International Journal of Engineering Research and Development
ISSN: 2278-067X, Volume 1, Issue 6 (June 2012), PP.01-09
www.ijerd.com

Research Method to Optimize Logger for a Telecom Application Running on Embedded System

M.Rajendra Prasad[1], V.Sharath [2], G.Nagendra[3], K.Damodara Reddy [4], V.Sridhar[5]

[1]Associate Professor, ECE Department, Vidya Jyothi Institute of Technology, Hyderabad, India
[2]Associate Professor, EEE Department, Vardhaman College of Engineering, Hyderabad, India
[3]Associate Professor, ECE Department, VBIT, Proddatur, Kadapa, India
[4]Assistant Professor, EEE Department, Vardhaman College of Engineering, Hyderabad, India
[5]Assistant Professor, ECE Department, Vidya Jyothi Institute of Technology, Hyderabad, India

Abstract**—**Most of the embedded telecom applications running on customized processor boards are needed to record their activities in the form of log messages. These messages help telecom application developers to troubleshoot virtually all kinds of system and application problems. The logger also provides valuable early warning signs and also provides with crucial forensic data .Hence Logger's performance is to be investigated and optimization is required for logger. In this project we are proposing a method to optimize logger for IPBTS application running on embedded system. This paper describes the procedure to asses and analyse logger with the syslog for telecom application and also elaborates the method to measure the logging time with a test application which contains two threads running on customized processor board. These papers presents the procedure to measure logging time and compare it with optimized syslog package logging time on linux operating file system. With this research experiment for logger, we minimized Layer 1 processing time in IPBTS product development project from 1700 micro seconds to 160 micro seconds. Layer 1 processing time includes channel activatation i.e. non combined CCCH channel, 3 TCHF, 1 SDCCH8 and mode of activation is by BTS controller stub. This paper also presents the experimental results to optimize syslog package for IPBTS application development on linux environment running on MPC 8548E processor board.

Keywords**—**IP BTS, layer1, embedded systems, MPC 85xx processor board, linux kernel.

I. INTRODUCTION

This document describes, and is written to conform to, author guidelines for the journals of AIRCC series. It is prepared in Microsoft Word as a .doc document. Although other means of preparation are acceptable, final, camera-ready versions must conform to this layout. Microsoft Word terminology is used where appropriate in this document. Although formatting instructions may often appear daunting, the simplest approach is to use this template and insert headings and text into it as appropriate.

BTS software is responsible for channel encoding/decoding, multiplexing/de-multiplexing, call establishment, speech communication through full rate speech channel, transmission of speech through RTP on Abis interface [3], support of short circuited call, and support of transmission of signalling over TCP/IP on Abis interface [10]. Audit mechanism for UNIX operating system and telecom application messages [4] [5] [13] is defined as syslog. To incorporate logging into their projects, and to provide system administrators with a single point of management for collecting, distributing and processing audit data syslog has been designed. Its design makes it more facile for embedded or telecom application developers and system analysts to perform the below functions. Mostly, the data will be influenced by a lot of people, which finally ends up in a system log. Ball rolling will be started by the developer, by determining

- Events which create a log entry
- Kinds of information in the log entry
- Significance of events
- How the information will be formatted

If the end user has control over the quantity or types of logs that are generated, application administrator or local system should decide the actions that have to be applied to the log data, based on the telecom application or system component that generated the log entry, and the significance assigned to the log entry by its developers. Because of a coarse-grained system, it has lot of options for collecting and acting on syslog data logger [11] [12]. Application administrator or local system can choose amongst the following alternatives:

- write the log entry to a local file(particular to that application)
- write the log entry to a remote system running syslog (loghost) throughout these configuration documents
- The log entry must be broadcasted to users logged into the host computer
- The log entry should be written upon the system console.

To maximize flexibility and facilitate-of-use, architecture has been designed. There are very few guidelines available for system administrators or developers, which implies that implementing an enterprise logging system includes a lot of trial and error.

BEHAVIOUR OF SYSLOG

The file /etc/syslog.conf controls the storage or distribution of data from different applications and subsystems on the host operating system, and of different levels of severity based on embedded system. This configuration file consists of multiple lines. One of the main security log systems [5] [7] on a UNIX system [9], which is naturally implemented on every Linux machine as well, is the log–keeping facility, which logs all user actions, processes, system events etc. The configuration file of the so–called syslog [11] [12] daemon determines which and how long logged information will be kept. The default location of all logs is /var/log, containing different files for access log, server logs, system messages etc.

The syslog function sends a logging message to the logging facility. Each message has a priority argument that is a bitwise or of a severity level and a facility value. The severity level controls how the log message is acted upon and the facility value records the originator of the message.

Facility values (from syslog.h) include LOG_USER, used to indicate that the message has come from a user application, (the default), and LOG_LOCAL0, LOG_LOCAL1, up to LOG_LOCAL7, which can be assigned meanings by the local administrator. The severity levels in descending order of priority are described in table 1.

PRIORITY LEVEL DESCRIPTION

LOG_EMERG	An emergency situation messages
LOG_ALERT	High-priority problem, such as database corruption messages
LOG_CRIT	Critical error, such as hardware failure messages
LOG_ERR	Error messages
LOG_WARNING	Warning messages
LOG_NOTICE	Special conditions requiring attention messages
LOG_INFO	Informational messages
LOG_DEBUG	Debug messages

Table 1

The architecture model of IPBTS telecom application [6] controller is shown in Figure 1. It also represents the BTS Controller block diagram, functionalities and its interfaces with other protocols like Layer 1, Link Access Protocol for the Dm channel (LAPDm), LAPD Interface (LAPDIF), and Radio Resource Management at BTS (RRBTSM). This paper describes the optimization method for logger which reduces Layer 1 processing time and all other logging functionalities.

