WEB-BASED DATA ACQUISTION SYSTEM USING A 32 BIT SINGLE BOARD COMPUTER AND GNU/LINUX

WAN MUHAMAD AZMI MAMAT

UNIVERSITI MALAYSIA PERLIS

2009

WEB-BASED DATA ACQUISITION SYSTEM USING A 32 BIT SINGLE BOARD COMPUTER AND GNU/LINUX

By

WAN MUHAMAD AZMI BIN MAMAT
(0630210069)

A thesis submitted
In fulfilment of the requirements for the degree of
Master of Science (Computer Engineering)

School of Computer and Communication Engineering
UNIVERSITI MALAYSIA PERLIS

2009

UNIVERSITI MALAYSIA PERLIS

Authors Full Name	:	WAN MUHAMAD AZMI BIN MAMAT
Date of Birth	:	SEPTEMBER 11, 1982
Title	:	WEB-BASED DATA ACQUISITION SYSTEM USING A 32 BIT SINGLE BOARD COMPUTER AND GNU/LINUX
Academic Session	:	2008 / 2009

I, hereby declare that the thesis becomes the property of Universiti Malaysia Perlis (UniMAP) and to be placed at the library of UniMAP. This thesis is classified as :

CONFIDENTIAL	☐	(Contains confidential information under the Official Secret Act 1972)*
RESTRICTED	☐	(Contains restricted information as specified by the organization where research was done)*
OPEN ACCESS	/	I agree that my thesis is to be made immediately available as hard copy or on-line open access (full text)

I, the author, give permission to the Universiti Malaysia Perlis to reproduce this thesis in whole or in part of the purpose of research or academic exchange only (except during a period of ……. years, if so requested above).

Certified by:

…….	…….
SIGNATURE	**SIGNATURE OF SUPERVISOR**
820911-11-5241	ASSOCIATE PROFESSOR DR. R. BADLISHAH BIN AHMAD
……..	……
(PASSPORT NO. / NEW IC NO.)	NAME OF SUPERVISOR
Date: July 15, 2009	Date: July 15, 2009

NOTES : * if the thesis is CONFIDENTIAL or RESTRICTED, please attach with the letter from the organization with period and reasons for confidential or restriction.

GRADUATE SCHOOL

UNIVERSITI MALAYSIA PERLIS

PERMISSION TO USE

In presenting this thesis in fulfillment of a post graduate degree from the Universiti Malaysia Perlis, I agree that permission for copying of this thesis in any manner, in whole or part, for scholarly purpose may be granted by my supervisor(s) or, in their absence, by the Dean of the Graduate School. It is understood that any copying or publication or use of this thesis or part thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and Universiti Malaysia Perlis, for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or in part should be addressed to:

Dean of Graduate School

Universiti Malaysia Perlis (UniMAP)

Jalan Meranti Paya, Off Jalan Bukit Lagi

01000 Kangar

Perlis

APPROVAL AND DECLARATION SHEET

This thesis titled Web Based Data Acquisition System using a 32 Bit Single Board Computer and GNU/Linux was prepared and submitted by **Wan Muhamad Azmi bin Mamat (Matrix Number: 0630210069)** and has been found satisfactory in terms of scope, quality and presentation as partial fulfilment of the requirement for the award of degree of Master of Science (Computer Engineering) in Universiti Malaysia Perlis (UniMAP). The members of Supervisory committee are as follows:

R. BADLISHAH BIN AHMAD, Ph. D.
Associate Professor
School of Computer and Communication Engineering
Universiti Malaysia Perlis
(Main Supervisor)

MOHD RIZON BIN MOHAMED JUHARI, Ph. D.
Associate Professor
School of Mechatronic Engineering
Universiti Malaysia Perlis
(Co-Supervisor)

Checked and Approved by

.....
(ASSOCIATE PROFESSOR DR. R. BADLISHAH BIN AHMAD)
Dean / Main Supervisor
School of Computer and Communication Engineering
Universiti Malaysia Perlis

(Date: ……)

School of Computer and Communication Engineering
Universiti Malaysia Perlis

ACKNOWLEDGEMENT

Alhamdulillah, the only One that must be praised is Allah, who gave me the life, guidance and strength to work from the beginning to the end of this thesis.

