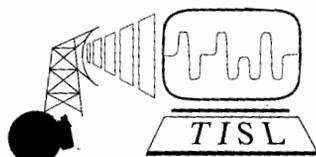


ATM Cell Level Measurement Capabilities On An OC-12c ATM/SONET Gateway

Darren S. Braun
Joseph B. Evans
Douglas Niehaus

TISL Technical Report TISL-9770-38

January 1996



Telecommunications and Information Sciences Laboratory
The University of Kansas Center for Research, Inc.
2291 Irving Hill Road Lawrence, Kansas 66045

Acknowledgments

I would like to thank Dr. Raymond Dean for his encouragement and guidance in introducing me to digital designs and computer hardware. I want to thank committee chair Dr. Joseph Evans for having confidence in me and assigning me the task of the Traffic Measurement chip. I would like to express thanks to committee member Dr. Doug Niehaus for his help and opinions in the development of the Traffic Measurement chip and to Dr. Victor Frost for giving me the employment opportunity to work on the MAGIC project.

Lastly, I want to thank Hugo Uriona and Benjamin Ewy for their invaluable help and for making me one of the Hardware guys. I commend them for their steadfastness and dedication in completing one of the first ATM switch components to operate in the OC-12 class with a bandwidth of 622 Mb/s.

Abstract

The design and implementation of a Traffic Measurement chip residing on an OC-12 ATM/SONET Gateway is presented in this paper. The Traffic chip allows ATM cell level measurements at the full rate of 622 Mb/s, providing interarrival times, time series and an insight into traffic burstiness. The provision of a single Xilinx chip on the Gateway dedicated to measurements presents the opportunity of a un-impinged look at cell behavior inside the switch. Additionally, the Traffic chip can provide AAL5 level measurements observed at the cell level.

Herein, two applications of the Traffic Measurement chip are presented. The first is the verification of a real time UNIX system designed to be an ATM reference traffic source. The reference traffic source generates traffic according to various probabilistic distributions believed to be representative of real traffic. The second application is the investigation of flow control mechanisms on various host workstation ATM cards. Recently in the ATM community, there has been debate on the performance issues of various types of flow control. Flow control can be subject to scheduling delays in an operating system, thus making a source bursty and causing congestion in a network. The Traffic chip exposes the real behavior of pacing and flow control on streams as they exist inside the ATM switch.

Contents

1	Introduction	1
1.1	Motivation for the Traffic chip	1
1.2	Contribution	2
1.3	Outline	3
2	Background	4
2.1	Asynchronous Transfer Mode (ATM)	4
2.2	SONET	7
2.2.1	SONET Frame Format	7
2.3	AN2 Switch Overview	9
2.4	Gateway Overview	10
2.4.1	Transmit Section	11
2.4.2	Receive Section	12
2.4.3	Support Section	12
2.5	Line Card Processor	13
3	Standard Version of the Traffic Chip	14
3.1	Specification	14
3.2	Functional Description	14
3.3	Signals	17
3.4	Load and Control Section	18
3.4.1	Module LCPHeaderEnable	18

3.4.2	Module HeaderRegister	21
3.4.3	Module Control	21
3.4.4	Module InputRegister	22
3.4.5	Most and Least Significant SixteenBitRegisters	22
3.5	FIFO Section	22
3.5.1	Module FIFOStateMachine	22
3.5.2	Module ThirtyTwoBitMux	23
3.5.3	Module ThirtyTwoBitRegister 2 and 3	23
3.5.4	Module TriStateBus	23
3.5.5	TriStateIOB	24
3.6	Receive Probes	24
4	Run Length Encoded Version of the Traffic Chip	25
4.1	Specification	25
4.2	Functional Description	27
4.3	Load and Control Section	30
4.3.1	Module Control	30
4.3.2	Module InputRegister	30
4.3.3	Module SixteenBitRegister	31
4.3.4	Module SixteenBitMux	31
4.4	FIFO Section	31
4.4.1	Module FIFOStateMachine	31
4.4.2	Module TriStateBus	32
5	Design Verification	33
5.1	Digital Simulation	33
5.2	Hardware Testing	34
5.3	Findings	35
6	Data Retrieval and Operation	37

