Design and Implementation of Multi-channel Data Acquisition Based on µC/OS-Ⅲ

Wang xiaokai, Guo gaizhi, Wang zhonghua

College of Computer and Information Engineering, Inner Mongolia Normal University
China

514693252@qq.com, ciecggz@imnu.edu.cn

Key words: µC/OS-Ⅲ, data acquisition, DSPIC30F6014

Abstract. This paper achieves task scheduling of the multi-channel data acquisition based on the operating system of µC/OS-Ⅲ It realizes transplanting the operating system of µC/OS-Ⅲ on DSPIC30F6014 to acquire the data of the voltage current temperature and power through)The experiment shows that it can acquire and process multi-data timely, then displays the information on the LCD .

Introduction

The multi-channel data acquisition is information processing module commonly used in the industrial control. The data it acquired is the parameter often used in industrial production. So the module of multi-channel data acquisition based on µ C/OS-Ⅲ in this paper is main functional module of the electrical regulator. It can control the conduction angle of SCR(silicon controlled rectifier) to regulate the voltage current temperature power and so on to reach the technical parameter in the industrial production according to the list of the parameter and multi-channel data timely. The processing method of multi-channel information is portable and can be used in every embedded device to acquire the multi-channel information with wide application prospect.

The hardware design

The description of dsPIC30F6014 chip. dsPIC30F6014 is the digital signal controller made by the world leading provider of microcontroller and analog semiconductor (Microchip).It uses high-performance modified RISC integrates with MCU and DSP seamlessly, and the CPU module uses 16-bit (data) and 24-bit(instruction) modified Harvard architecture with an enhanced instruction set including significant support for DSP. The dsPIC30F6014 has MCU and DSP double engine, so its instruction set has two classes of instructions: MCU and DSP. These two instruction classes are seamlessly integrated into a single and execute from one execution unit. The whole data space of processor is 64K, because the chip itself contains a DSP engine, and many DSP operations are executed in parallel, thus data space is split into two blocks, referred to as X and Y data memory. Each of the memory blocks have their own independent address generating unit (Address Generation Unit, AGU).The instructions of MCU class operates solely through the X memory AGU, which accesses the entire data space as one linear data space. Certain DSP instructions operate through the X and Y AGU to support read operation of dual operand at the same time, which splits the data space into two parts. dsPIC30F6014 also has abundant peripherals which allow devices to exchange the information with the outside world. The peripheral feature of dsPIC30F6014 is high-current sink/source I/O pins, five 16-bit timers/counters and 16-bit timers can be optionally configured as two 32-bit timers, 3-wire SPI modules, two addressable UART module with FIFO buffer and conversion rate with 200Kips, 12-Bit Analog-to-Digital Converters(A/D) with 16 input channels[1].

Configuration of A/D module. The dsPIC30F6014 A/D converter is convenient for acquisition data. The main job is the configuration relevant registers according to the sampling rate of acquisition signal and numerical format after quantified[2] .Follow the following steps to perform an A/D module.

1. Select source voltage to match the expected of analog input
2. Select the clock to match desired data rate with processor clock
3. Determine how sampling will happen
4. Determine how inputs will be allocated to the S/H channel
5. Determine how the conversion results will appear in the buffer
6. Select interrupt rate
7. Switch on A/D module

The software design

Device configuration. The user can use the device configuration register to customize certain aspects of the device. The device configuration registers are nonvolatile memory locations in the program memory. It can save dsPIC6014 configurations during power-off. The configuration registers save global configuration information of devices, such as the source of oscillator, the mode of watchdog timer, code protection and so on.

```
_FOSC(CSW_FSCM_OFF);              // turn off failsafe
_FWDT(WDT_OFF);                   // turn off   watchdog
_FBORPOR(PBOR_OFF);               //power-up timer disable
_FGS(CODE_PROT_OFF);          //user program memory isn't code-protected
```

Configuring A/D module. This system mainly uses AN3-AN6 channel to acquire voltage, current, temperature and power, then stores the data to ADCBUF and transfers these data to the microprocessor. The microchip displays the data on the LCD[3]