I would like to extend my gratitude to my respectful supervisor, Associate Professor Dr. R. Badlishah bin Ahmad, who guided me with sincerity and commitment during this research work.

Thank you very much to my beloved wife, Nurul Wahidah binti Arshad, for her never ending support and always stands by me. I would like to express my gratitude and happiness to my parents, Wan Mamat bin Wan Muda and Che Besah binti Ali, my mother in law, Halimah binti Mat Amin, and all my family relatives. Thank you very much for your understanding and support.

My colleagues have worked closely with me on this research work. I would like to thank to Ahmad Nasir, Shuhaizar, Faizal, Abdul Halis, Norzila and Farhan for their help and suggestion. Thanks to Mostafijur, Nasim, Yachine, Hilal and the rest of Embedded Computing Cluster members for your cooperation.

Thanks to all staff of the School of Computer and Communication Engineering, Universiti Malaysia Perlis especially for Mr. Zahereel Ishwar and Dr. Ghosson for their suggestions and ideas. My appreciation also goes to laboratory technicians, Mr. Basri and Mr. Zhahir who gave me permission to use the laboratory equipment. Thank you to Associate Professor Shaikh Nasir and Mrs. Zinnirah who helped me a lot in compiling this thesis.

A special thank to the Ministry of Science, Technology and Innovation (MOSTI) for sponsoring me during this research work. Thank you to all Graduate School staff for their great job in assisting postgraduate students.

Last but not least, I would like to express my greatest appreciation to all who have helped me in doing this research, may Allah bless all of you.

Wan Muhamad Azmi bin Mamat
Universiti Malaysia Perlis
wm.azmi@yahoo.com

TABLE OF CONTENTS

Page

DECLARATION OF THESIS i

PERMISSION TO USE ii

APPROVAL AND DECLARATION SHEET iii

ACKNOWLEDGEMENT iv

TABLE OF CONTENTS vi

LIST OF TABLES xi

LIST OF FIGURES xii

LIST OF ABBREVIATIONS xvii

ABSTRAK (BM) xx

ABSTRACT (ENGLISH) xxi

CHAPTER 1: INTRODUCTION

1.1 Overview ……...1

1.2 Problem Statement ……2

1.3 Research Objectives ……..2

1.4 Research Scopes……3

1.5 Thesis Organization ……... 3

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction …….. 4

2.2 Acquisition Hardware ……... 5

2.2.1 PC-Based Acquisition Platform ……5

2.2.2 Embedded-Based Acquisition Hardware ……..6

2.2.2.1 Microcontroller ……8

2.2.2.2 Field Programmable Gate Array ……...9

2.2.2.3 Digital Signal Processing Board ……... 10

2.2.2.4 Single Board Computer ……10

2.2.2.5 The Keys of SBC Selection …….. 13

2.3 Data Transmission …… 14

2.3.1 Serial Communication …….. 15

2.3.2 Ethernet and Wireless Networking ……... 16

2.3.3 Client-Server Communication Model……... 19
2.3.3.1 Transmission Control Protocol (TCP) …… 20
2.3.3.2 User Datagram Protocol (UDP) ……... 21
2.4 Embedded Linux Platform ……22
2.4.1 The Advantages of Linux …… 23
2.4.2 Real-Time Operating System …… 24
2.5 Summary ……...24