6.1	Chip Operation	37
6.1.1	trafficvc(vc,stream)	37
6.1.2	traf_pulse(stream, time)	38
6.1.3	enable_traffic(stream)	38
6.1.4	disable_traffic(stream)	38
6.2	Raw Cell Traffic Capture	38
6.2.1	Setdev	39
6.2.2	Readtraffic	39
6.2.3	Statusdev	39
6.3	Post-processing	40
6.3.1	Run Length Processing	41
6.3.2	Standard Processing	43
6.4	VC setup	44
7	Verification of ATM Reference Traffic Source	46
7.1	Description of system	46
7.2	Method	46
7.3	Evaluation	48
7.4	Further Evaluation	54
7.5	Conclusion	54
8	Examination of Flow Control	55
8.1	Method	55
8.2	DEC OTTO Cards	56
8.2.1	DECstation 3000 model 600, "Alpha"	57
8.2.2	DECstation 5000 model 240	61
8.3	Fore SBA Cards	65
8.4	Examination of ATMTIMES on an Alpha	68
8.5	Conclusions	69

9 Conclusion	72
9.1 Summary and Conclusions	72
9.2 Future Work	72
A	76

List of Tables

2.1	Payload Type Table	5
3.1	Traffic Chip Address Table	21
6.1	Traffic Chip Probes	38

List of Figures

2-1	ATM Cell Format	5
2-2	STS-1 SONET Frame	8
2-3	STS-3 SONET Frame	8
2-4	622 Mb/s Line Card	10
2-5	ATM/SONET Gateway Architecture	11
3-1	Traffic Measurement Cell	15
3-2	Traffic Measurement Payload Word Configuration	16
3-3	Traffic Measurement Load and Control	19
3-4	Traffic Measurement FIFO	20
3-5	TriState IOB Enable Timing	24
4-1	Traffic Measurement Payload Word Configuration	25
4-2	Traffic Measurement Payload Encode Word Configuration	26
4-3	Traffic Measurement Cell, RLE Version	26
4-4	Traffic Measurement Load and Control	28
4-5	Traffic Measurement FIFO	29
7-1	ATM Reference Traffic Source, System Design	47
7-2	AAL5 Header Interarrivals for T=4000uS	49
7-3	AAL5 Trailer Interarrivals for T=4000uS	49
7-4	AAL5 Header Time Series for T=4000uS	50
7-5	AAL5 Trailer Time Series for T=4000uS	50
7-6	AAL5 Header Interarrivals for T=8000uS	51

7-7	AAL5 Trailer Interarrivals for T=8000uS	52
7-8	AAL5 Header Time Series for T=8000uS	52
7-9	AAL5 Trailer Time Series for T=8000uS	53
8-1	Permanent Virtual Circuit Configuration	56
8-2	Histogram of cell interarrival times, pacing @25 Mb/s	58
8-3	Histogram of cell interarrival times, pacing @25 Mb/s, all data . .	58
8-4	Time series of cell arrival times, pacing @25 Mb/s	59
8-5	Time series of cell arrival times, pacing @25 Mb/s, all data	59
8-6	Time series of cell arrival times, pacing @25 Mb/s	60
8-7	Time series of cell arrival times, pacing @25 Mb/s, all data	60
8-8	Histogram of cell interarrival times, pacing @22.6 Mb/s	62
8-9	Time series of cell arrival times, pacing @22.6 Mb/s	62
8-10	Time series of cell arrival times, pacing @22.6 Mb/s	63
8-11	Time series of cell arrival times, pacing @22.6 Mb/s	63
8-12	Time series of cell arrival times, pacing @22.6 Mb/s	64
8-13	Histogram of cell interarrival times, PCR = 25 Mb/s	66
8-14	Histogram of cell interarrival times, PCR = 25 Mb/s	66
8-15	Window Averaged Usable Cell Rate vs Arrival Time,PCR = 25 Mb/s	67
8-16	Histogram of cell interarrival times, PCR = 25 Mb/s	67
8-17	Window Averaged Usable Cell Rate vs Arrival Time,PCR = 25 Mb/s	68
8-18	Rate vs time plot of ABR stream, observed by Traffic chip	70
8-19	Rate vs time plot of ABR stream, observed by atmtimes	70

