ISSN: 2278 – 7798
International Journal of Science, Engineering and Technology Research (IJSETR)
Volume 2, Issue 7, July 2013

Based on the combination of advanced RISC machines and ARM-Linux development of ARM based embedded system for robot applications

[1]Vaishak N L, [2]C.G.Ramachandra, [3]Shilpa.B

***Abstract*— in this project, we introduce the configuration of embedded system, and then present a robot control system based on an embedded operating system and Advanced RISC Machines. Based on the combination of advanced RISC machines and ARM-Linux, this project involves development of embedded robot control systems for industrial applications. The design of embedded control system for robot includes four main aspects, i.e., system structure, functions, hardware, and software design. In the development of this system, some features are included such as hierarchy structure, modular hardware, and structured software, to make the system suitable for a variety of robots applications through some hardware adjustment and software customization only. The effectiveness of proposed approach has been verified by a straight line Motion Demonstration of a 6-DOF series manipulator. The effectiveness of proposed approach has to be verified and tested. Here, we are using Quasar Technology for GUI application.**

***Key words*—ARM Linux, Embedded system, RISC, DSP**

I. INTRODUCTION

With the development of the engineering science and technology, the utilization of automation or automatic devices will be significantly maximized in a number of fields, such as from small engineering operations to large and heavy operations like ocean resource exploitation [6], etc. However, it is least bothered for what purpose for utilizing the automatic devices here we consider robots, almost all of them are made up of two parts, the mechanical body and the control system. The mechanical body not only represents the basic characteristic of a robot [2], but also determines that it is an unmanned system. Dealing with tasks in dangerous and complex environments, the robot should be entitled the abilities of thinking and making decisions to some extents, moreover, it needs multi motors work together co-ordinately for the motion control of a robot[6]. When considering all of this demand complex algorithms including motion control algorithm and pattern recognition algorithm. As far as we know, it is difficult for robots we discussed above to carry a computer with them as a motion control system, at the same time, it is also very difficult for the robot control system with only low-performance microprocessors to deal with so many complex calculations [10]. Fortunately, it is just a possible solution to overcome these difficulties through developing embedded systems and 32-bit microprocessors [4].

An embedded system is a special-purpose computer system, which is completely encapsulated by the device it controls, so there are some specific requirements for each system, such as functions, reliability, cost, size, and power consumption [13], etc. Based on the computer technology, an embedded system is designed for specific-application with hardware and software that could be tailored to adapt the system requirements. As the core device of an embedded system, the embedded microprocessor can be an 8-,16- or 32-bit microprocessor[8]. Because of the limitation from the lower performance such as low running speed, low addressing capability and high power consumption, etc, the 8- or 16-bit microprocessors impossible to meet the needs of some complex embedded applications. In the field of the 32- bit embedded system application [11], ARM (Advanced RISC Machine) gains extraordinary success. Generally speaking, the ARM-kernel microprocessors present as smaller size, lesser utilization of power, lesser price, and relatively higher performance[7], etc. For instance, with a maximum number of register and high instruction executing speed, many of data operations are completed in registers [2].

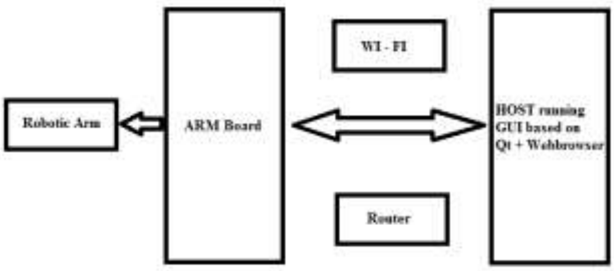

Figure 1. Functional Diagram

II. SYSTEM REQUIREMENTS SPECIFICATION

A system requirements specification (SRS) is a complete description of the behavior of the system to be developed. It includes the functional and non functional

***Vaishak N L** [1], Dept. of Mechanical Engineering, VTU/ S.C.E.M Mangalore. Mangalore, India,Mob No: +919902537523*
***C.G.Ramachandra**[2]Dept.of Mechanical Engineering VTU/ S.I.T Mangalore. Mangalore, India,Mob No: +919886976352*
***Shilpa B**[3], Dept. of Information Science Engineering, VTU/ C.C.E.M Benjanapadav Mangalore Mangalore, India,*
Mob No:+919449745266

requirement for the software to be developed. The functional requirement includes what the software (model) should do and non functional requirement include the constraint on the design or assumptions made during the course of the project. Requirements must be measurable, testable, related to identified needs or opportunities, and defined to a level of detail sufficient for the system design.