CHAPTER 3: METHODOLOGY

3.1 Introduction …….. 26
3.2 System Design……26
3.3 Software Development Modules ……...29
3.4 Programming Languages ……... 30
3.4.1 C Programming ……31
3.4.2 Shell Script ……32
3.5 Socket Programming ……33
3.5.1 TCP Fundamental…… 34
3.5.2 UDP Fundamental ……37
3.6 Open Source Application Software's ……... 38
3.6.1 Apache Server……... 38
3.6.2 MySQL …… 39
3.6.3 PHP ……... 40
3.6.4 Gnuplot …… 41
3.7 Operating System ……..42
3.7.1 Embedded Operating System – TSLinux …… 43
3.7.2 Source Code Compilation ……44
3.8 Summary ……...46

CHAPTER 4: HARDWARE PLATFORM

4.1 Introduction …….. 48
4.2 PESS's Hardware Component ……... 48
4.2.1 TS-5500 Single Board Computer ……..49
4.2.1.1 Memory …….. 51
4.2.1.2 I/O Address ……52

4.2.1.3 Serial Port 52
4.2.2 Sensors 53
4.2.3 Matrix Keypad 54
4.2.4 Alphanumeric LCD Panel 55
4.3 Server for PESS-n System 55
4.4 Hardware Setup 56
4.4.1 Getting Started with TS-5500 56
4.4.2 Analog Input Reading 58
4.4.2.1 Interfacing Analog Input to ADC 58
4.4.2.2 ADC Register's Configuration on TS-5500 60
4.4.2.3 Experimental Setup for High Output Impedance Sensor 61
4.4.3 Matrix Keypad Setup 62
4.4.4 LCD Setup 62
4.5 Summary 64

CHAPTER 5: SYSTEM DESIGN AND IMPLEMENTATION

5.1 Introduction 65
5.2 DAS System Overview 65
5.3 DAS Software Architecture 68
5.4 PESS Software Development 69
5.4.1 Analog Input Preprocessing 70
5.4.1.1 Converting Analog Input to Digital Output 70
5.4.1.2 Bitwise Operation 79
5.4.2 Interactive System Design 82
5.5 Data Transmission 88
5.5.1 Socket using TCP Protocol 88
5.5.1.1 Client Process over TCP 89
5.5.1.2 Server Process over TCP 93
5.5.2 Socket using UDP Protocol 96
5.5.2.1 Client Process over UDP 97
5.5.2.2 Server Process over UDP 103
5.6 Web Page Development 106
5.6.1 PHP Implementation 106
5.6.2 MySQL Database Application 107

5.6.3 Integrating PHP and MySQL ……108
5.7 Shell Scripting Implementation …… 108
5.7.1 Loading Data into Database ……..109
5.7.2 Date Parameters Calculation ……110
5.7.2.1 Calculating Date Parameters for Current Data ……110
5.7.2.2 Determining Plot Mode for Selected Data ……115
5.7.3 Plotting Graph using *Gnuplot* ……... 120
5.7.3.1 Current Data Plot Format ……121
5.7.3.2 Selected Data Plot Format ……...122
5.7.4 Automate Running Process ……...123
5.8 Summary ……124

CHAPTER 6: RESULTS AND ANALYSIS

6.1 Introduction …….. 125
6.2 Verification of System Functionalities ……. 125
6.2.1 Analog Input Reading Verification ……...126
6.2.2 Data Transmission Verification …… 127
6.2.3 Database System ……... 130
6.2.4 Date Calculation Verification ……... 130
6.2.5 Plotting using Gnuplot …….. 132
6.3 PESS Implementation ……... 133
6.3.1 ADC Reading Result ……134
6.3.1.1 Bit Error ……..134
6.3.1.2 Reading Accuracy …….. 139
6.3.2 Sensor' Readings ……...143
6.3.3 Result of Interactive System Testing …… 147
6.4 Data Transmission Performance Evaluation ……150
6.4.1 Recovery Time …….. 151
6.4.2 Maximum Backup Data …… 158
6.4.3 Data Losses ……... 160
6.4.4 TCP and UDP Overall Performance …… 165
6.5 User Interface Implementation …… 167
6.5.1 Current Data Result …….. 170
6.5.2 Selected Data Result …… 171