II. RELATED WORKS

Today due to the of economical leverage and overwhelming advancements in the development of the IP based telecom applications running on embedded system researchers [2] are focussing on IP based telecom appositeness issues and investigating new approaches. Musabekov S.B describes [10] the procedure with establishing GSM link over Satellite Abis interface, which is defined between base transceiver station (BTS) and base station controller (BSC). An augmented performance approach being explored by simulation. Rajendra Prasad M [2] proposes the procedure for transplanting linux kernel on PowerPC based custom board which is considered as an embedded system targeted for IPBTS application software. GU zhaojun wang chao presented a secured host based secure state model [1].Horn. B discussed a web based logger for general purpose set of component interfaces for logging data in a heterogeneous computing distributed environment[8]. Rajendra Prasad emphasizes the research method describing with elaborate design of RTP module for IP-BTS. BTS provides the physical connection of an MS to the network in the form of Air-interface. [3]. In this paper we are describing the research method to optimise Layer 1 log messages processing time in IPBTS product development research project.

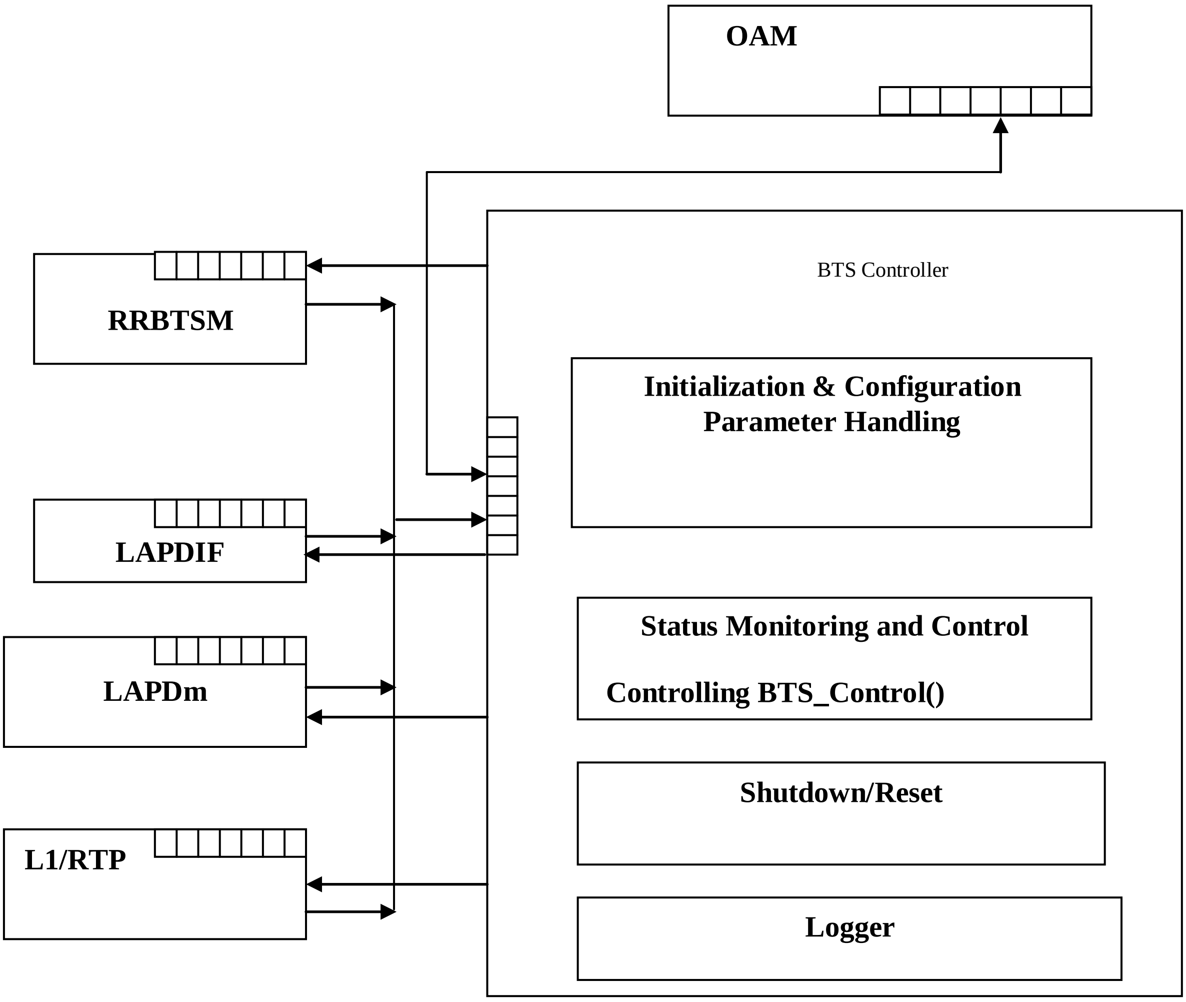

OAM
BTS Controller
RRBTSM
Initialization & Configuration Parameter Handling
LAPDIF
Status Monitoring and Control
Controlling BTS_Control()
LAPDm
Shutdown/Reset
L1/RTP
Logger

Figure 1: BTS Controller

III. EXPERIMENT SET UP

For the above experiment a customized processor board is designed for a telecom IPBTS application logger testing. The technical specifications are tabulated in Table 1. The PPC based target boards are based on free scale MPC8548E PowerPC processor. The Processor board is designed as flexible board to accommodate MPC8548E and MPC8543E processors. The top side view and bottom side view of customized board are depicted in Fig 3 Fig 4. The detailed information about the processor (MPC 8548E) can be available in [16].Because of the open source, rich code resource and for supporting many kinds of CPU architecture, linux is making steady progress in the embedded arena.

