TELKOMNIKA, Vol.11, No.11, November 2013, pp. 6706~6713
e-ISSN: 2087-278X

Design of a New Intelligent Controller with Switch Management

Guangfu Wang
Sichuan Electromechanical Institute of Vocation and Technology, Panzhihua, Sichuan, China, 617000
e-mail: wangguangfu126@126.com

Abstract

This paper has a study on a new intelligent Programmable Logic Controller based on industrial Ethernet (IPLCbIE). It proposes a complete solution for the target segment: MMM, WWW, F&B, Medium Hydro power. The scopes of the solution discussed in this paper are the process and field part of the IPLCbIE architecture. For the process part the scope of our design includes the controllers, the engineering software tools (Unity Pro) and the interfaces with the other equipment of this level, typically other controllers or tools like asset management, SCADA or OPC server. Safety and HSBY controllers are also part of our design. For the field part, the scope of our program includes the remote I/Os and the interfaces with the other equipment of this level, typically distributed I/Os & devices. In this paper, we make a study on the design of the controller. It includes the system architecture of IPLCbIE, the application of network module. And it focuses on the discussion of design for the kernel module of the PLC-network module NOC. Our design also includes the interfaces with the operation & management tools. The hardware design is especially introduced in detail in my paper.

Keywords: *PLC, industrial ethernet, switch management*

1. Introduction

Nowadays, more and more network applications, such as data distribution, distant education and distributed database, work on the Ethernet communication mode. For the future merge of networks, the computer network needs the capability to support the traditional industrial message broadcast business. So how to support the field bus communication based on Ethernet in the industrial network is the net researcher's important direction. Ethernet is the most common Intranet and access network. Previous Ethernet is a network shared by all hosts. It can't support the group communication [1-3].

The Intelligent PLC Controller based on Industrial Ethernet (IPLCbIE) is a new product design for the different module which mates with industrial PLC platforms respectively. These designs share a common hardware architecture which will support both the Modbus Protocol and the Control and Information Protocol (CIP) as specified by the ODVA [4].

The goal of the IPLCbIE is to provide a complete solution for the target segment: MMM, WWW, F&B, Medium Hydro power. IPLCBIE Control Platform program address the Process Control and the Discrete Control bricks of this offer.

It is a control colution aligned with the end user strategy, delivering value to our 5 targeted verticals, taking care of our installed base. It provides a mid/long term solution to substitute Premium & middle range platforms. Due to INTEL micro-processor obsolescence, it rationalizes the end user range portfolio, and takes advantage of renewal to reduce development & maintenance costs.

The scope of the solution we will discuss is the process and field part of the IPLCbIE architecture. Our design also includes the interfaces with the operation & management tools. For the process part the scope of our program includes the controllers, the engineering software tools (Unity Pro) and the interfaces with the other equipment of this level, typically other controllers or tools like asset management, SCADA or OPC server. Safety and HSBY controllers are also part of our design. For the field part, the scope of our program includes the remote I/Os and the interfaces with the other equipment of this level, typically distributed I/Os & devices. In this paper, we will make a study on the design of IPLCbIE. It includes the system architecture of IPLCbIE, the application of network module. And it focuses on discussion of the

Received May 20, 2013; Revised June 23, 2013; Accepted July 26, 2013

design of the most important module–network module NOC. The hardware design is introduced in detail in my paper.

2. IPLCbIE System Architecutre

The IPLCbIE system architecture is shown in Figure 1. The new design includes a series of components of PLC controller:

M5xx new CPU modules - A new range of CPUs will support the new features of the solution.

The CRP function will be integrated in CPU M5xx. The CID/CIM service will also available with CPU M5xx, but only for local rack [5-8].

Network modules - The new solution includes 3 new network modules: CTRL Factorycast NOC for Control manages the interface with the control network and provides routing capabilities between the control network and two other independent sub-networks. 1 Factorycast Generic NOC and 1 non-Factorycast Generic NOC for Distributed I/Os manage standard Ethernet devices and also provide Unity connectivity to CPU via Ethernet.