6.6 Summary ……...174

CHAPTER 7: CONCLUSION AND FUTURE WORKS

7.1 Introduction ……...176

7.2 Future Works ……178

7.3 Research contributions ……179

PUBLICATIONS 180

REFERENCES 181

APPENDICES

Appendix A: Hardware Specification of TS-5500 SBC 188

Appendix B: Specification of OPA2111 Operational Amplifier 189

Appendix C: Specification of Max 197 ADC 192

Appendix D: Bit Operators 197

Appendix E: Gnuplot Scripts for Selected Data 198

Appendix F: Result of Reading Accuracy 202

Appendix G: Preparing Standard Solution 204

LIST OF TABLES

Table	Table's Title	page
2.1	Embedded Linux SBCs	12
3.1	The TCP socket system functions	35
3.2	List of several of Gnuplot's command	41
4.1	I/O location for ADC	60
4.2	Control register for ADC	60
6.1	Average of bit error reading of 0 V input from DC power supply	137
6.2	Average of bit error reading of SBC GND input	138
6.3	Reading accuracy result of first input	141
6.4	Average reading accuracy result of five inputs	141
6.5	Step resolution quantity for medium average reading	142
6.6	Comparison of TCP and UDP criteria	150
6.7	Duration taken to detect connection broken and reconnect using TCP	156
6.8	Duration taken to detect connection broken and reconnect using UDP	156
6.9	Data transmission performance of TCP and UDP	166

LIST OF FIGURES

Figure	Figure's Title	Page
2.1	Data acquisition system architecture ……	4
2.2	A simple DSP system ……	10
2.3	D-Type 9 pin RS232 serial port ……	16
2.4	D-Type 25 pin RS232 serial port ……..	16
2.5	Client-server relationships at application protocol ……	19
2.6	OSI model network layers and Internet protocol suite …….	19
3.1	PESS hardware design ……..……	27
3.2	PESS system's architecture ……..	27
3.3	PESS-n hardware design …….	28
3.4	PESS-n system's architecture ……	28
3.5	PESS-n software modules …….	29
3.6	Basic telephone network ……...	33
3.7	Basic socket architecture ……...	34
3.8	Flow chart of client-server over TCP protocol ……	35
3.9	Flow chart of client-server over UDP protocol ……	37
3.10	Accessing MySQL command line interpreters (CLI) ……..	40
3.11	Compilation steps using *chroot* method ……...	46
4.1	TS-5500 Single Board Computer …….	50
4.2	Rear panel of TS-5500 ……...	51
4.3	LM35DZ Temperature sensor……...	53
4.4	Light Diode Resistor (LDR) …….	53
4.5	Copper (Cu^{2+}) ion-selective electrode ……..	54
4.6	Ion-selective reference electrode ……..	54
4.7	The 4x4 matrix keypad ……	55
4.8	The 24x2 alphanumeric LCD panel ……...	55
4.9	The 9 pin null modem serial cable …….....	56
4.10	Setting wired network, *eth0* using DHCP mode ……...	57
4.11	Signal conditioning circuit block diagram ……	59
4.12	Signal conditioning circuit ……	59
4.13	Experimental setup for ion-selective electrodes using signal	