PROCESSOR BOARD SPECIFICATIONS

CPU	Memory	Ethernet	Operating System
Free scale MPC8548E PowerPC Processor	DDR2 - 256 MB Boot Flash – 32 MB User Flash – 32 MB 2 serial EEPROMS (Boot sequencer EEPROM and Board ID EEPROM of 8Kb)	Two 10/100 Ethernet ports (DP83848C Fast PHY Ethernet ICs)	Embedded Linux (2.6.10)

Table 2

In the recent years linux as operating system has been attractive choice in many embedded telecom applications. Optimization of syslog is tested on linux operating system and its version is 2.6.10.Porting of linux kernel to the custom boards is described by M.Rajendra Prasad [2]. The images for this embedded system are shown in the figure 2. U-boot is used as a boot loader and linux kernel image of 2.6.10 version running on customized processor boards along with customized file system with optimized syslogd-1.4.1 package. The ram disk file system contains the BTSSW and syslogd-1.4.1 package, which is a vital portion of this research project. The source code files referenced above are taken from open source repositories. There will be changes in these components to minimize Layer1 processing time and other functionalities and these changes are beyond the scope of this paper. During the experiment presented in this paper, execution time of two threads (Thread 1 and Thread 2) on Processor board with optimized code of syslogd package are tabulated

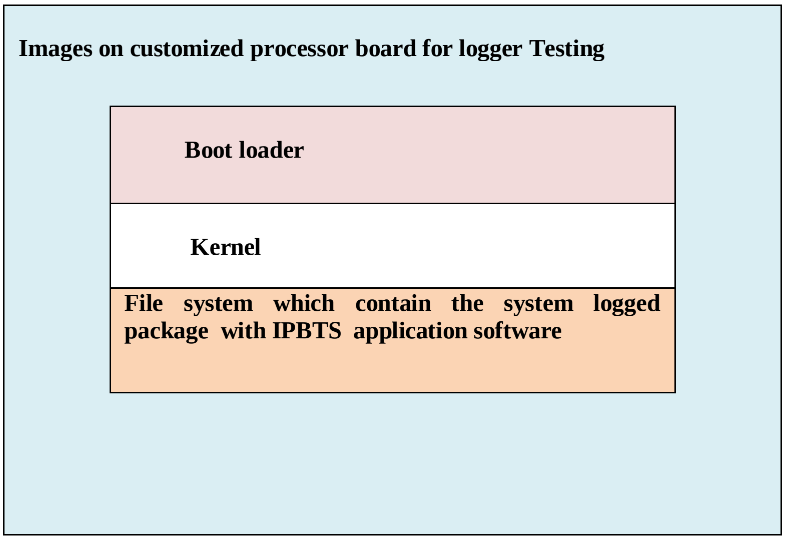

Images on customized processor board for logger Testing
Boot loader
Kernel
File system which contain the system logged package with IPBTS application software

Figure 2 Processor board (IPBTS Controller) images

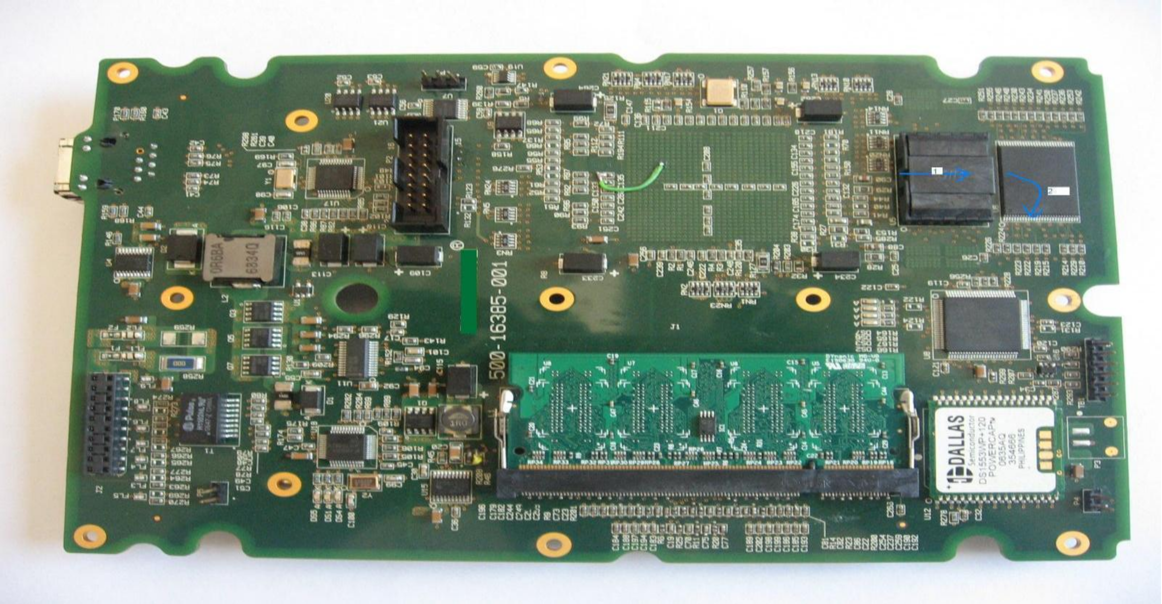


Figure .3 Top side view of custom based embedded system board

IV. RESULTS AND ANALYSIS

In this section, optimization method for logger is proposed and results of a test application comprises with two threads are analysed.