X80 I/O Drops adapters – This solution includes a new drop's adaptors: a new module BME CRA allows the connection on the device & I/Os Ethernet network of a drop including one or two racks of X80 modules. This CRA is compatible with Ethernet Backplane [9].

M5xx new EDRS module - This solution includes all the equipment and accessories needed to build the validated topologies of the Control network and the Device I/O network. EDRS is short for Integrated Dual Ring Switch. A new module BME EDRS allows: To support both DIO & RIO on the same physical network [10]; Connect a sub-ring of IO devices, in a Daisy Chain Loop, to the RIO Main Ring; Compatible with Ethernet rack.

Unity Pro V8.0 - A new release of Unity pro will provide all the necessary specific software tools to Design, Configure, Commission, Operate and maintain the Controller, its I/Os and the managed devices.

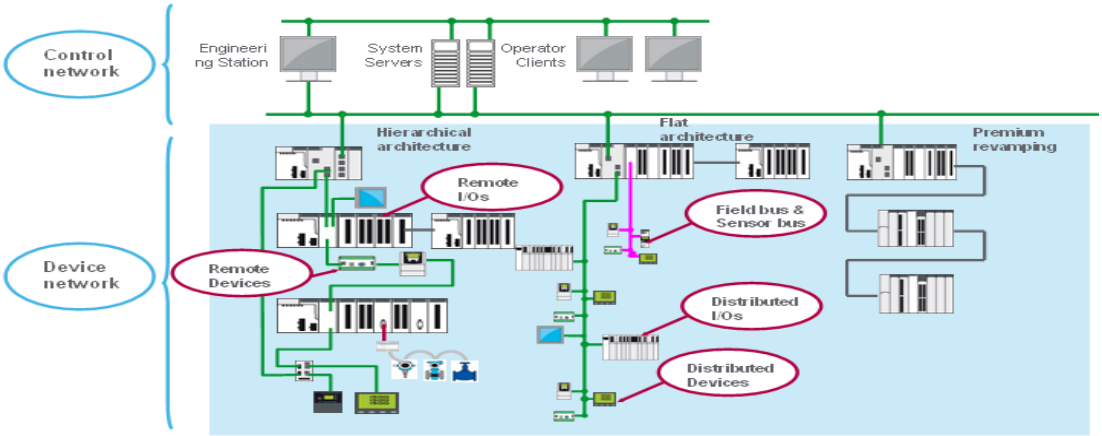

Figure 1. IPLCbIE System Architecture

3. Application of Network Module

The NOC control head module is mainly responsible for providing network transparency between devices located on a device network (including remote I/O and/or distributed I/O devices), an extended distributed I/O network, and a control network, while preserving determinism for remote I/O devices on the device network.

In our new network module, it operates in a redundant network that uses the RSTP protocol. It configures IP parameters and device configuration files for I/O devices. It supports Hot Standby functionality. It operates with other EIO head modules (CRP, NOC) or operates without being interlinked with these head modules on the local rack. Figure 2 is the connections example between devices. This example shows the maximum cable lengths between remote I/O and distributed I/O devices and a control network in an EIO installation [11].

In the Figure 2, 1 is CRP remote I/O head module, 2 is NOC distributed I/O head module interlinked with the CRP module, 3 is NOC control head module interlinked with the NOC module on the local rack. 4 is distributed I/O sub-ring. 5 is dual-ring switch (DRS) configured for copper-to-fiber and fiber-to-copper transition on the main ring (connecting the distributed I/O sub-ring and the distributed I/O cloud to the main ring). 6 is distributed I/O cloud, 7 is remote I/O drop on the main ring. 8 is remote I/O drops on the remote I/O sub-ring. 9 is DRS on the main ring (connecting the remote I/O sub-ring to the main ring). 10 is control network (connected by the 140 NOC 781 00 module). 11 is main ring.