Software requirements specification will contain states what the software will do. What the software has to do is directly perceived by its users- either human user or other software systems. The common understanding between the user and developer is captured in requirements document. The writing of software requirements specification reduces the effort, as careful review of the document can reveal omissions, understandings, and inconsistencies early in the development life cycle.

A. Software Requirements

This section describes software requirements to our project. These are as follows:

- Programming language: C
- Linux kernel version 2.6
- GTK (GIMP Toolkit)
- GDB (GNU Debugger)
- Qt creator

B. Hardware Requirements

This section defines hardware requirement that are to be supported by the software, including logical structure, physical addresses, expected behavior, etc. The minimum hardware requirements are as follows:

- PC(Intel Pentium Processor with more than 400 MHz)
- S3C2440 Embedded Board
- Ethernet Cable
- Serial Cable
- USB cable
- USB network adapter
- Wireless Router
- Robotic Arm

III. SYSTEM ANALYSIS

A. Existing System

With the development of the science and technology, the application of robots will be significantly increased in a number of fields, such as

- Outer space exploration,
- Ocean resource exploitation, etc.

However, no matter what purpose for utilizing the robots, almost all of them are made up of two parts,

- The mechanical body and
- The control system.

The mechanical body not only represents the basic characteristic of a robot, but also determines that it is an unmanned system. Dealing with tasks in dangerous and complex environments, the robot should be entitled the abilities of thinking and making decisions to some extents, moreover, it needs multi motors work together co-coordinately for the motion control of a robot.

B. Problems in Existing System

Because of the limitation from the lower performance such as low running speed, low addressing capability and high power consumption, etc, the 8- or 16-bit microprocessors cannot meet the requirements of some complex embedded applications. Complex algorithms including motion control algorithm and pattern recognition algorithm. It is also impossible for the robot control system with only low-performance microprocessors to deal with so many complex calculations.

C. Proposed System

It is just a possible solution to eliminates the difficulties occur in existing system, through Developing embedded system, 32-bit microprocessors.

An embedded system is a special-purpose computer system, which is completely encapsulated by the device it controls, so there are some specific requirements for each system, such as functions, reliability, cost, size, and power consumption, etc. Based on the computer technology, an embedded system is designed for specific-application with hardware and software that could be tailored to adapt the system requirements. As the core device of an embedded system, the embedded microprocessor can be an 8-,16- or 32-bit microprocessor.

D. ADVANTAGES OF PROPOSED SYSTEM

To overcome the limitation occurs in the existing system, a trust based technique is implemented. In the field of the 32-bit embedded system application, ARM (Advanced RISC Machine) gains tremendous success. The ARM-kernel microprocessors present as smaller size, lower power consumption, lower cost, and relatively higher performance, etc. For instance, with a plenty of register and high instruction executing speed, most of data operations are completed in registers.

III. DESIGN

The main purpose of this design is to meet the requirements of controlling the multi- robots system, meanwhile, to pay attention to some characteristics, such as: size, weight and power consumption, etc. to which the robot is sensitive. The design of embedded control system includes four aspects, i.e., system structure, functions, hardware, and software design.

The embedded robot control system designed in this paper is applied to a 6-DOF serial robot and performs fine characteristics of reliability, real-time and general-purpose capability. Moreover its small size and low power consumption fulfill the requirements. Without computer, the control system can carries out the control arithmetic on the server controller (ARM) and control the client controllers (DSP) through bus. Thus the robot is entitled the ability of making decision independently to some extent. So far, there are some shortcomings in this developed embedded control system as well, such as the design of control arithmetic and improvement of functions, on which further study should be conducted. With the development of the science and technology, the application of robots will be significantly increased in a number of fields, such as outer space exploration, ocean resource exploitation, etc.