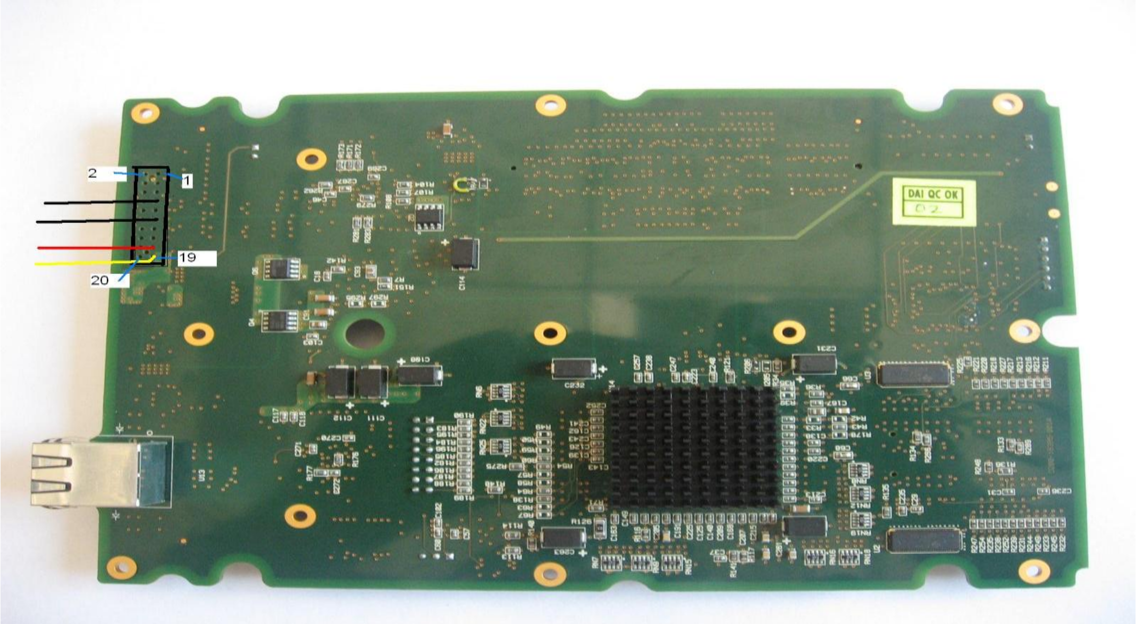


Figure .4 Bottom side view of custom based embedded system board.

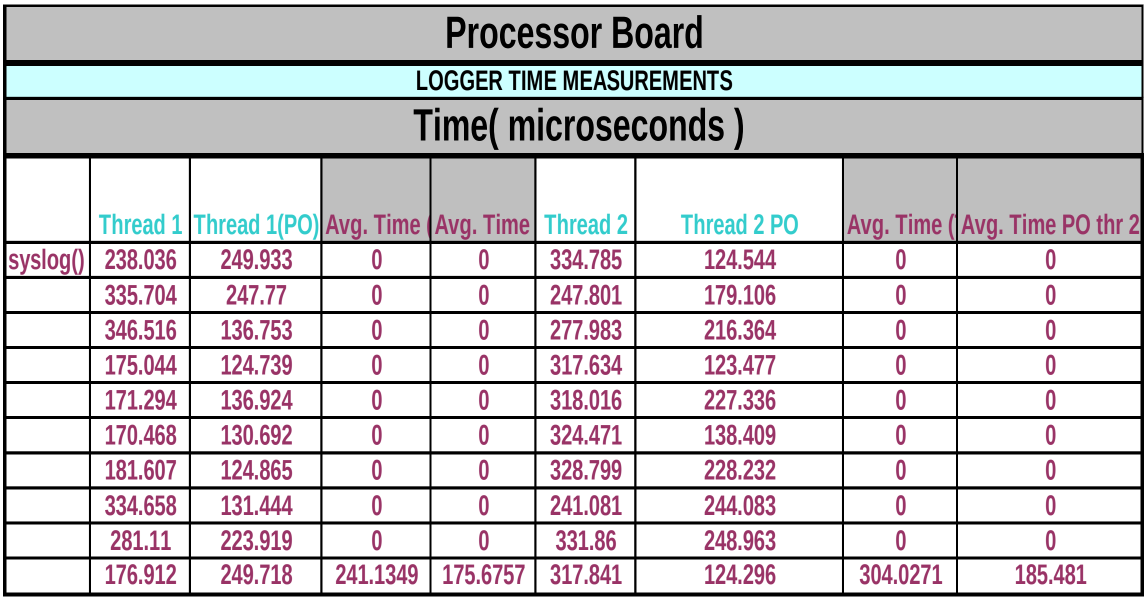
PROPOSED OPTIMIZATION METHOD

1. LOG_PERROR flag was removed from openlog() system call as it was printing the messages being logged on the console
2. Code relating to gethostbyname method is removed from openlog() system call
3. Code regarding UDP socket is removed as in our project we do not require remote logging option.
4. Select system call code is optimized as only UNIX domain sockets are being created.

A. TESTING OF LOGGER BY APPLYING PARTIAL OPTIMIZATION METHOD (PO)

A test application is designed to measure the logger performance. Test application comprises of two threads that are created and calculated the logging time for syslog [12] as shown in the table 3 on the processor board on linux 2.6.10 version environment. Each time measurement above (except Avg. Time measurement) is calculated for an average of 1000 readings. Avg. Time is the average of five set of readings as shown the Table III. The average logging time for thread 1 is 241.1349 micro sec on processor board without any optimization technique.

After applying of step 1 of optimization method partial optimization (PO) thread 1 processing time is reduced to 175.6757 micro sec as shown in the Table III and same technique is analysed and assessed in graph as shown in the fig 5..Similarly thread 2 log processing times are analysed and depicted in Table 3 and fig 6 as a graph..