Chapter 1

Introduction

1.1 Motivation for the Traffic chip

Given that a network obeying a certain protocol exists, the measurement of traffic in that network can be done in two ways. One could use a network analyzer to examine the characteristics of a link in a network. This involves physically tapping onto a line and sampling the line. Knowing the protocol the network uses, the sampled data can be examined and characterized. Another method involves the use of a computer that is a node on the network. The data obtained in this case is typically the traffic intended for that machine.

Let us clarify the problem by specifying the protocol and the type of transport media. The protocol is ATM (Asynchronous Transfer Mode) over SONET (Synchronous Optical Network) and the physical media is optical fiber. There are problems with both measurement techniques when applied to the specific job of determining ATM cell arrivals and departures. Let us also specify that point of interest to examine, is the point where various streams of traffic incur congestion and regulation - inside the ATM switch.

The problem with a network analyzer is that the OC-12 rate equipment required to perform the task is expensive and a physical tap into the fiber link can only be done at the end of a fiber where it either connects to a host or switch. Note, the point of examination is on an I/O boundary, and not inside the switch.

The problem with a computer gathering the data is the bandwidth of ATM/SONET is quite high, 155 Mb/s, and the arrival of cells at this rate can be on the order of 1 cell every 2600nS. Since the operating system on most high performance workstations is multitasking, the timestamp of a cell arrival is subject to scheduling delays. The delay is variable depending upon the number of jobs a system must currently address. While useful data can be gathered, the measurement point still remains at the host.

The design of the KU OC-12 Gateway provides a chip which can examine traffic as it passes through the Gateway. In an ATM/SONET network, a link is made from a host to an appropriate port on an ATM switch. The ATM switch serves to channel ATM cells from an arrival port to a destination port based on cell header information. The channel can be setup to pass through any card installed in the switch. When the Gateway resides in the switch, measurements are possible inside the switch with the Traffic chip. The traffic report accurately portrays the relative positions of cells in a stream giving a true indication of traffic statistics.

1.2 Contribution

The original proposal for the Traffic Chip was laid out by Joe Evans, Gary Minden, Victor Frost and Murat Bog. A rough prototype of the Traffic chip was originally designed by Srinu Seetharam and later revised by myself and Scott Shumate. The complete and working revision of the Standard version of the Traffic chip including synthesis was performed by myself which includes all of the hardware level debugging. The concept of the Run Length Encode version was proposed by Joe Evans. The design and synthesis the Run Length Encode version of the Traffic chip was completed by myself which includes the all of the hardware level debugging.

1.3 Outline

The organization of this paper is as follows:

Chapter 1 provides an introduction into the motivation for the Traffic chip and explains the contribution made by the author.

Chapter 2 provides the necessary background for understanding the Traffic chip. This includes a description of the protocols and the hardware on which the Traffic chip exists.

Chapter 3 provides the specification, description and design of the initial Traffic chip called the Standard Traffic Version.

Chapter 4 provides the specification, description and design of the improved Traffic chip called the Run Length Encode Version.

Chapter 5 describes the methods of verification performed on the Traffic chip designs.

Chapter 6 describes the operation of the chip. Additionally, it provides the "man" pages for the support software that allows cell capture and data post processing.