	conditioning circuit ……...	61
4.14	Loading the matrix keypad driver module ……...	62
4.15	Quick LCD installation procedure ……	63
5.1	PESS state diagram and running processes listing…….....	66
5.2	PESS-n state diagram ……	67
5.3	Software architecture of PESS system ……	68
5.4	Accessing shared memory segment ……...	68
5.5	Software architecture of PESS-n system ……...	69
5.6	The analog input processing stages …….….....	70
5.7	Algorithm to convert analog signal to digital signal through ADC ..	71
5.8	Flow chart of *sensor* process …….	72
5.9	Shared memory file descriptor code ……	73
5.10	Accessing ADC IO permission code ……	73
5.11	ADC control register, Hex 196 ……	74
5.12	Positive reading subroutine flow chart ……	75
5.13	Negative reading subroutine flow chart ……	76
5.14	Converting binary base value to decimal base value ……	77
5.15	Checking connection status ……..	78
5.16	Backup process flow chart ……	79
5.17	Codes to determine the sign bit ……	80
5.18	Codes to read and combine LSB and MSB readings ……	81
5.19	Example of LSB and MSB readings ……	81
5.20	Arithmetic shift to left operation and the LSB and MSB combination result ……	82
5.21	One's compliment and two's compliment codes ……...	82
5.22	Main flow chart of an interactive system ……	83
5.23	Flow chart of *System Info* subroutine ……	84
5.24	Flow chart *Sensor Reading* subroutine ……	84
5.25	Flow chart *Control System* subroutine ……	85
5.26	Write a status to enter runlevel 6 ……...	86
5.27	Write a status to enter runlevel 0 ……..	86
5.28	Algorithm to retrieve information of the PESS system …….	87
5.29	Algorithm to provide current sensor' readings for interactive system ……	87

5.30 Algorithm to control the PESS system 88
5.31 Flow chart of *client* process using TCP protocol 90
5.32 Flow chart to store successful connection status in TCP *client* 91
5.33 Flow chart to store failed connection status in TCP *client* 91
5.34 Flow chart to send backup file in TCP *client* 92
5.35 The *server* flow chart using TCP protocol 94
5.36 Flow chart to save data in TCP *server* 95
5.37 Write status for normal connection in TCP *server* 95
5.38 Write status for broken connection in TCP *server* 96
5.39 The *client* flow chart over UDP protocol 98
5.40 Read data subroutine flow chart in UDP *client* 99
5.41 Send data subroutine flow chart for normal connection in UDP *client* 99
5.42 Retry connects subroutine flow chart 100
5.43 Set network connection status as broken 101
5.44 Reset the broken status 101
5.45 Flow chart of sending backup data over UDP 102
5.46 Flow chart of *server* process in UDP protocol 104
5.47 Save data subroutine flow chart of UDP *server* 105
5.48 Set the broken status in UDP *server* 105
5.49 Set the normal connection status in UDP *server* 106
5.50 Checking *mysqld* deamon process status 107
5.51 The PHP script with MySQL commands 108
5.52 The algorithm to load data into database at PESS-Server 109
5.53 Algorithm to calculate *yesterday* and *S7day* parameter of *staticscript* 113
5.54 Code in *dynamicscript* to select Gnuplot script 116
5.55 Algorithm to calculate date parameter for *dynamicscript* 118
5.56 Algorithm for current reading Gnuplot script.....121
5.57 Code to plot graph for *yesterday* parameter of Iron sensor's data …121
5.58 Code to plot graph for all sensors for monthly parameter 122
5.59 Setting date and time format in Gnuplot script 122
5.60 Setting date format in Gnuplot script 122
5.61 Algorithm for selected reading Gnuplot script 123