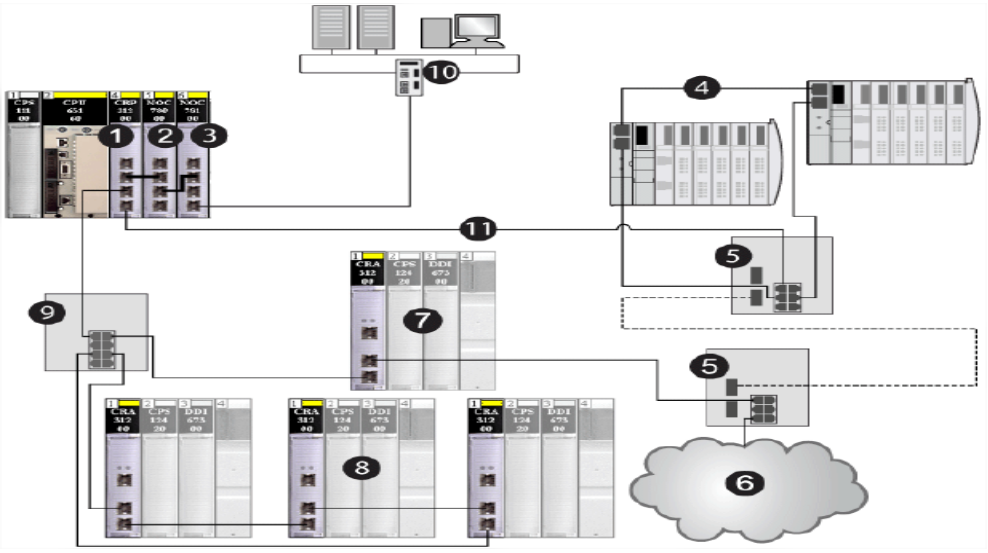

Figure 2. Connections of Network Modules

4. Hardware Design

The design uses the same platform as used by the Casper EtherNet modules with a few changes. Memory size has been increased to 32MB Flash and 32MB SDRAM and the addition of a Marvell 88E6165 multi-port EtherNet Switch. The modules has four 10/100 ports. The processor and memory architecture are schematically identical with each having its own unique backplane interface [12].

As shown in Figure 3, an MPC870 132Mhz processor manipulates data between the backplane, local memory and EtherNet Switch. On power up code stored in Flash is mirrored and executed from SDRAM providing a faster execution time. Ethernet traffic is exchanged between the 870 internal MAC and external EtherNet Switch with integrated PHY's.

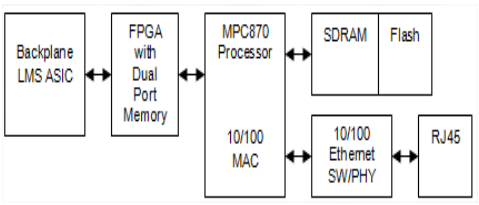

Figure 3. NOC Overview

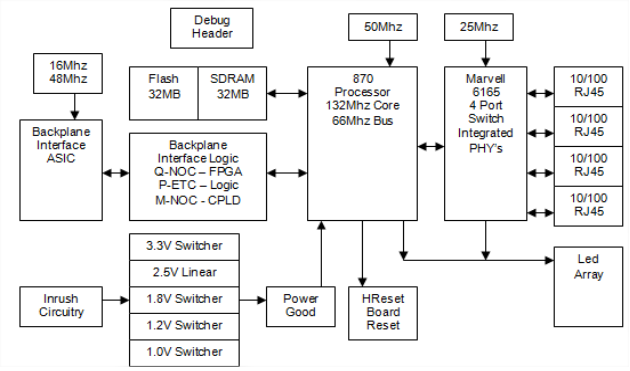

Figure 4. Ethernet Interface Design

The Ethernet design is the most important part of network module NOC [13]. The design is shown in Figure 4.

4.1. MPC870 Processor

Complete definition of this sub-system, component or function (Designed functions, Interface description, Operating mode with sequence diagrams, Software source code location, Material characteristics, BOM,….)