Chapter 7 describes an application of the Traffic chip where the attention given to the data was at the packet level. The project was the verification of a real time ATM reference traffic source.

Chapter 8 describes an application of the Traffic chip where the attention given to the data was at the cell level. The project was the examination of flow control mechanisms in host ATM peripherals.

Chapter 9 provides a conclusion to the design of the Traffic chip and the data it generated.

Chapter 2

Background

2.1 Asynchronous Transfer Mode (ATM)

ATM(Asynchronous Transfer Mode) is a transfer mode that has been developed to accomplish the Broadband Integrated Services Digital Network or B-ISDN[11]. It is asynchronous from the standpoint that the recurrence of user information in cells is not necessarily periodic[1]. ATM is essentially a packet switching method of transporting fixed sized packets, called cells, through a network. It is also a connection oriented protocol meaning that a predefined path between end points is setup before any transmission takes place.

The cells are of the format described in ATM specification v3.0 consisting of 48 bytes of information and 5 bytes of header information totaling 53 bytes. Figure 2-1 provides a graphical representation of a cell defined for UNI connections.

All information for routing and management is contained in the cell header and consists of the generic flow control (GFC), Virtual Path Identifier (VPI), virtual channel identifier (VCI), payload type (PT), cell loss priority (CLP), and header error check (HEC).

GFC is meant to provide flow control at the customer site. The GFC is overwritten by ATM switches and does not remain constant through an end to end transmission. The VPI and VCI fields serve to route the cell. The actual number of bits needed for routing is negotiated between the user and the network.

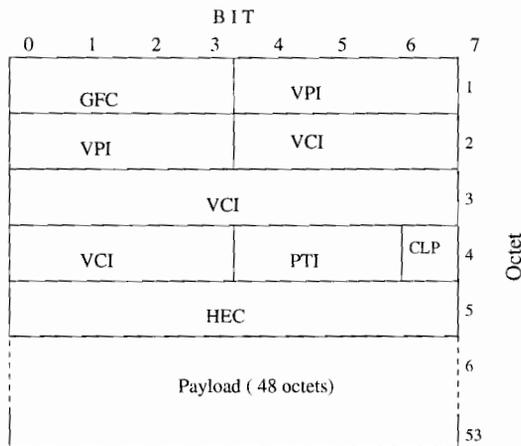


Figure 2-1: ATM Cell Format

Requirements for the Virtual Path Identifier and Virtual Circuit Identifier fields are that the VPI be adjacent to the VCI bit sequence. The least significant bit of the VPI is located at bit 3 in the second octet and for the VCI it is located at bit 3 of the fourth octet, as shown in Figure 2-1[1].

Table 2.1: Payload Type Table

PT	Significance
000	User data cell, no congestion SDU = 0
001	User data cell, no congestion SDU = 1 , AAL5 EOP
010	User data cell, congestion SDU = 0
011	User data cell, congestion SDU = 1
100	Segment OAM flow related cell
101	End to end OAM flow related cell
110	Resource management cell
111	Reserved

The PT or Payload Type field is a 3 bit field is used to indicate whether a cell contains user information or whether it contains Management information. Additionally, it can be used to indicate network congestion information. Table 2.1 describes the 8 different combinations and their meanings. The SDU or (Service Data Unit) is a unit of information whose identity is preserved from one end of a

layer connection to the other[1].

The CLP bit is used to indicate the loss priority of a cell. If network congestion occurs, it is sometimes necessary in an ATM network to discard cells. When a CLP bit is set, it indicates to the network management that this cell has no special priority and can be discarded. When the bit is not set, it indicates to the network that the cell has priority and should not be discarded, unless absolutely necessary.

The HEC field has two purposes: provides insight into corrupt headers and allows hardware to perform cell delineation. The HEC can provide a single bit error correction as stated in the ATM specification[1]. The Head Error Control code is contained in the last octet of the ATM cell header and is calculated for the first four octets of the cell header.