6.1 1.0 V input from DC power supply …….. 126
6.2 Analog input reading verification output ……..127
6.3 The client-server layout in PESS-n ……...127
6.4 Result of data transfer verification testing using TCP protocol ……128
6.5 Result of data transfer verification testing using UDP protocol …… 129
6.6 Content example of table *info* ……130
6.7 Result for current reading date's parameters ……131
6.8 Determine plot mode for selected reading …… 132
6.9 Loading Gnuplot script file ……132
6.10 Graph example for previous seven days data ……133
6.11 0 V input from DC power supply ……..135
6.12 Reading of 0V input from DC power supply ……136
6.13 0V input from ground signal of SBC ……..... 137
6.14 Reading result of SBC GND input ……138
6.15 First input for reading accuracy testing ……140
6.16 Reading accuracy testing result of first input ……140
6.17 Average step resolution quantity …….. 142
6.18 The LDR reading ……...143
6.19 The temperature sensor' readings …… 144
6.20 Copper sensor' readings versus standard solution density ……145
6.21 Mercury sensor' readings versus standard solution density ……. 145
6.22 Chromium sensor' readings versus standard solution density ……..146
6.23 Iron sensor' readings versus standard solution density …….156
6.24 Sensor' readings (mV) versus standard solution mixture's density (ppm) …… 147
6.25 Hardware used in interactive systems……... 147
6.26 Interactive system menu selection …… 148
6.27 Current system's information results …… 148
6.28 Example of current sensor' readings ……149
6.29 Control selection menu …… 149
6.30 Start process automatically during boot up process ……. 150
6.31 Recovery time in data transmission using UDP protocol for 60 seconds broken duration ……152

6.32 Recovery time in data transmission using TCP protocol for 60 seconds broken duration ……153
6.33 Recovery time taken using TCP ……... 154
6.34 Recovery time taken using UDP ……...155
6.35 Time taken to detect broken connection using TCP and UDP …… 157
6.36 Time taken to reconnect for TCP and UDP ……157
6.37 Successful sending 50 KB backup data using TCP …….. 158
6.38 Successful sending 100 KB backup data using TCP ……159
6.39 Successful sending 1.6 MB backup data using TCP ……159
6.40 Successful sending 62 KB backup data using UDP ……..159
6.41 Failure of sending 67 KB backup data using UDP ……160
6.42 Readings in database before connection broken of TCP connection testing …… 161
6.43 Readings in database after 120 seconds broken of TCP connection testing …… 161
6.44 Quantity of data losses for short broken duration ……162
6.45 Readings in database before connection broken of UDP connection testing …… 162
6.46 (a) Readings in database after 10 minutes broken of UDP connection testing …… 163
6.46 (b) Next readings in database after 10 minutes broken of UDP connection testing ……163
6.47 Quantity of data losses for medium broken duration ……164
6.48 Comparison of quantity data losses between backup process …….. 165
6.49 Total recovery time taken to detect connection broken and reconnect ……166
6.50 Main page of DAS user interface ……..168
6.51 Data centre page ……169
6.52 Individual sensor selection page ……169
6.53 Result of Mercury sensor for *today* parameter ……. 170
6.54 Graph of previous 7 days reading for all sensors ……..171
6.55 Select start and end date ……172
6.56 Result of selected reading for all sensors ……..173
6.57 Mercury sensor' readings in selected reading mode ……174

LIST OF ABRREVIATIONS

2G	Second Generation
3G	Third Generations
ADC	Analog to Digital Converter
ARM	Advanced RISC Machine
ASIC	Application-Specific Integrated Circuit
AUV	Autonomous Underwater Vehicle
BER	Bit Error Ratio
BSD	Berkeley Software Distribution
CAMAC	Computer Automated Measurement and Control
CF	Compact Flash
CLI	Command Line Interpreter
COBOL	COmmon Business Oriented Language
COTS	commercial off-the-self
CPU	Central Processing Unit
DAC	Digital to Analog Converter
DAQ	Data Acquisition
DAS	Data Acquisition System
DAU	Data Acquisition Unit
DC	Direct Current
DCE	Device Communication Equipment
DECT	Digital Enhanced Cordless Telecommunications
DOS	Disk Operating System
DSP	Digital Signal Processing
DSSS	Direct Sequence Spread Spectrum
DTE	Device Terminal Equipment
EBX	Embedded Board eXpandable
EOC	End of Conversion
EOS	Embedded Operating System
EPA	Environment Protection Agency
FOTRAN	FORmula TRANslator
FPGA	Field Programmable Gate Array
FTP	File Transfer Protocol

