Design of the AC Motor Speed Regulator Controlled by ARM

Jing-Long Huang[a]   Zhong-zhong Peng[b] and Qing-Pan[c]

Hunan Provincial Key Laboratory of Health Maintenance for Mechanical Equipment, Hunan University of Science and Technology, Xiangtan, 411201, China

[a] xthjn@126.com, [b]beyond52176@163.com, [c]p710811@163.com

Keywords: ARM; u C/OS-II; AC motor; Vector Control

Abstract. The paper proposes an AC motor speed regulate system based on ARM. The system's hardware core is the LPC2131, in which processor successfully transplanted the µC/OS-II real-time operating system. In the form of task, each functional module of the system can be achieved on the processor LPC2131.The system is simple hardware circuit, software module to realize the complex control algorithms. Experiments show that the multi-task classify program is feasible on the embedded real-time operating system based on LPC2131 and µC / OS-II, it ensures the system's real-time, has good multi-task operating and transferring characteristics and improves the system stability and reliability.

Introduction

Digital technology as an AC induction motor control strategy is becoming more and more sophisticated in the high-performance control system. Real-time response speed and stability are important factors which determine the performance of AC motor speed regulation. Digital AC Frequency Control System is a typical embedded system, its digital features reflected in two aspects: first, all of input and output signals are digital, not analog signals; second, real-time multitasking operating system is introduced [1-3]. In the system, a variety of switching signals, such as manual switching signals, start and stop signals, require a very short response time, but for a variety of feedback, such as voltage and current, require being deal with quickly. There are two ways to improve the response speed, the one is use faster micro-controller, and another is use multi-task real-time embedded operating system. ARM both has the features of high running speed and rich peripheral functions, so it is suitable to the digital AC motor speed regulate system. µC / OS-II is an embedded real-time multitasking operating system, It can raise awareness of various on-site response speed of the signal, but also improve the reliability of the control software. Reasonable use of embedded real-time operating system has the following advantages [4]:

(1) CPU according to the priority of each task to dynamically switch the various tasks to ensure the real-time control requirements.

(2) CPU is used with highest efficiency; each task can be run in a timely manner.

(3) The software reliability of the control system will be greatly improved.

ARM-based system architecture

The AC motor speed regulation system can be divided into three parts: the main circuit section, the control section and driving section. The hardware block diagram of the system is shown in Fig.1. Among them, the system main circuit section is a classic AC-DC-AC inverter circuit which is composed of the rectifier module, IGBT inverter module and other components. The control section

is all-digital design and its core is ARM microcontroller. The control section is used to complete the functions of control algorithm, PWM pulse generation, the detection of electrical running signals, and communication with the host computer and soon.

LPC2131 is a 32-bit RISC processor with ARM7 TDMI kernel, its highest frequency can reach up to 60MIPS. LPC2131 has the advantages of the DSP and the 8-bit microcontroller, both of them have the features of high running speed and rich peripheral functions, 128-bit wide memory interface and unique accelerator architecture enable 32-bit code execution at maximum clock rate. It has all of the necessary functions in the situation of AC motor speed regulation system, including: 47 GPIO provides a convenient links to external LCD to display system parameters or links to external keyboard

Fig.1 Block diagram of system hardware

to set the operating parameters; 8 10-bit A / D converter, conversion time is only 2.44μS, can be used to receive voltage and current feedback signal;two industry-standard 16C550 UART, 2 high-speed I2C interface as well as SPI interface can connect to external PC and other devices when needed.

ARM Embedded System output 6-channel PWM pulse firstly, then these pulse through optical isolation in which interference is minimized into the driver section, finally through the inverter module apply to the switching elements. The test section, through conversion and isolation, transport the speed, current, and voltage back to the system through a certain algorithm, the chronological relationship between the outputs of a certain duty cycle of the PWM pulse, to achieve frequency control. To ensure system security, there is also equipped with a set for overvoltage, overcurrent, over temperature and power failure detection.

System software

Software overall plan. μ C / OS-II is a complete, portable, curing, cutting of preemptive real-time multitasking kernel, it has the advantages of small kernel, high real-time performance, open source etc. so that it is especially suitable for the projects which call for strong real-time and reliability.

In this paper, μ C/OS-II is successful transplanted to LPC2131 processor and then the μ C/OS-II is expanded. Expanded work has three main parts: firstly, writing drivers for the external device; secondly, writing application program interface function; thirdly, tasks division.

When tasks have been divided, it is necessary to try to make the task modules with high cohesion, low coupling between different modules. According to the system design requirements, the following five tasks modules are established:

(1) Message processing task: it accepts the messages from other modules and makes appropriate responses.

(2) Communication task: it accepts AC motor control parameters and command.