The ATM protocol stands as a layered architecture and provides a basis for a structured protocol. At the lowest level is the Physical Layer. The physical layer in an DEC AN2 uses SONET (Synchronous Optical Network) as the method by which to transmit bit information. Optical fiber provides the bandwidth capabilities necessary to implement a 622 Mb/s link in an AN2 switch.

The next layer, the ATM layer, serves to perform cell multiplexing, demultiplexing and routing of cells. Routing by switching is achieved by use of the VPI and VCI fields of the cell header. The information obtained in the header is used to look up address information for the outgoing link. The new replacement information is then put into the header and the cell is forwarded.

Above the ATM layer is the Adaptation layer which provides the services to assemble ATM cells from higher layers of the protocol stack such as IP. As well, the Adaptation layer provides services for the disassembly of cells and for cell loss. The most common types of adaptation layer are as follows:

- AAL1 is used to support Constant Bit Rate applications such as voice data.
- AAL3/4 is a combination of an AAL 3 and AAL 4 protocol. It provides services for both connection and connection less oriented protocols.

- AAL5 is a layer intended for use in computer applications where very large messages need to be sent. The segmentation layer is a single bit in the cell header signaling the end of the datagram field. The convergence layer is an 8 byte trailer consisting of a Length field and CRC. A CPI bit and User to User field exist but are not currently used[4]. The length field is used to indicate the number of bytes sent. The large 32 bit CRC is very robust and can serve to detect and correct errors.

In this paper, we will consider only the AAL5 which is intended for packet type traffic consisting primarily of data.

2.2 SONET

The physical layer defines the point where SONET can be used in association with ATM to provide an SDH based interface. The SONET standards provide a frame in which to transport ATM cells over optical fiber. The frame contains overhead for stations on the fiber. These stations are called the section, line and path. Additionally, the frame provides an envelope called the Synchronous Payload Envelope (SPE) where actual ATM cells are placed.

2.2.1 SONET Frame Format

The SONET standards define frames for various line rates. The size of a frame depends on the line rate. An STS-1 or Synchronous Transport Signaling 1 is a 51.84 Mb/s capacity standard. The entire STS-1 frame consists of 810 bytes and repeats at 125 μ s intervals thus yielding a nominal bit rate of 51.84 Mb/s. The Synchronous Payload Envelope (SPE) is the area reserved for transporting the data, also called payload, and contains 783 bytes per frame.

STS-1 frames can be multiplexed into higher rate signals denoted by the terminology STS-N, where N denotes the number of STS-1's that have been byte