Perform an A/D conversion according to the following steps

```
ADCON2=0X042C;                    // V REFH = AVDD     VREFL = AVSS
ADCON3=0X000F;                    //Tad=8Tcy
ADCON1=0X0044;                    //compare end sampling using timer3
ADCSSL=0X00F0;                //scan input select from AN3, AN4, AN5, AN6
ADCON1bits.ADON=1;                //turn on A/D
```
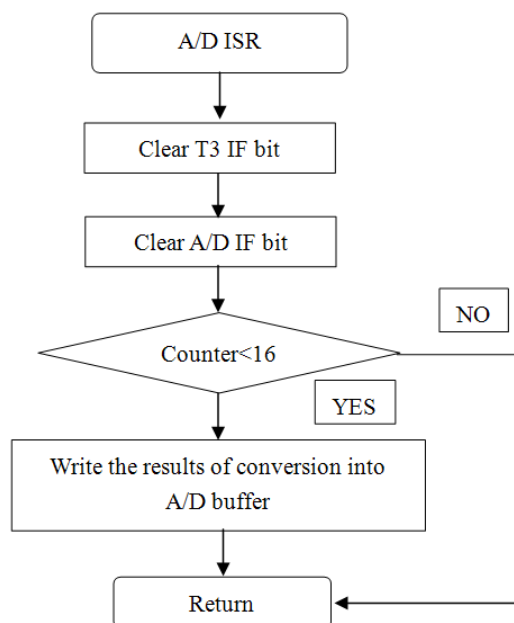

Fig 1 The interrupt service flow diagram

Porting µC/OS-Ⅲ. There are two main jobs of µC/OS-Ⅲ transplant, One thing is the design of the interrupt handler program, the other thing is the design of the time program .1. The design of the interrupt handler is an important part of hardware abstraction layer in the embedded operating system. ForµC/OS-Ⅲ, every interrupt handler must be written in assembly language. In addition handling the

interrupt is not the same in different compilers. The dsPIC compiler retains a interrupt vector for each interrupt. When the interrupt occurs it will automatically jump to the interrupt vector. The first step is saving the interrupt scene, if this is not interrupt nesting, then saves the pointer of the task stack, clears the interrupt flag status, calls the C interrupt service function, interrupt exits).if this is interrupt nesting, then not saves the pointer of the task stack, calls and processes the C interrupt service function. 2. The design of the system time. μC/OS-Ⅲ, like other computer operating systems, uses the hardware time to realize the system time with a millisecond time interrupt. The design uses timer 1 for system time. The step of initializing timer is that: 1 clear TM R1 bit2 setup Time1 interrupt for desired priority level 3 clear the Time1 interrupt status flag 4 enable Time1 interrupts 5 starts Time1 with prescaler settings and clock source set.2 select internal clock source. 3 set the frequency demultiplication, interrupt cycle (10 ms), the interrupt priority. 3 clear interrupt flag bit and enable interrupt. 4 start a timer[4].

μC/OS-Ⅲ architecture: (1) APP.C and APP.H are the main the application file. (2) *.h and *.c are library files provided by the manufacturer of CPU. (3) BSP.H and BSP.C are interface functions to peripherals in the aim circuit board. (4) OS_CFG_APP.C and other files are files that are processor-independent and μC/OS-Ⅲ. (5) OS_CPU.H and other files are files that are processor-dependent and make modification when you transplant μC/OS-Ⅲ. (6) CPU.H and other files summarize main function of CPU, for example disable and enable interrupts. (7) μC/LIB is a series of source that provide common functions such as the operation of the string.(8) 0S_CFG.H and OS_CFG_APP.H define mainly μC/OS-Ⅲ features such as idle task stack size ,tick rate, etc[5]
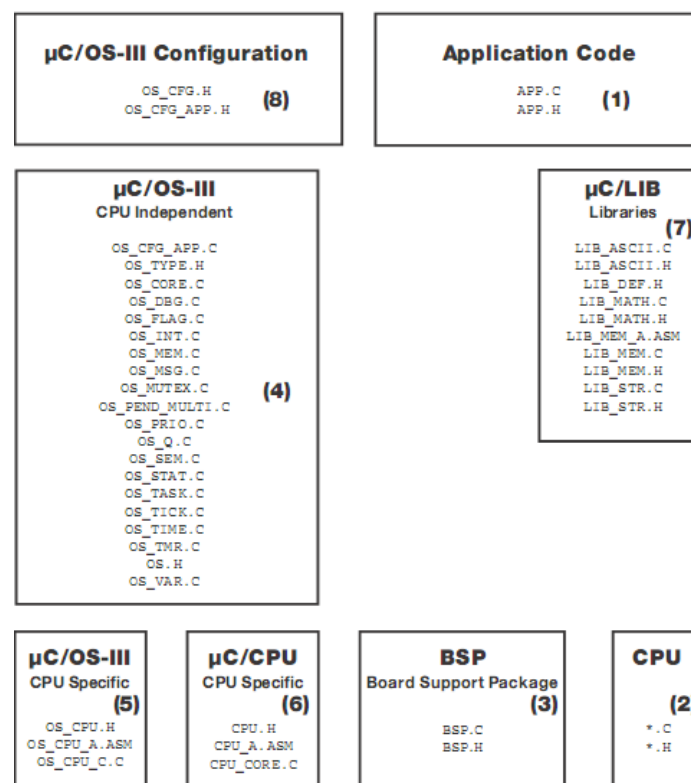