FTT	Fast Fourier Transform
GCC	GNU Compiler Collection
GPIB	General Purpose Interface Bus
GPL	General Public License
GPS	Global Positioning System
HDL	Hardware Descriptive Language
HMI	Human-Machine Interface
HTTP	HyperText Transfer Protocol
I/O	Input / Output
IC	Integrated Circuit
IEEE	Institute of Electrical and Electronics Engineer
IR	Infra Red
ISO	International Organization of Standardization
LAN	Local Area Network
LCD	Liquid Crystal Display
LFS	Linux From Scratch
LSB	Least Significant Bit
LVDS	Low Voltage Differential Signalling
MAN	Metropolitan Area Network
MSB	Most Significant Bit
NI	National Instrument
OFDM	Orthogonal Frequency Division Multiplexing
OOP	Object-Oriented Programming
OS	Operating System
OSD	Open Source Definition
OSI	Open Systems Interconnection
OSS	Open Source Software
PC	Personal Computer
PCB	Printed Circuit Board
PCI	Peripheral Component Interconnect
PESS	Portable Embedded Sensing System
PESS-n	Portable Embedded Sensing System – Network mode
PHP	PHP: Hypertext Preprocessor
PID	Process Identifier

ppm	part per million
RAM	Random Access Memory
RISC	Reduce Instruction Set Computing
RTO	retransmission timeout
RTT	round-trip time
rttvar	smoothed mean deviation RTT estimator
SBC	Single Board Computer
SCP	Secure Copy
SPI	Serial Parallel Interface
srtt	smoothed RTT estimator
SSH	Secure Shell
SSL	Secure Socket Layer
Tcl	Tool command locator
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol / Internet Protocol
TLS	Transport Layer Security
TS	Technologic Systems
UDP	User Datagram Protocol
UEI	United Electronic Industries
URL	Uniform Resource Locator
USB	Universal Serial Bus
VAX	Virtual Address Extension
VHDL	VHSIC Hardware Descriptive Language
VHSIC	Very High Speed Integrated Circuits
VME	VersaModule Eurocard
VMS	Virtual Memory System
WiFi	Wireless Fidelity
WiMax	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WMAN	Wireless Metropolitan Area Network
WPAN	Wireless Personal Area Network
WWAN	Wireless Wide Area Network
WWW	World Wide Web

SISTEM PEROLEHAN DATA BERASASKAN LAMAN SESAWANG MENGGUNAKAN KOMPUTER PAPAN TUNGGAL 32 BIT DAN GNU/LINUX

ABSTRAK

Kebanyakan kajian dalam Sistem Perolehan Data (DAS) sebelum ini menggunakan komputer sebagai platform alatan di mana ianya adalah besar dan tidak bolehubah kedudukan. Dalam kajian ini, Sistem Pengesanan Terbenam Bolehubah (PESS) telah dibangunkan menggunakan alatan sistem terbenam komersial sedia-ada (COTS). Teknologi terkini COTS lebih fleksibel dari sudut bolehubah kedudukan, bolehubah saiz dan bolehubah konfigurasi. PESS dibangunkan dengan integrasi antara Komputer Papan Tunggal (SBC) model TS-5500 sebagai teras unit pemproses, papan kekunci, paparan LCD dan sensor-sensor yang dilekatkan pada litar antaramuka. Pembangunan perisian pada PESS menggunakan sumber terbuka GNU/Linux yang membenarkan pengubahan pada perpustakaan dan pemacu alatan. PESS telah dikembangkan dengan menghubungkan kepada pelayan melalui rangkaian TCP/IP bagi memenuhi keperluan simpanan data yang lebih besar. Sistem dua-nod jenis rangkaian ini dinamakan PESS-n yang terbahagi kepada PESS dan PESS-Server. PESS-n dilengkapi dengan kaedah pembetulan-ralat yang boleh mengurangkan kehilangan data semasa kegagalan rangkaian, selain bertujuan menghantar data salinan-penduaan. Suatu antaramuka telah dibangunkan pada PESS-Server bagi membenarkan data yang diperolehi dipersembahkan dalam bentuk grafik disamping sebagai platform perkongsian maklumat. Purata ralat bit yang diperolehi pada perolehan masukan analog adalah konsisten dan tidak melebihi tiga unit resolusi. Bagi analisis prestasi penghantaran data melalui rangkaian, dua protokol telah dinilai iaitu Protokol Kawalan Penghantaran (TCP) dan Protokol Bentukdata Pengguna (UDP). Walaupun UDP memberikan masa pemulihan yang lebih cepat, namun ia hanya mampu menghantar data sehingga 64 KB sahaja. Manakala TCP pula berjaya menghantar data yang bersaiz besar sehingga 1.6 MB yang menyamai data selama 14 hari. Daripada analisis, diketahui bahawa kehilangan data menggunakan TCP dan UDP adalah sangat kecil (2-3 unit data).