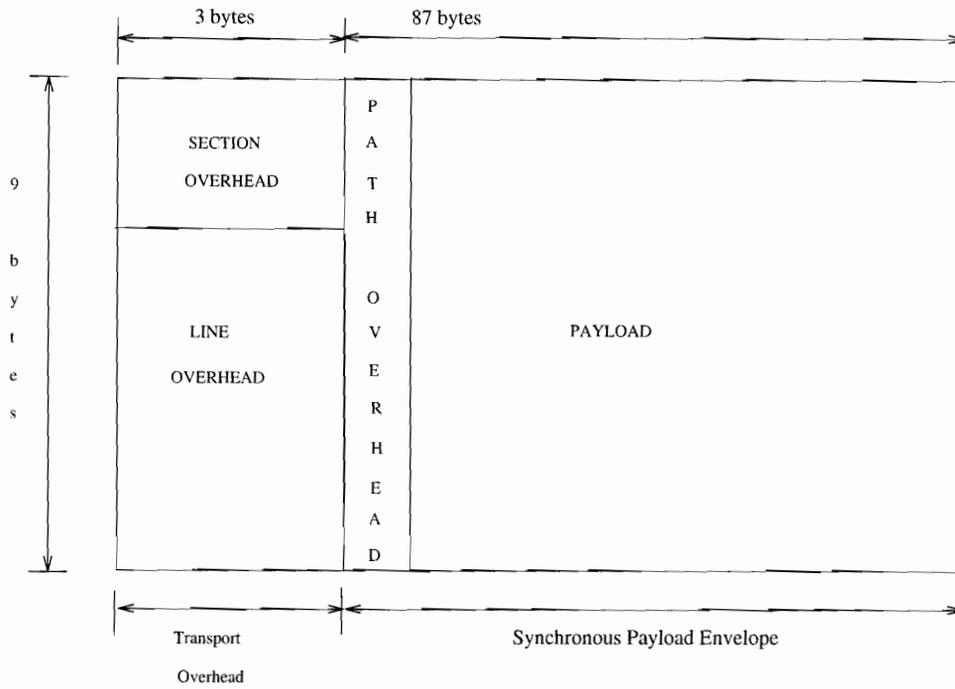


Figure 2-2: STS-1 SONET Frame

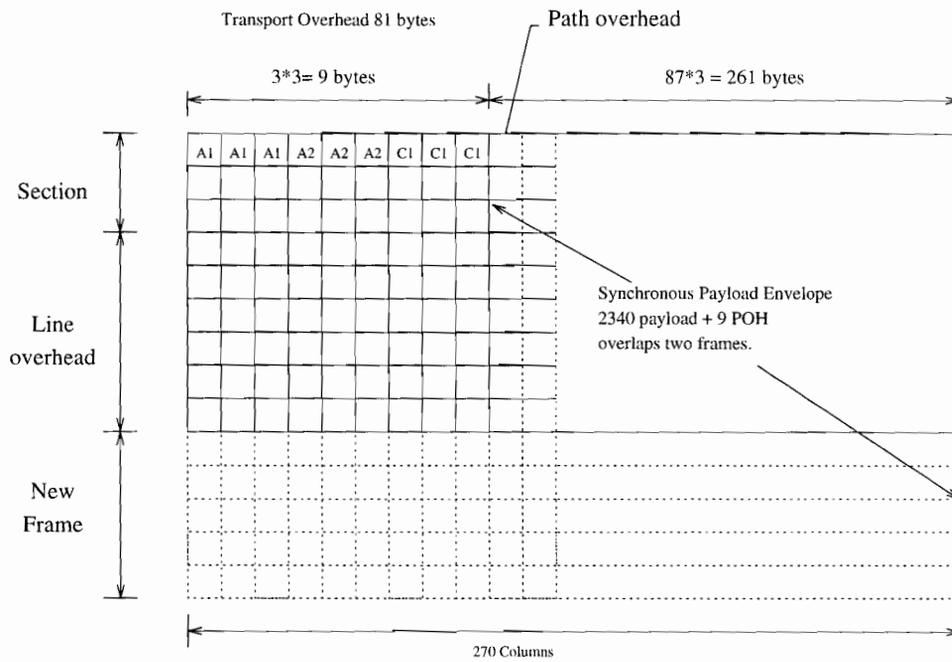


Figure 2-3: STS-3 SONET Frame

multiplexed together. The optical signal equivalent to STS-N is given the designation Optical Carrier (OC)-N and during transmission of frames, another frame of the same configurations follows every 125 μ s.

The STS-3 frame defines a rate of 155.52 Mb/s and has a payload size of 2340 bytes minus the path overhead. Since the frame occurs every 125 μ s, the capacity available to ATM after stripping SONET overhead is 149.760 Mb/s[1].

An STS-12 frame defines a rate twelve times the basic SONET rate which is 622.08 Mb/s. The capacity available to ATM after stripping SONET overhead is 600.768 Mb/s[8]. To make an STS-12 frame, 12 STS-1 or 4 STS-3 streams could be multiplexed together. Essentially The STS-12 frame is a scaled up version a STS-1 frame [3, 4].