Fig 2 : μC/OS-Ⅲ architecture

Conclusions

The paper explains in detail the design and implementation of multi-channel data acquisition based on μC/OS-Ⅲ. The paper realizes 4-channel data acquisition with the dsPIC30F6014. The experiment shows that this system can acquire and process timely multi-channel data and schedule multi-task with μC/OS-Ⅲ operating system. Then it meets the need of the electrical regulator. This design has the very strong versatility, just to replace the system' microprocessor--- dsPIC30F6014A

by other processor chip, can be realize real time signal acquisition and storage processing through appropriate modifications in accordance with the method described in this paper, with a very broad application prospects.
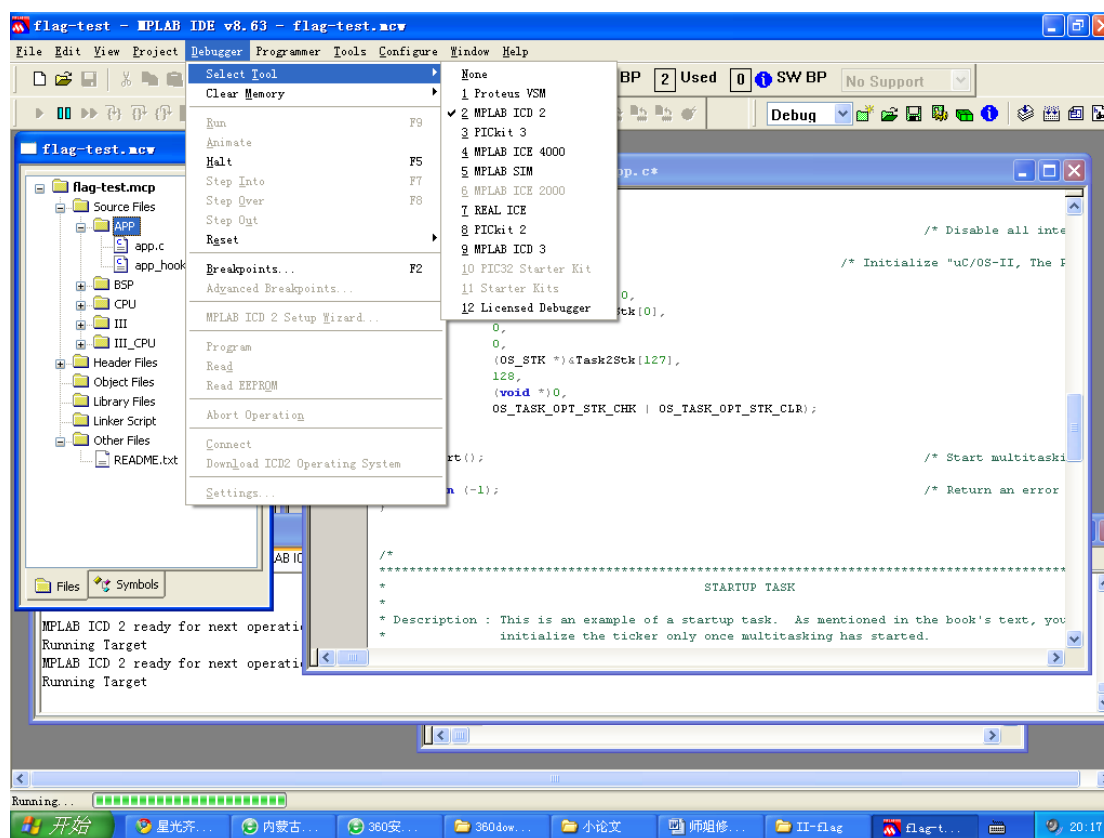

Figure 3 The running system

Acknowledgment

References

[1] Microchip Technology Inc．dsPIC30F Enhanced Flash 16-Bit Digital Signal Controllers General Purpose and Sensor Families Data Sheet［EB/OL］．www.microchip.com，2006．

[2] Wang fang.High-speed signal acquisition and processing system and its application. Northwestern polytechnic university .2006

[3] Wen xia Design and Implementation of the signal acquisition circuit

[4] information on www.micrium.com

[5] Jean J.Labrosse μC/OS-Ⅲ The Real-Time Kernel[M] Micriµm 2011