Processor Board

LOGGER TIME MEASUREMENTS

Time(microseconds)

	Thread 1	Thread 1(PO)	Avg. Time (	Avg. Time	Thread 2	Thread 2 PO	Avg. Time (	Avg. Time PO thr 2
syslog()	238.036	249.933	0	0	334.785	124.544	0	0
	335.704	247.77	0	0	247.801	179.106	0	0
	346.516	136.753	0	0	277.983	216.364	0	0
	175.044	124.739	0	0	317.634	123.477	0	0
	171.294	136.924	0	0	318.016	227.336	0	0
	170.468	130.692	0	0	324.471	138.409	0	0
	181.607	124.865	0	0	328.799	228.232	0	0
	334.658	131.444	0	0	241.081	244.083	0	0
	281.11	223.919	0	0	331.86	248.963	0	0
	176.912	249.718	241.1349	175.6757	317.841	124.296	304.0271	185.481

Table 3

B. TESTING OF LOGGER BY APPLYING OPTIMIZATION METHOD.

After applying proposed optimized method discussed above to syslogd-1.4.1 package the same thread1 processing time with the optimized syslog is reduced to 61.005 micro sec and thread 2 is reduced to 102.8871 as shown in the Table 4 and a graph is drawn and analysed in graph 3.

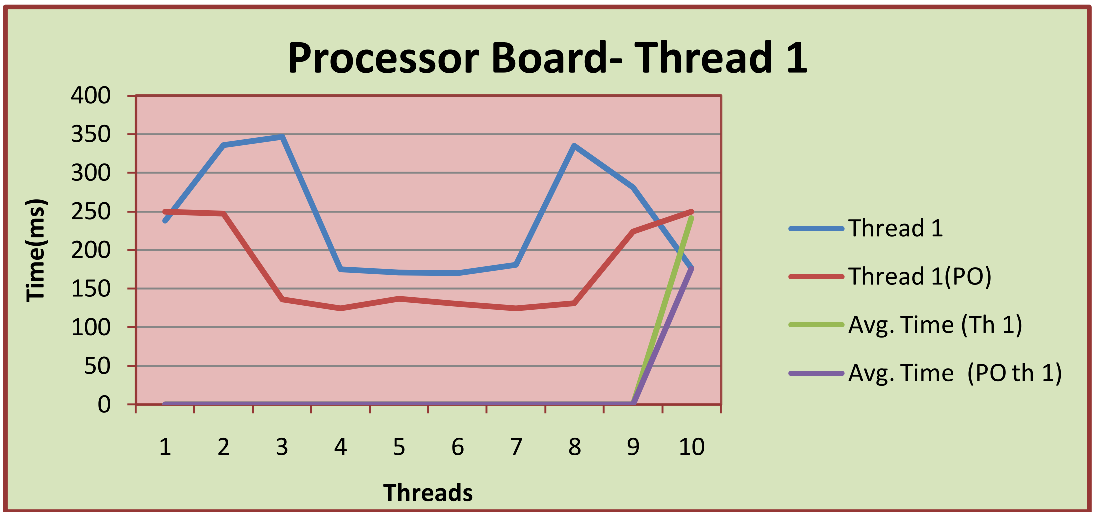


Figure.5 Performance of thread 2 on custom based embedded system board.

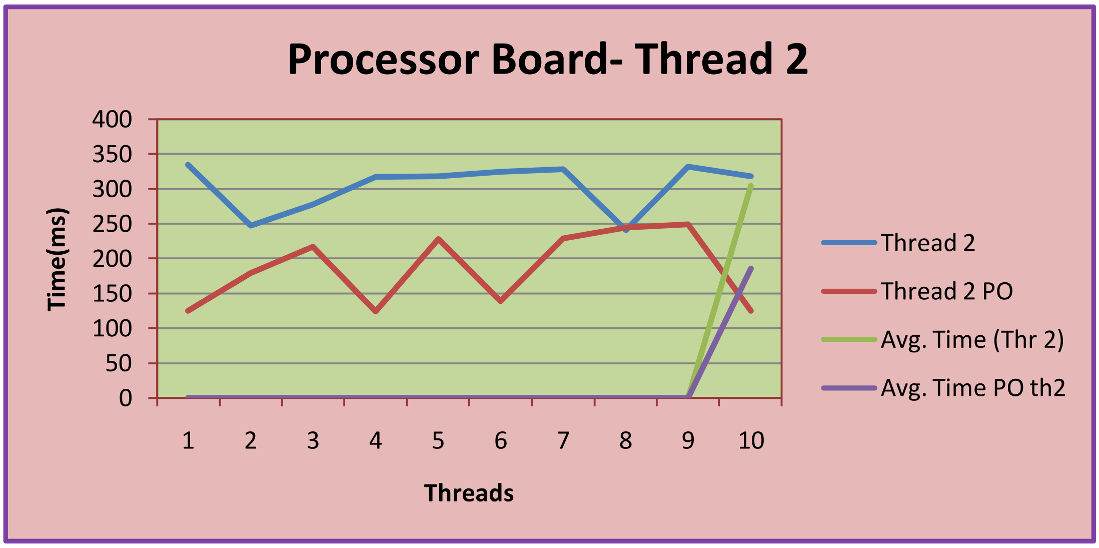
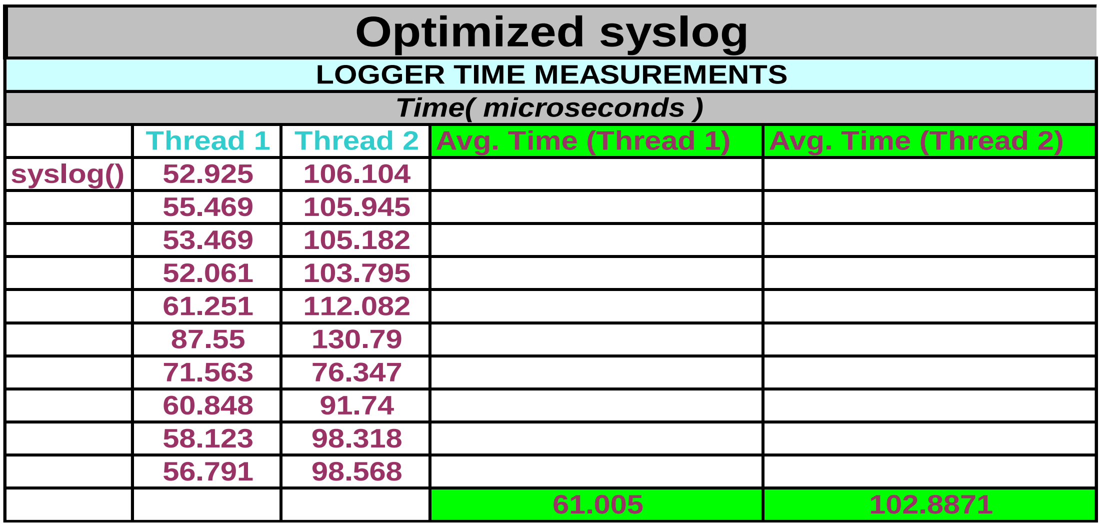