A. Hardware design

The server controller is suggested to adopt S3C2410 supplied by Samsung as its CPU. The S3C2410 is a 16/32-bit, 266MHz, low power consumption, and high performance RISC microprocessor with ARM920T as its kernel, which is particularly suitable for real-time control. Meanwhile, it supports Window CE Palm OS Symbian OS, Linux and real-time operating system, etc.

Moreover, S3C2410 supplies abundant equipment inside to bring down the cost of the whole system and there is no need of collocating excess equipment any more. The function of integrated circuit include: separate 16KB instruction Cache and 16KB data Cache, MMU virtual memory management, LCD controller, supporting NAND Flash system induction, outside memory controller, three-channels UART, four channels DMA, four-channels PWM timer, 117 currency I/O interface, 24-channels outer interrupt recourse, RTC with calendar, 8-channels 10bit ADC and touch screen interface, IIC-BUS interface, IIS-BUS interface, USB mainframe, USB equipment, two-channels SPI, and interior PLL clock times frequent count. The peripheral inside the chip is connected with the bus inside of the chip as well.

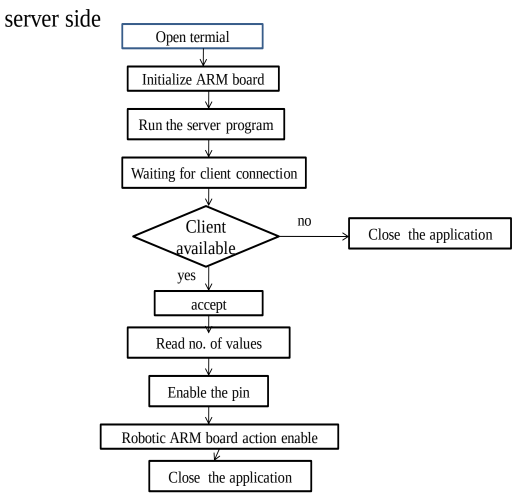

Figure 2. Flow chart of connection establishment at server side

The bus inside chip of S3C2410 adopts advanced microprocessor bus architecture (AMBA). AMBA includes advanced high-performance bus (AHB), advanced system bus (ASB), and advanced peripheral bus (APB). Both AHB and ASB are all high performance bus designed with larger bandwidth, and supporting multi-master equipment and multislave equipment. Actually APB is a secondary bus providing AHB/ASB with a simply extending as low power consumption and interface. APB is connected with AHB or ASB through the bridge. The architecture of S3C2410 is AHB+APB, as shown in Figure 3; it is the interior architecture of S3C2410.

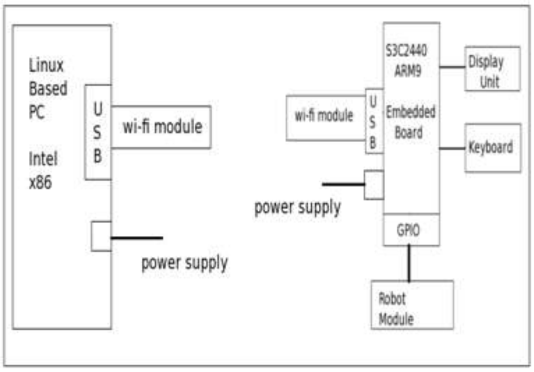

Figure 3: Architecture of hardware design

B. Software Design

The operating system is one of the most important parts in the whole system. It is in charge of managing all available resource and distributing them among different tasks in order. According to different resource distribution strategies, there are two kinds of the operating system: one is Real-Time Operating System; the other is Time-Sharing Operating System.

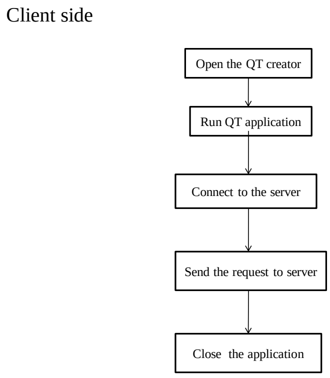

Figure.4. F low chart of connection establishment at server side

The operating system is one of the most important parts in the whole system. It is in charge of managing all available resource and distributing them among different tasks in order. According to different resource distribution strategies, there are two kinds of the operating system: one is Real-Time Operating System; the other is Time-Sharing Operating System.