(3) Measuring task: it can be taken to deal with external interrupt;

(4) PWM task: it is responsible for the PWM output by the PI controller;

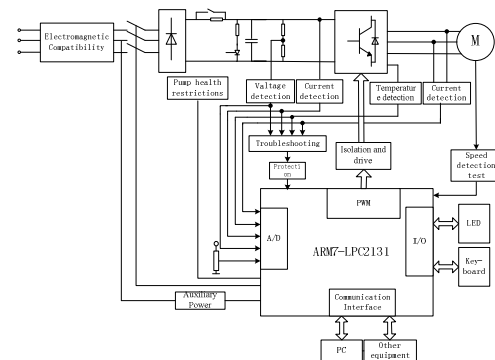(5) Man-machine dialogue task: it is used to display the current operating condition and the parameters.

More than five tasks, plus the system's own idle task, a total of six tasks, the message processing task has the highest priority while the idle task has the lowest priority. Information processing task is the system's core mission, which according to the current information, decided which task will run or suspend or resume. The block diagram of system software after μ C/ OS-II real-time kernel expanding is shown in Fig.2.

Vector control strategy. The basic idea of vector control is modeled AC motor as DC motor [5, 6]. The specific steps include:

Fig.2 Block diagram of system software

(1) Current detection circuit measures the stator current and use the Clarke and Park transformation transform the current: i_A, i_B, i_C into rotating coordinate system's DC component: i_M, i_T, then, i_M, i_T is used as negative feedback signals in the current loop;

(2) Using the speed detection circuit to measure the motor speed, then, the speed and deviation between speed and given speed are used as the input of PI regulator, finally, the speed PI regulator output current control for torque reference T-axis component ;

(3) Input the DC component i_T, i_M and the speed to the rotor flux position calculation module to find the rotor flux position;

(4) Input the deviation between the T-axis reference current component i_{Tref} and the feedback current i_T and i_M to the current PI regulator, respectively, output the phase voltage components: V_{Mref} and V_{Tref} in rotating coordinate system;

(5) Phase voltage component: V_{Mref} and V_{Tref} ,inverse into the stator phase voltage vector: $V_{s\alpha ref}$ and $V_{s\beta ref}$;

(6) According to $V_{s\alpha ref}$ and $V_{s\beta ref}$, the voltage space vector module generate the PWM pulse to control the inverter.

The C program of Clark transforms based on ARM as follows:

```
Void clark_calc(clarke*v)
{v->alpha=Q_mpy(v->as,_Q(1.22474487139159);//sqrt(3/2)=1.22474487139159
v->beta=Q_mpy(v->as+2*v->bs,_Q(0.70710678118654);// 1/sqrt(2)= 0.70710678118654
}
```

ARM-based Park transforms C program main code is as follows:

```
Extern float32 sin_tab[];
Void park_calc(park *v)
{if(v->angle+0.25>1.0)
{sin=sin_tab[(int16)(v->angle*256)];
cos=sin_tab[(int16)(v->angle*256)-192];}
Else
{sin=sin_tab[(int16)(v->angle*256)];
cos=sin_tab[(int16)(v->angle*256)+64];}
v->ds=Q_mpy(v->alpha,_Q(cos)+Q_mpy(v->beta,_Q(sin));
v->qs=Q_mpy(v->beta,_Q(cos)-Q_mpy(v->alpha,_Q(sin));
}
```
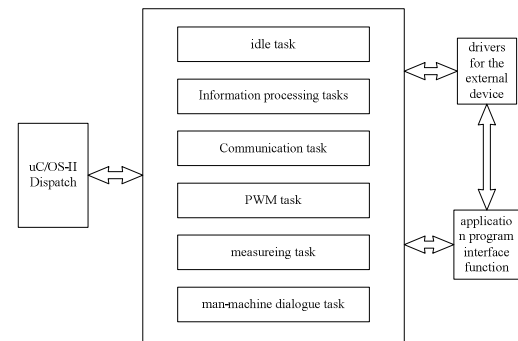
PID regulator. In the PID algorithm, the scale factor's role is to accelerate the system's response in the case of static difference, and helps to reduce the static error. However, too large scale factor causes a greater overshoot and oscillation. The integral's role in the performance of the "cumulative effect" is elimination static error. As long as the deviation is not zero, it can control the amount of cumulative effects and reduce the bias, it can guarantee the system in stable. Differential's role is helping reduce the overshoot, overcome the oscillation, but it is sensitive to disturbance.

PID controller's transfer function is:

$$G(s) = \frac{X_0(S)}{X_i(S)} = K_p(1 + \frac{1}{T_i s} + T_d s) \tag{1}$$

Where, K_p is proportional coefficient, T_i is integral time constant, T_d is derivative time constant.