Figure .6 Performance of thread 2 on custom based embedded system board



Optimized syslog				
LOGGER TIME MEASUREMENTS				
Time(microseconds)				
	Thread 1	Thread 2	Avg. Time (Thread 1)	Avg. Time (Thread 2)
syslog()	52.925	106.104		
	55.469	105.945		
	53.469	105.182		
	52.061	103.795		
	61.251	112.082		
	87.55	130.79		
	71.563	76.347		
	60.848	91.74		
	58.123	98.318		
	56.791	98.568		
			61.005	102.8871

Table 4

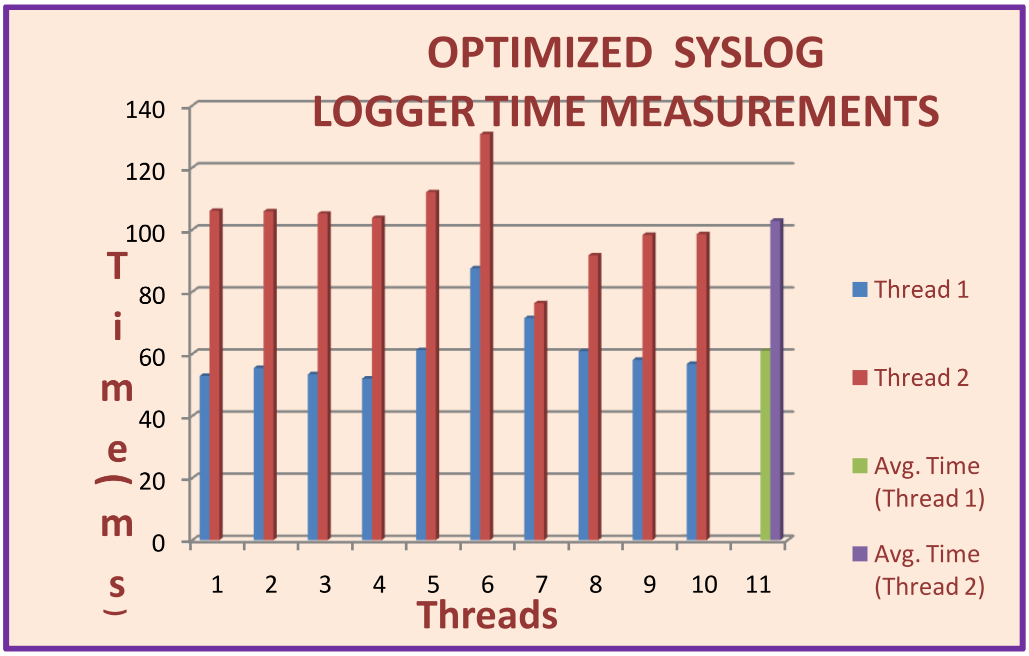
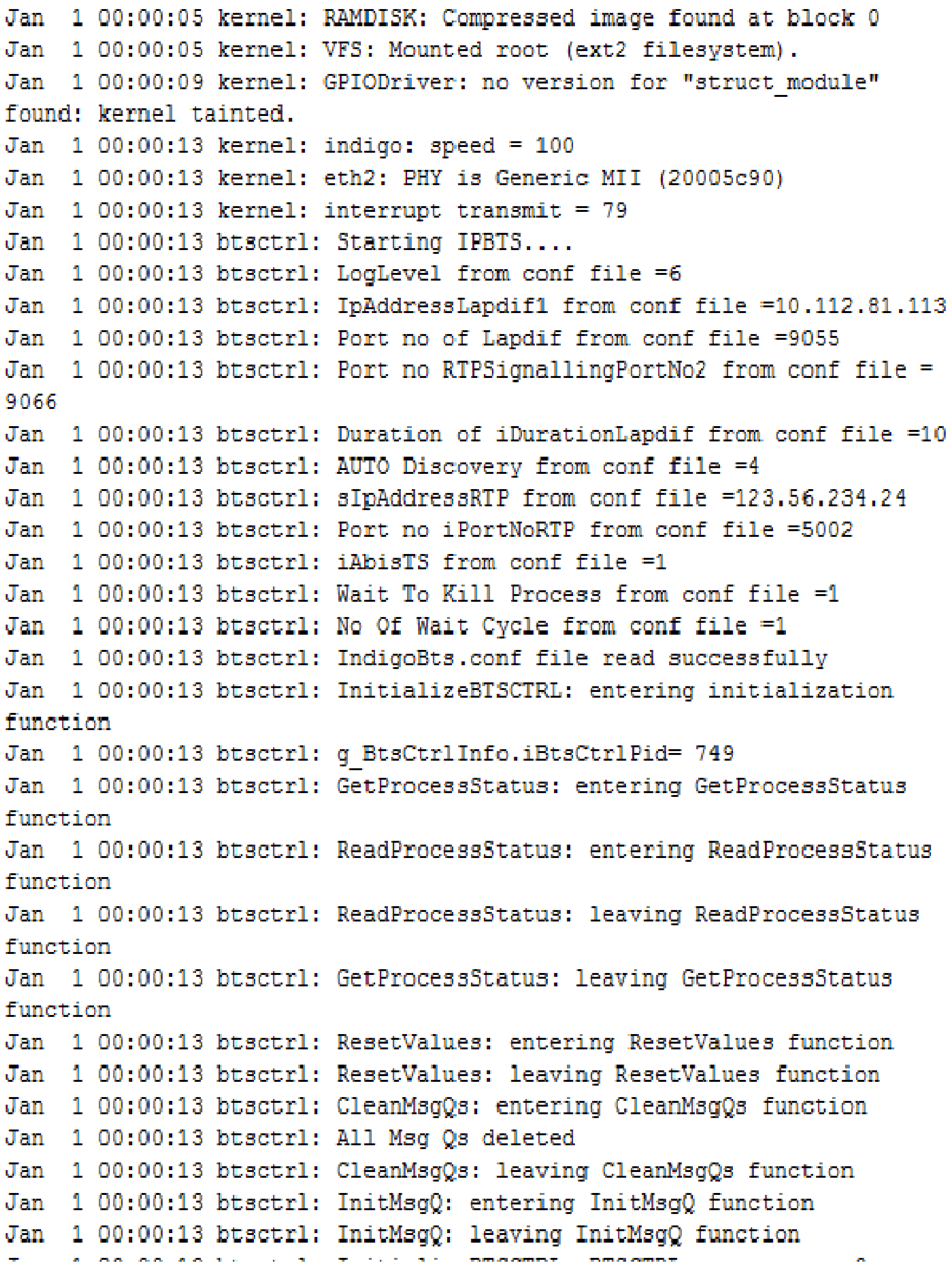