The processor is a 32bit Freescale MPC870 integrated communications controller with the core operating at 132MHz and the external bus operating in 2:1 mode or 66MHz. Embedded is 8KB of I-Cache and D-Cache, three memory controllers (GPMC/UPMA/UPMB), two 10/100 FEC controllers (Fast Ethernet Controller), one SMC (no Handshake) Serial Controller, JTAG controller, BDM (Background Debugger) and multiple programmable general purpose I/O. Core power supply is 1.8V with 3.3V I/O.

1) Clock Generation

Complete definition of this sub-system, component or function (Designed functions, Interface description, Operating mode with sequence diagrams, Software source code location, Material characteristics, BOM,….).

On power up the MODCLK [1:2] pins are sampled [1:0] using pull-up/pull-down resistors which configures the internal PLL in 1:1 mode using the external oscillator frequency of 50Mhz. Core and bus frequency remain at 50Mhz until the BootRom reconfigures the PLL for 132Mhz core with the bus operating in 2:1 mode or 66Mhz.

2) Memory (Flash & SDRAM)

BootRom and Executive code is stored in 16bit wide (32MB) Flash using Spansion MirrorBit Flash. On power up the code image in Flash is uncompressed into 32bit wide (32MB) SDRAM providing faster execution times. Options have been provided to support larger memory densities.

3) Memory Controller & Wait States

Three memory controllers GPMC, UPMA, UPMB are used to access memory and peripherals. The GPMC is used for Flash, UPMA is used for SDRAM and UPMB is used to access the backplane GPX ASIC on the PETC and MNOC. See section on UPM controller. Initially Flash or CS0 is configured for 30 wait states. Wait state “TA” flag (Terminate Cycle) is determined by setting the “ORx” register “SETA” bit.

4) UPM Memory Controller

The 870 UPM (User Programmable Memory Machine A&B) is an internal 64x32 bit wide memory array whose bits are directly mapped to CS, BS[0:3] and GPL[0:5] output pins used to multiplex the control inputs on SDRAM or GPX3. Freescale provides an MPCinit utility (MPC860 UPM Programming Tool) that graphically displays the UPM array values and waveforms with edit capability. Source file is a “.mgp” text file, output is a “.s” text file. To generate the output file select “code generation” “ordered”. At the end of this “.s” file is a UPMA/B table that FW can paste into the BootRom code.

5) Ethernet 6165 Switch

Complete definition of this sub-system, component or function (Designed functions, Interface description, Operating mode with sequence diagrams, Software source code location, Material characteristics, BOM,….)

Figure 5 is the design of switch interface. A Marvell 88E6165 multi-port Gigabit Switch with five integrated EtherNet triple speed (10/100/1000) PHY’s in MII PHY Mode (or Reverse MII) will communicate with the 870 processor. In all 3 modules the Ethernet Ports will operate at 10/100Mbps and the MII bus at 25Mhz.

A 25Mhz crystal is multiplied by the internal PLL providing all necessary clocks. On power up an 870 GPIO pin holds the 6165 in Reset while it configures the internal registers. Three power supplies are necessary, 1.0V for the Core, 1.8V for the EtherNet PHY’s and 2.5V for I/O. MII RXD data uses a 2.5V to 3.3V (LVC8T245) level shifter to meet the (Vih) input requirements of the 870 processor, MII TXD is 3.3V tolerant along with the other I/O control pins.