This system software is based on the RT Linux which is Real-Time and multi-tasks operating system. RT Linux is changed from the kernel of Linux by adding RT Linux kernel control between the process of Linux and hardware interrupt. It carries out a virtual interrupt mechanism in the kernel of RT Linux.

Figure 5. Qt software

C. IMPLEMENTATION

Initializing ARM board and GPIOG pins and establishing the connection

To initialize ARM board first have to address which is IPV4 address which is of 32 bit and this is the standard address that we should define first. And also we should define the port number.

```
#define ADDR "192.168.1.230"
#define PORT 3069
```

Then from 34 GPIO pins in this project, only 9 GPIOG pins have been used. Particularly for GPIOG pins one should define

```
#define GPIOG_CON 0x56000060
#define GPIOG_DAT 0x56000064
```

Then calculated the virtual address that address will be calculated by using the pin numbers, those numbers are whenever that GPIO pin is enable value will be ‘01’, if it is disable then that value will be ‘00’.By using these criteria that address of GPIOG will become 0X155555.

To click on open button, we have used c language , in if it returns value 1 it will send the request to server then that robot will show the action ,if it returns negative value then server will not show the action

❖ **Function OPEN.**

```
if(checked)

{
ui->status->setText("Opening");
emit sendout('1');
}
else
emit sendout('s');
}
```

To click on close button, we have used c language , in if it returns value 2 it will send the request to server then that robot will show the action ,if it returns negative value then server will not show the action.

❖ **Fuction to CLOSE**

```
if(checked)
{
    ui->status->setText("Opening");
    emit sendout('2');
}
else
    emit sendout('s');
}
```

D.TESTING

Testing performs a very critical role for quality assurance and for ensuring the reliability of the software. The aim of testing is to find the maximum possible number of errors. To state simply, the aim of testing is to identify maximum number of errors with a, minimum amount of effort and realistic effort and realistic time period. Testing is the phase where the errors from the previous phases are detected. It is time consuming and perhaps one of the most important phases of Software Development.

Test Plan

Each and every module should be tested for correctness and completeness. The system is to be tested for functional requirements that were to be satisfied in order to satisfy the proper functioning of the system. Also the interactions of the modules with other modules are properly tested. Test cases are designed before starting to code, and their outcome is found out manually. The mismatch of outcome of these test cases and the corresponding manual results depicts an error. For an error-free program, the developer or tester would like to determine all the test cases. It is very difficult to test with all possible test cases due to cost and effort needed to generate a test case, time consumed to execute the program. Hence, more number of errors has to be detected with minimum number of test cases.

Table 7.1 Testing and validation of Robot Actions

MODULE1	GIVEN INPUT	MODULE2	GIVEN INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT	REMARKS
Sender(QT)	Ip address	Receiver(GPIO)	Ip address	Connection must be established	Conection must be established	Tested ok
Sender (QT)	Click on open button	Receiver(GPIO)	Enable pin gpiog0	Gripper Open	Gripper open	Tested ok
Sender (QT)	Click o n close button	Receiver(GPIO)	Enable pin gpiog1	Motor gripper close	Motor gripper close	Tested ok
Sender(QT)	Click on that up1 button	Receiver(GPIO)	Enable pin gpiog3	Motor wrist up1	Motor wrist up	Tested ok
Sender (QT)	Click on that down1	Receiver(GPIO)	Enable pin gpiog5	Motor wrist down	Motor wrist down	Tested ok
Sender (QT)	Click on that up2	Receiver(GPIO)	Enable pin gpiog6	Motor ankle up	Motor ankle up	Tested ok
Sender (QT)	Click on that down2	Receiver(GPIO)	Enable pin gpiog7	Motor ankle down	Motor ankle down	Tested ok

IV. CONCLUSION

The embedded robot control system designed in this paper is applied to a 6-DOF serial robot and performs fine characteristics of reliability, real-time and general-purpose capability. Moreover its small size and low power consumption fulfill the requirements. Without computer, the control system can carries out the control arithmetic on the server controller (ARM) and control the client controllers (DSP) through bus. Thus the robot is entitled the ability of making decision independently to some extent.

So far, there are some shortcomings in this developed embedded control system as well, such as the design of control arithmetic and improvement of functions, on which further study should be conducted.