Fig.7 Performance of thread 1 & 2 on custom based embedded system board with optimized syslog.

IP BTS Controller application running on customized PowerPC based processor board will support default logging and tracing for all the protocols and modules using the linux logging facilities. In this section we have optimized logger to minimize logging time for our IPBTS research project. The log messages for IPBTS with optimized syslogd-1.4.1 package in the linux file system are depicted in the figure 8.



```
Jan  1 00:00:05 kernel: RAMDISK: Compressed image found at block 0
Jan  1 00:00:05 kernel: VFS: Mounted root (ext2 filesystem).
Jan  1 00:00:09 kernel: GPIODriver: no version for "struct_module"
found: kernel tainted.
Jan  1 00:00:13 kernel: indigo: speed = 100
Jan  1 00:00:13 kernel: eth2: PHY is Generic MII (20005c90)
Jan  1 00:00:13 kernel: interrupt transmit = 79
Jan  1 00:00:13 btsctrl: Starting IPBTS....
Jan  1 00:00:13 btsctrl: LogLevel from conf file =6
Jan  1 00:00:13 btsctrl: IpAddressLapdif1 from conf file =10.112.81.113
Jan  1 00:00:13 btsctrl: Port no of Lapdif from conf file =9055
Jan  1 00:00:13 btsctrl: Port no RTPSignallingPortNo2 from conf file =
9066
Jan  1 00:00:13 btsctrl: Duration of iDurationLapdif from conf file =10
Jan  1 00:00:13 btsctrl: AUTO Discovery from conf file =4
Jan  1 00:00:13 btsctrl: sIpAddressRTP from conf file =123.56.234.24
Jan  1 00:00:13 btsctrl: Port no iPortNoRTP from conf file =5002
Jan  1 00:00:13 btsctrl: iAbisTS from conf file =1
Jan  1 00:00:13 btsctrl: Wait To Kill Process from conf file =1
Jan  1 00:00:13 btsctrl: No Of Wait Cycle from conf file =1
Jan  1 00:00:13 btsctrl: IndigoBts.conf file read successfully
Jan  1 00:00:13 btsctrl: InitializeBTSCTRL: entering initialization
function
Jan  1 00:00:13 btsctrl: g_BtsCtrlInfo.iBtsCtrlPid= 749
Jan  1 00:00:13 btsctrl: GetProcessStatus: entering GetProcessStatus
function
Jan  1 00:00:13 btsctrl: ReadProcessStatus: entering ReadProcessStatus
function
Jan  1 00:00:13 btsctrl: ReadProcessStatus: leaving ReadProcessStatus
function
Jan  1 00:00:13 btsctrl: GetProcessStatus: leaving GetProcessStatus
function
Jan  1 00:00:13 btsctrl: ResetValues: entering ResetValues function
Jan  1 00:00:13 btsctrl: ResetValues: leaving ResetValues function
Jan  1 00:00:13 btsctrl: CleanMsgQs: entering CleanMsgQs function
Jan  1 00:00:13 btsctrl: All Msg Qs deleted
Jan  1 00:00:13 btsctrl: CleanMsgQs: leaving CleanMsgQs function
Jan  1 00:00:13 btsctrl: InitMsgQ: entering InitMsgQ function
Jan  1 00:00:13 btsctrl: InitMsgQ: leaving InitMsgQ function
```

Figure .8 Optimized log messages of IPBTS

V. CONCLUSIONS

In this paper we proposed an optimized logger for telecom application (IPBTS) running on customized PPC [15] based board which reduces 10 times of layer 1 processing time and also syslog performance is analyzed and assessed for IPBTS application. The performance levels are very much superior than the existing open source syslog package. The changes also discussed briefly this paper. This method can be applied to any syslog based log systems for any telecom application running on any platforms for different operating systems.

VI. ACKNOWLEDGEMENTS

Thanks to Wireless & Embedded System Group, Prof. G.Sri Ram Reddy and faculty members of ECE Department, VJIT for their suggestions and encouragement in the process of this research paper. We would like to dedicate this paper to Mr. M. Rajendra Prasad’s mother Late B.Suseelamma for her motivation towards this research project.