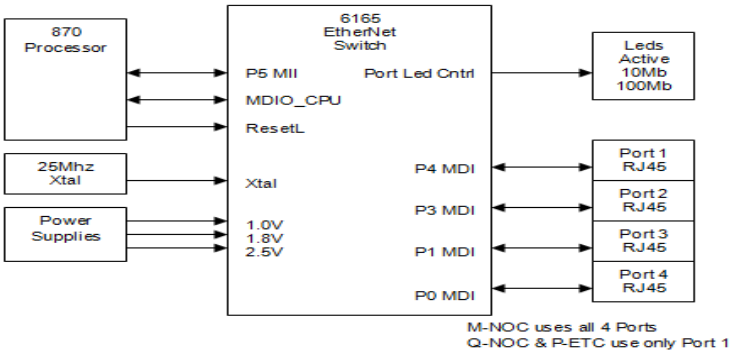

Figure 5. Switch Interface

4.2. CPLD/FPGA Programmable Devices

1) CPLD

A Lattice LC4064ZE with 3.3VIO and 1.8V core contains logic modeled in Verilog after the Loki Premium processor to GPX interface using Lattice Classic V1.2 tools. Device is a 5.8ns device with a TjMax of 105°C and packaged in a 100TQFP.

2) XP2-30 FPGA

A Lattice LFXP2-30E with 3.3VIO/1.2V core containing logic and dual port memory modeled in Verilog after the NOE771 processor to LMS ASIC interface using Lattice ispPRO V7.2 tools. The device on chip flash configures the internal SRam array in 2ms. Device has a TjMax of 100°C with 200 I/O packaged in a 256BGA.

3) FPGA Instantiated Dual Port Memory

Instantiated internally is two 2KB synchronous clocked dual port memories. The 870 CLKOUT is used to clock the memories at 66MHz, thus CPU writes on the falling edge and the LMS ASIC writes on the rising edge. Contention when both sides access same address simultaneously no longer exists as each side is clocked on different edges. This eliminates the dual port busy flag which inserted wait states as presented to the 870 “TA” input. Thus the OR3 SCY field should be set for 2 wait states and the SETA bit set for internal TA.

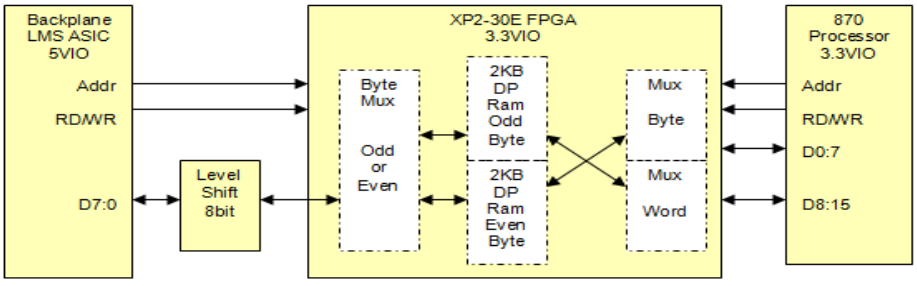

Figure 6. Dual Port Memory Backplane Interface

Figure 6 is the design of dual port memory backplane interface. The dual port memory clock enable or the internal dual port chip enable are accessible on two external pins with pull-ups for debug. When active high indicates a read or write cycle is in progress. CLKENA on Pin K16 or R159 indicates Read/Write to CPU Side & CLKENB on Pin K14 or R158 indicates Read/Write to LMS ASIC Side.

4) FPGA In-Circuit Program

The FPGA onboard flash may be programmed in three ways, as is shown in Figure 7.

a. Lattice HW-USBN-2A in-circuit download cable using Debug Card.

b. FPGA SPI in Slave Mode to 870 SPI in Master Mode.

c. FPGA is downloaded using 870 GPIO running an embedded version of VM.