REFERENCES

1. R. M. Murray, Z. X. Li, S. S. Sastry, “A Mathematical Introduction to Robotic Manipulation”, America: CRC Press, 1994.
2. W. H. Chen, G. L. Yang, Edwin H. L. Ho, and I. M. Chen, “Interactive Motion Control of Modular Reconfigurable Manipulators”, Proceeding of the 2003 IEEE/RSJ Inter. Conference on Intelligent Robots and Systems Las Vegas, Nevada. October 2003.
3. L. Romdhance, “Design and analysis of a hybrid serial parallel manipulator”, Mechanism and Machine Theory, 1999, Vol. 34: 1037-1055.
4. K. Yaghmour, “Building Embedded Linux Systems”, America, O’Reilly, 2004.
5. Michael Barr, “Programming Embedded Systems in C and C++” , America: O’Reilly, 2001
6. J. Catsoulis. ”Designing Embedded Hardware”. America: O’Reilly, 2001
7. L. P. Yan, H. Zeng, K. Song, “Analysis and Design for Embedded Intelligent Family Gateways”, Control and Automation, 2005, 2: 14-16.
8. T. Z. Sun, W. J. Yuan, and H. F. Zhang, “Manual for Embedded Design and Linux Driver Development: Based on ARM9 Microprocessors”, Beijing: Publishing House of Electronics Industry, 2005
9. Y. Chen, M. Li, Y. Yang. “Analysis and Practice of Open-Code Embedded System Software: Based on SKYEYE and ARM Development Platform”, Beijing: Beijing University of Aeronautics and Astronautics Press, 2003
10. L. G. Zhou, Foundation and Practice for ARM Microprocessors, Beijing: Beijing University of Aeronautics and Astronautics Press, 2003
11. J. G. Li, X. Y. Nie, and Z. M. Jiang, “Development Specifications for ARM Application Systems”, Beijing: Tsinghua University Press, 2003
12. H. P., Liu, W. J Wang, and Y. Jiang, “Application and Development on C Language for TMS320LF240x DSP”, Beijing: Beijing University of Aeronautics and Astronautics Press, 2003
13. S. Wang, J. J. Li, and J. H. Liang, “Embedded Autopilot Design of Micro-Robots”, Journal of Beijing University of Aeronautics and Astronautics, 2005, 31(7): 775-779

AUTHOR'S PROFILE

Vaishak N L is working in Sahyadri College of Engineering and Management as Assistant Professor in the department of Mechanical Engineering. He received the M-Tech degree in Industrial Automation and Robotics in Srinivas Institute of Technology, Mangalore from Visvesvaraya Technological University and B.E degree in Vivekananda College of Engineering and Technology Puttur in 2008. He had presented 3 national conference and one international conference papers in the field of robotics. He also published one international journal paper in the field of Automation and Robotics. He is an active member of ISTE. His area of interest is total quality management in the preventive maintenance (in processing industry) and soft automation.

Ramachandra C G is working in Srinivas Institute of Technology as Associate Professor in the department of Mechanical Engineering. Presently he is pursuing Ph.D degree from Visvesvaraya Technological University. He received the M-Tech degree in Maintenance Engineering from Sri Jayachamarajendra college of Engineering, Mysore under Visvesvaraya Technological University in 2002 and B.E degree in Mechanical Engineering from P.E.S College of Engineering, Mandya under Mysore University in 1998. He had presented 10 national conference and 8 international conference papers in various fields. He has published 13 international journal and 3 national journal paper in various fields. He is an active member of ISTE and IEI. He has done the project in the area of condition monitoring, maintenance management information system etc. His area of interest is Maintenance Engineering, Management Information System, Condition Monitoring and Production Engineering.

SHILPA B is working in Canara Engineering College, Benjanapadavu as Assistant Professor in the department of Information Science and Engineering. She received the M-Tech degree in Industrial Automation and Robotics in Srinivas Institute of Technology, Mangalore from Visvesvaraya Technological University and B.E from Vivekananda College of Engineering and Technology Puttur in 2010 under Visvesvaraya Technological University. She had presented 3 national and international conference papers in various fields. She is an active member of ISTE. She has done the project in the area of Matlab.