WEB-BASED DATA ACQUISITION SYSTEM USING A 32 BIT SINGLE BOARD COMPUTERS AND GNU/LINUX

ABSTRACT

Most of the previous research in Data Acquisition System (DAS) had used Personal Computers (PCs) as hardware platform which were bulky and not portable. In this research, a Portable Embedded Sensing System (PESS) has been developed using commercial off-the-shelf (COTS) embedded system. Current COTS technology provides more flexibility in term of portability, scalability and configurability. PESS is made-up of an integration of TS-5500 Single Board Computer (SBC) as its computing core, a matrix keypad, a LCD display and sensors that attached to the interfacing circuit. As for the software part, PESS uses the open source GNU/Linux that allows modification to its libraries and device drivers. PESS has been extended by enabling a TCP/IP network connection to a server in order to accommodate bigger data archiving. These two-nodes network-based system is called PESS-n which divided to PESS and PESS-Server. PESS-n is equipped with an error-correction mechanism that can minimize data loses in the event of network failures, in addition to sending the backup data. A user interface has been developed at PESS-Server side to allow the acquired data to be presented in graphical format as well sharing of data via Internet. On the analog acquisition, not more than three steps resolution for the average bit of error has consistently been achieved. As for the network data transfer performance, both Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) have been evaluated. While UDP provide faster recovery times during simulated network failures, it has limitation up to 64 KB of data payload only. As for the TCP, it has been able to transmit large data up to 1.6 MB which equal to fourteen days of data. It is found that the data losses of TCP and UDP are very small (2-3 unit of data).

CHAPTER 1

INTRODUCTION

1.1 Overview

Data acquisition is the process of bringing a real-world signal, such as a voltage, into the computer, for processing, analysis, storage or other data manipulation (Rongen, n.d.). Generally, Data Acquisition Systems (DAS) are used to electronically monitor or gather data from the external physical environment (Ng, 1994). DAS normally consists of three elements: acquisition hardware, input and storage/display unit. The acquisition hardware plays a vital role in influencing the performance of DAS. Most of the previous research has used Personal Computer (PC) as the acquisition hardware. The trend was then changed from standard PC to high speed PC to provide better performance in terms of data processing and data transferring.

The embedded processor board has become a new alternative platform for DAS application. Several embedded processor board used as acquisition hardware are microcontroller, Field Programmable Gate Array (FPGA), Digital Signal Processor (DSP) and Single Board Computer (SBC). The microcontroller is the most popular platform for small and simple application because of its low cost. Some developments use FPGA as Data Acquisition Unit (DAU). FPGA allows modification of internal logic circuitry without touching hardware component. The DSP board is mostly used in applications that handle real-time computation process.

The other current trend on embedded technology application is the Single Board Computer (SBC). One major advantage of using an SBC is that it can handle multitasking processes since it run with a modular Operating System (OS). The development can be done using high level language such as C, Java and Perl which are widely used, flexible and have a lot of support from the open source community. However, the key to select a suitable processor board depends on the purpose of its application so that the optimum