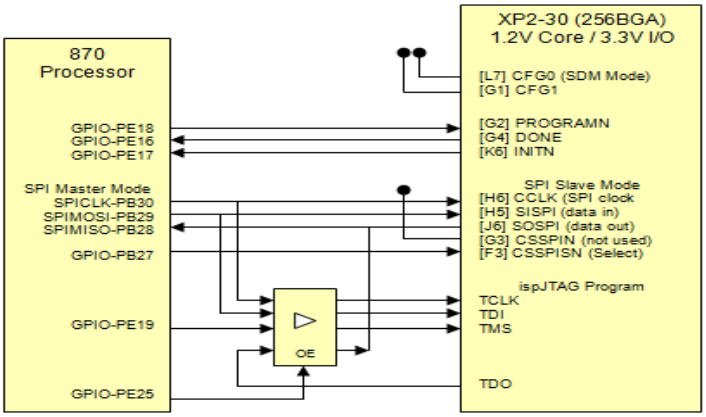

Figure 7. FPGA On Board Program

4.3. FPGA Step2 with Integrated Processor

To enhance backplane performance a larger FPGA (XP2-30) has been selected in order to incorporate an embedded processor and associated program/data memory as a secondary project, as is shown in Figure 8. This implementation would embed a Lattice Mico32 IP processor which would be field upgradeable [14]. The new algorithm would be downloaded to the FPGA using either the SPI bus or an embedded version of the Lattice ispVM system running on the 870. Should a module never desire this option then a lower cost XP2-5 in a 256BGA can be used in its place and would implement only the basic dual port interface. This will require the source code to be recompiled for the smaller device. I/O pins were chosen from the XP2-5 device to insure backwards compatibly.

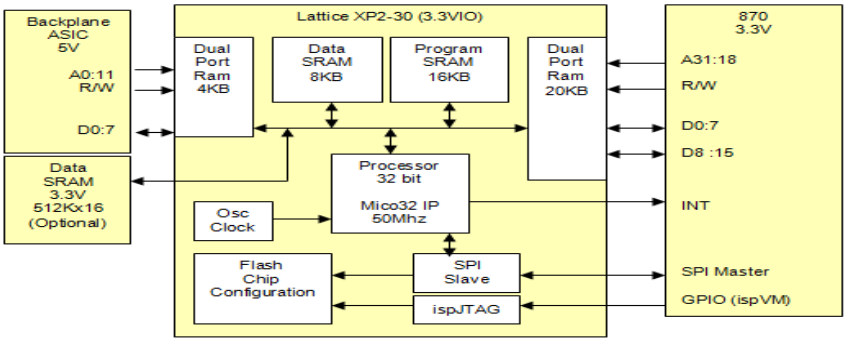

Figure 8. FPGA Step2 with Mico32 Processor

4.4. Power Supplies & Reset

Figure 9 is the design of power supply. Multiple power rails are required to support devices with low voltage cores along with inrush current management on power up or hot swap. All 3 designs share similar power supply architectures with some slight variation. Two synchronous buck regulators (Intersil ISL9105 @600ma & ISL8012 @2A) with integrated fets along with a linear regulator (TI TPS79301 @200ma) are incorporated.

On power up there is a soft start R/C controlled PFET that charges a 100uf tantalum cap to 5 volts. Once the voltage monitor (TI TPS3808) senses the cap voltage at 4.65V for 20ms it then enables all power supplies simultaneously. The switching regulators (3.3, 1.8, 1.2, 1.0V) all have a 1ms built in soft start and the linear regulator (2.5V) tracks the 3.3V rail. Once the regulators have reached approximately 90% of there output voltage the internal PG (Power Good) outputs a 210ms Power OK signal input to the processor.

