ROM code at the same time. In the 1-wire communication, no matter the host reads data or data is written into the slave device, the 1-wire host starts the operation of each bit. According to the characteristics of 1-Wire, when all the slave devices respond the host at the same time, the outcome is equivalent to the logic that all data bits are sent; after the slave sends the first bit in the ROM code, the host starts the next operation, and then the slave sends a complementary code of data of the first bit in its ROM code; a judgment shown in Table III can be made to the first bit in the ROM code according to the data bit read twice.

E. Monitoring Program on PC side

The monitoring program (as shown in Fig. 5) of a PC machine realizes the serial communication between a terminal machine for the TMPSYS and the PC machine, performs analysis processing to the data acquired from 12 sensors, and finally, displays the temperature data at the monitoring point on a program interface. And when a monitoring point is out of the set threshold of temperature, the monitoring program also gives an alarm signal; a control panel of an operational program can achieve the purposes of adding the monitoring point and

TABLE I
TASK PRIORITIZATION

Task [a]	TStart	TRead	TExpt	Process	TSend	KeyTask
Priority	2	3	4	5	6	7

[a] TStart = TMPStart , Tread = TMPRead , TExpt = TMPExpt , Process = TMPProcess , TSend = TMPSend .

TABLE II
64-BIT REGISTRATION CODE IN THE ROM IN THE 1-WIRE BUS DEVICE

MSB 64-bit registration code in the ROM LSB		
8-bit CRC check code MSB LSB	48-bit serial number MSB LSB	8-bit family code MSB LSB

TABLE III
RETRIEVAL INFORMATION BIT

Bit (actual value)	Bit (complementary code)	Conclusions
0	0	The current bits in the ROM code of a slave include 0 and 1; differences exist
0	1	All the current bits in the ROM code of the slave are 0.
1	0	All the current bits in the ROM code of the slave are 1.
1	1	There is no response of a slave device on the bus.

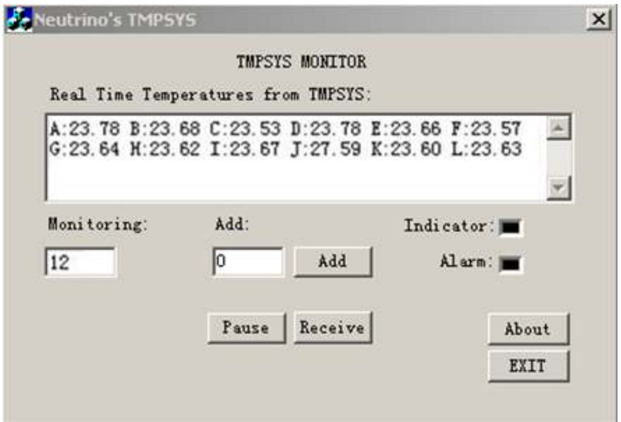

Fig. 5. Interface of the monitoring program on the PC side.

controlling a servo system for the terminal machine. An MSComm control is the core of the monitoring program.

F. Real-time test

The selected test environment is Keil uVision (v3.53), a logic analyzer carried in a Keil is used for timing test, the selected operating frequency of the C8051F020 is 25 MHz [14], and the selected switching frequency of the μC/OS-II (v2.00) is 50 Hz. The time difference is calculated from a tab file exported from the logic analyzer, that is, the response time is obtained. Refer to Table IV for the calculated evaluated data.

IV. Conclusions

Based on the above test, the kernel μC/OS-II can effectively realize multi-task scheduling with the support of a CIP-51 framework, and can accomplish the system function and can give a response within a short period time which is mainly determined by both the selected switching frequency of the μC/OS-II such as 50 Hz and the performance of CPU. Compared with the traditional back-fore ground system, the hard real-time kernel can enable the C8051F series single-chip microcomputer to realize more complex functions, and the subsequent upgrading of the application program does not affect the real-time property of the system. The sensor is simple and reliable in physical connection, convenient for expansion and maintenance of a bus, and particularly suitable for monitoring multipoint temperature; the keyboard and LCD are used as the man-machine interface, thus the core control chip with a JTAG interface provides a great convenience for detection and maintenance of the hardware system.

TABLE IV
System Response Test of the TMPSYS

Test items [b]	Switch tasks	Ip	Sas	Smm	Smq	Mpa	Mpl
Response time (us)	79	3	216	170	167	33	29

[b] Ip = Interruption preempt , Sas = Signal amount synchronization , Smm = Sending short messages through mailbox , Smq = Sending messages in queue , Mpa = Memory pool allocation , Mpl = Memory pool deal location.

References

[1] LU Sheng-jie, HUO Shu-yan. Single Bus Multi-point Temperature Detection Technology Based on DS1820[J]. Modern Electronics Technique,34(2):185-187, 2011.

[2] LI Gang, ZHAO Yan-feng. Principle and Application of 1-Wire Bus Digital Thermometer DS18B20[J]. Modern Electronics Technique,28(21):77-79,2005.

[3] ZHANG Yi-hui. Research on Synchronous Switching System of Air Preheated Based on C8051F [D]. Master Degree Candidate, Xi'an University of Technology, 2006.

[4] BI Jian-feng. The Research and Development of Intelligent Flowmeter Based C8051F060 [D]. Master Degree Candidate，Hebei University of Technology，2007.

[5] JU Rong, GUO Yi-qian. The application of DS18B20 in thermometric system [J]. Journal of Agricultural Mechanization Research,(1):224-226,2005

[6] Maxim Integrated Products.DS18B20 Programmable Resolution 1-Wire® Digital Thermometer. [EB/OL] . http://www.maxim-ic.com, 2013.

[7] ZHAO Jian-hua, WANG Wen-yong. Porting uC/OS-II kernel to 80C 51-family microprocessor [D]. Master Degree Candidate University of Electronic Science and technology, 2007.

[8] TANG Song-song. The Research of Integrated Development Environment Based on μC/OS-II for Intelligent Instrument Microprocessor [D]. Master Degree Candidate, Chongqing University, 2006.

[9] XU Jing-feng. Methods on Transplantation of μC/OS-II[J]. Computer Knowledge and Technology,4(29):493-494,2008.

[10] ZHAO Yue-qi, MA Rui-qing, LIANG Gui-yi, et al. Design and Realization of Intelligent Temperature Control System Based on C8051F Microcontrollers [J]. Computer Measurement & Control, 17(3):490-491,2009.

[11] Maxim Integrated Products. ±15kV ESD-Protected, 1μA, 3.0V to 5.5V, 250kbps, RS-232 Transceivers with Auto Shutdown [EB/OL]. http://www.maxim-ic.com, 2013.

[12] HUA Xian-zhe. Room temperature and humidity monitoring and energy-saving system [C].The 6th International Conference on Computer Science & Education. pages 537-540, 2011.

[13] Maxim.1-Wire Search Algorithm [EB/OL]. http://www.maxim-ic.com/app-notes/index.mvp/id/187 ,2013.

[14] LING Xu, ZHEN Chen, ZHANG Shi-lei et al. Research of serial communication system based on C8051F020 single chip[C]. 2010 International Conference on Computer Application and System Modeling. vol. 8 .2010 :V8-404-V8-407.