The proportional, integral, differential combination of three kinds of laws regulating can achieve fast, smooth and accurate adjustment effect. The ideal PID control formula in time domain can be obtained when equation (1) After Laplace transformation:

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) + T_d \frac{de(t)}{dt} \right] + u_0 \tag{2}$$

In order to implement PID control on the computer, it is necessary to discrete Eq.2, replaced the integral with the addition, the differential with the increments. Let the sampling period T, PID control formula can expression in discrete form:

$$u_k = K_p \left[e_k + \frac{T}{T_i} \sum_{i=0}^{k} e_i + \frac{T_d}{T}(e_k - e_{k-1}) \right] + u_0 \tag{3}$$

Similarly, it is easy to lists the control output at the (k-1)th sampling time:

$$u_{k-1} = K_p \left[e_{k-1} + \frac{T}{T_i} \sum_{i=0}^{k-1} e_i + \frac{T_d}{T}(e_{k-1} - e_{k-2}) \right] + u_0 \tag{4}$$

$$\Delta u_k = u_k - u_{k-1} = K_p(e_{k-}e_{k-1}) + K_i e_k + K_d(e_k - 2e_{k-1} + e_{k-2}) \tag{5}$$

Obviously, Δu_k is the control incremental, Eq.5 also called as incremental PID formula, Eq.3 often referred to as location-PID formula. In the position control algorithm, it is need to give the initial value of the system, and output are related to each state of the past, the calculation should be carried out continuously accumulate, it is difficult to program design. The incremental design and this deviation is only about, nothing to do with the initial value, and thus easy to implement. But in the actual control system, the need to control is the amount of absolute, rather than the incremental. Thus in the course of an incremental basis, the output position control algorithm is used.

Undoubtedly, the derivative term in some cases can improve the dynamic quality, but its operations and parameter tuning is more complex. So in industrial control, PID control is more often used in practical applications, namely:

$$u_k = u_{k-1} + K_p(e_{k-}e_{k-1}) + K_i e_k$$
$$= u_{k-1} + K_p(1 + \frac{T}{T_i})e_k - K_p e_{k-1} = u_{k-1} + \alpha \cdot e_k + \beta \cdot e_{k-1} \tag{6}$$

Speed regulation subroutine flow chart shown in Fig.3.
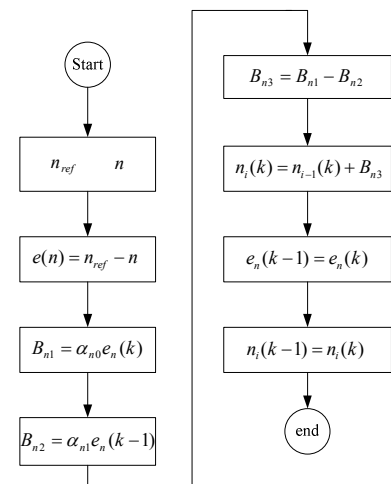
Fig.3 Flow chart of rotate controlling sub program

Experiments

The AC motor speed regulates system to be tested. Fig.4 is the output waveform at the same bridge arm. The measured value of dead time between the same bridges is 5μs, the measured value meets design requirements exactly. In addition, can be seen from the figure, the PWM frequency is 5 KHz, it is consistent with frequency inverter design.

Fig.5 is the power output of the phase line voltage waveform UAB, we can see, UAB is a series of pulse shape; a whole also reflects the variation of the sine function. It can be seen, the AC motor speed regulate system designed to meet the design requirements, to produce the correct waveform and the driving power to drive the main circuit to work properly.
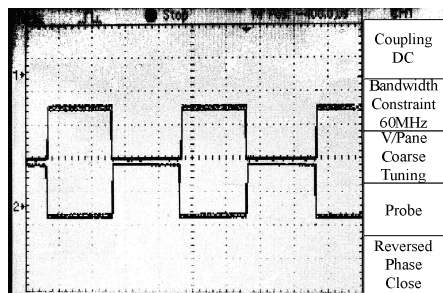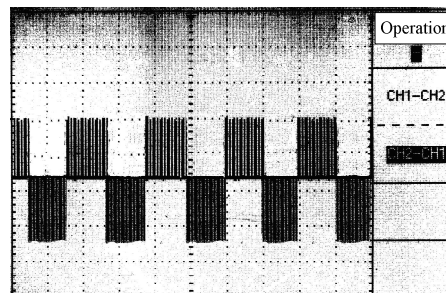

Fig.4 PWM waveform from ARM   Fig.5 Waveform of line voltage between A phase and B phase

Therefore, the results show that IGBT driver circuit is working normally. As it can be seen from the above results, PWM control output task program design is feasible, the programming is correct.

Conclusions

(1) The structure of the system is simple and reliable.

(2) Based on the digital embedded system developing plan, the system's Characteristics of real-time and accuracy is improved.

(3) Tests show that the LPC2131-based vector control system of induction motor is feasible.

Acknowledgement

References

[1] K. Johnson: Mining Journal Vol. 334(2000), p.268.

[2] Mills AP Jr: Mining Technology Vol. 77(1995), p.167-173.

[3] J.G. Jiang, Y.L. Zhao, Z.M. Shao. World Mining Equipment Vol. 1(2001), p.1-6.

[4] Beus, M.J., Ruff, T.M., McCoy, W.G. Industry Application Conference,1997.Thirty-Second IAS Annual Meeting,IAS'97,Conference Record of the 1997 IEEE Vol.3(1997), p.2091-2097.

[5] Z. Z. Liang: The 18th International Conference on Computers & Industrial Engineering (ICC & IE'95), Shanghai, China Vol.2(1995), p.799-802.

[6] Johansson: Bergsmannenmed Jernkontorets Annaler (2000),p.41-42.