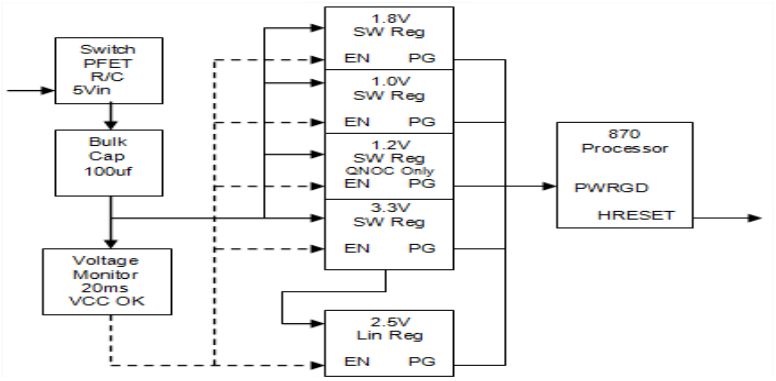

Figure 9. Power Supply

5. Conclusions and Perspective

In this paper, we proposed a new solution to aim the process and field part of the intelligent PLC (Programmable Logic Controller) based industrial Ethernet. It is a complete solution for the target segment: MMM, WWW, F&B, Medium Hydro power. We have discussed the interfaces with the operation & management tools. For the process part the scope of our design includes the controllers, the engineering software tools (Unity Pro) and the interfaces with the other equipment of this level, typically other controllers or tools like asset management, SCADA or OPC server. Safety and HSBY controllers are also part of our design. For the field part, the scope of our program includes the remote I/Os and the interfaces with the other equipment of this level, typically distributed I/Os & devices. In this paper, we make a study on the design of the controller. It includes the system architecture of IPLCbIE, the application of network module. And it focuses on the discussion of design for the kernel module of the PLC-network module NOC. The hardware design is especially introduced in detail in my paper. The main purpose of the NOC module is to provide transparency between the control network, the device network, and an extended distributed I/O network, while preserving device network determinism. In addition, the NOC module also provides services to communicate with PLC applications running on the control network. All of above has been verified and validated in our test for the product.

References

[1] Qian Zhang, Quji Guo, Qiang Ni, Wenwu Zhu, Ya-Qin Zhang. Source adaptive multi-layered multicast algorithms for realtime video distribution. *IEEE/ACM Transactions on Networking*. 2006; 8(6): 720-733.
[2] S McCanne, M Vetterli, V Jacobson. Low-complexity video coding for receiver driven layered multicast. *IEEE Journal on Selected Areas in Communications*. 1997; 15(6): 982-1001.
[3] J Byers, M Frumin, et al. *FLID-DL:congestion control for layered multicast*. Proc. Of NGC2000. Palo Alto,USA. 2000: 71-81.
[4] JC Bolot, T Turletti, I Wakeman. *Scalable feedback control for multicast video distribution in the internet*. Conference of the Special Interest Group on Data Communication ACM. SIGCOMM' 2012: 58-67.
[5] JC Bennett, H Zhang. *Hierarchical packet fair queuing algorithms.* IEEE/ACM Trans. on Networking. 1997; 5(5): 675-689.
[6] Johanson M. *Scalable Video Conferencing Using Subband Transform Coding and Layered Multicast Transmissio.* International Conference on Signal Processing Applications and Technology (ICSPAT'99). Orlando, Florida. 1999: 1-4.
[7] Karan Singh, Rama Shankar Yadav, Manisha Manjul, Rainu Dhir. *Bandwidth Delay Quality parameter Based Multicast Congestion Control.* International Conference on Advanced Computing and Communication (ADCOM 08)at Department of Information Technology. MIT, Anna University,Chennai. 2008: 993-998.
[8] L Vicisano, L Rizzo, J Crowcroft. *TCPlike congestion control for layeredmulticast data transfer*. Proceding of Conference on Computer Communications. 2012: 996-1003.
[9] Quach, Duc-Cuong. An improved direct adaptive fuzzy controller for an uncertain dc motor speed control system. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(2): 1083-1092.

[10] PitchaiVijaya, Karuppanan. Adaptive-fuzzy controller based shunt active filter for power line conditioners. *TELKOMNIKA Indonesian Journal of Electrical Engineering.* 2011; 9(2): 203-210.
[11] DING Xianghui, LI Ping. *An ultrasonic anemometer based on DSP and FPGA.* Applied Acoustics. 2011; 30(1): 46-52.
[12] ADutta, J Chennikara, W Chen. *Multicasting Streaming Media to Mobile Users.* IEEE Communications Magazine. 2003; 10: 81-89.
[13] Legout A Legout, EW Biersack. *Pathological behaviors for RLM and RLC*. International Conference on Network and Operating System Support for Digital Audio and Video, Chapel Hill, NC, USA. 2010: 164-172.
[14] Kwon GI, Byers J. *Smooth multirate multicast congestion control*. Proceedings of IEEE Infocom. 2003; 2: 1022-1032.