REFERENCES

[1]. GU zhaojun wang chao,"Statistic and analysis for host-based syslog", *Education technology and computer science* (ETCS), 2010 *second international workshop*,vol: 2, pp.277-280, 6-7 march 2010.

[2]. Rajendra Prasad.M, S. Ramasubba Reddy, V.Sridhar, "Framework to port linux kernel on powerpc based embedded system used for telecom application – ipbts", International Journal of Software Engineering & Applications (IJSEA), Vol. 2, No.4, pp127-139, 2011

[3]. Rajendra Prasad.M, G.Nagendra, Ravi Mathey , V.Sridhar "Research Method For Transmission Of Speech Through RTP Packets On Abis Interface For IPBTS", International Journal of Computer Information Systems,Vol. 3, No. 6, 2011

[4]. Slavicek, k. ledvinka, J.Javornik, M.Dostal,"Mathematical processing of syslog messages from routers and switches", Inst. Of comput.sci, Masaryk univ., Bron, in Information and automation for sustainability, ICIAFS 4th international conference, pp.463-468, 12-14 dec 2008

[5]. Tsunoda, H.maruyarna, T.ohta, K.waizurni, Y.keeni, G.M.Nernoto,"A prioritized retransmission mechanism for reliable and efficient delivery of syslog messages'', Fac. Of eng., tohoku inst. Of technol., Sendai, in Communication networks and service research conference, CNSR'09 seventh annual, pp.:158-165, 11-13 may 2009.

[6]. Chinghway lim singh, n.yajnik,"A log mining approach to failure analysis of enterprise telephony systems", Dependable systems and Network with FTCS and DCC, 2008 IEEE international conference, pp: 398-403, 24-27 june 2008.

[7]. Forte,D.V. maruti,C.vetturi,M.R.zambelli,"Secsyslog:an approach to secure logging based on covert channels",Incident response Italy project,univ. of Milano,crema,Italy, in Systematic approaches to digital forensic engineering, first international workshop,pp:248-263,7-9 nov,2005.

[8]. Horn,B. balakrishnan,H.Maniampadavathu,B.T.Warnes,J.elko,D.A,"A logger system based on web services",IBM systems journal,Vol:43 issue:4,pp:723-733,2004.

[9]. Simache,c.kaaniche,"Availability assessment of sunos/solaris unix systems based on syslogd and wtmpx log files:a case study,Laas-Cnrs,Toulouse,france,in dependable computing,11th pacific rim international sysmposium,pp:8,12-14 dec 2005.

[10]. Musabekov S.B, Srinivasan, P.K., Durai, A.S. Ibroimov R.R, (2008) "Simulation analysis of abis interface over IP over DVB-S2-RCS in a GSM over satellite network", *ICI 4th IEEE/IFIP International Conference* BC *Transactions on ECE*, Vol. 10, No. 5, pp120-122

[11]. Monteiro, S.D.S. erbacher,"Exemplifying attack identification and analysis in a novel forensically viable syslog model*", Systematic approaches to digital forensic engineering, SADFE'08 third international workshop*, pp: 57-68, 22 may 2008.

[12]. Fukuda, K. presto jst,"On the use of weighted syslog time series for anomaly detection*", integrated network management*, pp: 393-398, 23-27 may 2011.

[13]. K. Slavicek, J. Ledvinka, M. Javornik. "Mathematical Processing of Syslog Messages from Routers and Switches", 2008IEEE.

[14]. J.Kelsey, J.Callas, and A.Clemm. Signed syslog Messages. Internet Draft, December 2008.

[15]. "The PowerPC Architecture: A Specification for a New Family of RISC Processors", Second Edition, by International Business Machines, Inc.

[16]. MPC8548E PowerQUICC III™ Integrated Host Processor Family Reference Manual, MPC8548ERM Rev. 1, 2005.

AUTHORS

M.Rajendra Prasad obtained his B.E and M.E Electronics and Communication Engineering from SK University and Osmania University, Hyderabad respectively. He has 12 years of experience in embedded and telecom application research and development. He is currently working as Associate Professor, ECE Department, VJIT, Hyderabad. He has authored several Technical papers in International magazines and had more papers in Internatinational Journals. His main research interests are embedded systems, wireless protocols and RTOS.

V.Sharath Babu is working as Associate. Professor in EEE department in Vardhaman College of Engineering (Autonomous) Hyderabad, INDIA. He has 8 years of experience in electrical applicationdevelopment and control system engineering. His research interests includes embedded systems for electrical and electronic applications.

G.NAGENDRA is working as Associate. Professor in Department of Electronics and communication Engineering for VBIT, Proddatur, Kadapa . He completed M.Tech (Communications and Signal Processing) from S.K University, Ananthapur. He completed B.Tech Degree in E.C.E from J.N.T.U.H. His area of research includes Digital Signal Processing, Digital Image Processing, Embedded Systems, Mobile communications and Artificial neural networks.

K. Damodara Reddy is working as a senior Assistant professor in EEE department in Vardhaman College of Engineering (Autonomous) Hyderabad INDIA. He has 4 years of experience in research. His research interests includes embedded systems for Electrical & Electronics applications, power electronics and multi level inverters.

V. Sridhar working as Assistant professor in ECE department at Vidya Jyothi Institute of Technology, Hyderabad from 2009. He completed M. Tech with Specialization Wireless and mobile communication systems from vardhaman college of engineering (AUTONOMOUS) JNTUH in 2011.He has completed M.Sc (IT) from Nagarjuna University and completed Electronics and telecommunication engineering from VJIT, JNTUH in 2007. His areas of research interests include Wireless and Mobile communications, Digital signal processing, image processing, Telecommunications, communication systems, signal processing. He is Lifetime Member of ISTE and IETE.