2.3 AN2 Switch Overview

The AN2 is an ATM based switch developed by Digital's Systems Research Center. It consists of a 16 by 16 synchronous crossbar and uses input buffering. Each of the sixteen ports can accept one of two types of line cards: a quad-line OC-3 (155 Mb/s) or a single OC-12 (622Mb/s) card [15]. The cell data bus is 32 bits wide and operates on a 40 nS clock. The total bandwidth available per card is 32bits/40nS or 800 Mb/s. Since there are 16 cards, the aggregate bandwidth of the switch backplane is 12.8 Gbits/s.

Host machines can be connected to the OC-3 ports on the quad-line cards via fiber optic cable. The host machines generate ATM cells encapsulated in SONET OC-3 frames.

Each line card has three components:

1. An *input unit*, which receives SONET frames and strips them of overhead to expose ATM cells. The cells are buffered and later forwarded to the crossbar.

2. An *output unit*, which buffers ATM cells and creates SONET frames for fiber transmission.
3. A microprocessor, known as the LCP, which is responsible for managing the input and output units.

Each port or laser supports up to 32768 virtual circuits. In addition, the AN2 switch features no head-of-line blocking and hop-by-hop based flow control to guarantee no queue overflow[10].

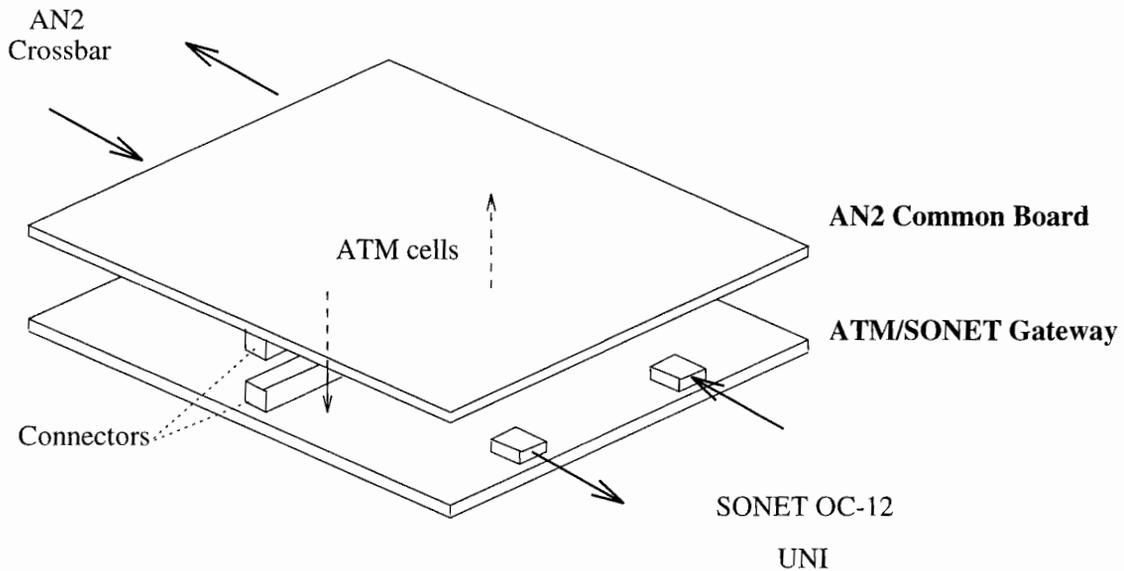


Figure 2-4: 622 Mb/s Line Card

2.4 Gateway Overview

The ATM/SONET gateway is a hardware device that connects the AN2 switch to a SONET based B-ISDN ATM network at 622 Mb/s[10]. The gateway card can operate in OC-12c or 4×OC-3 modes and features support chips for experimental pacing and performance measurement at the ATM cell level. This reconfigurable nature mandates the use of flexible logic components. The gateway is implemented

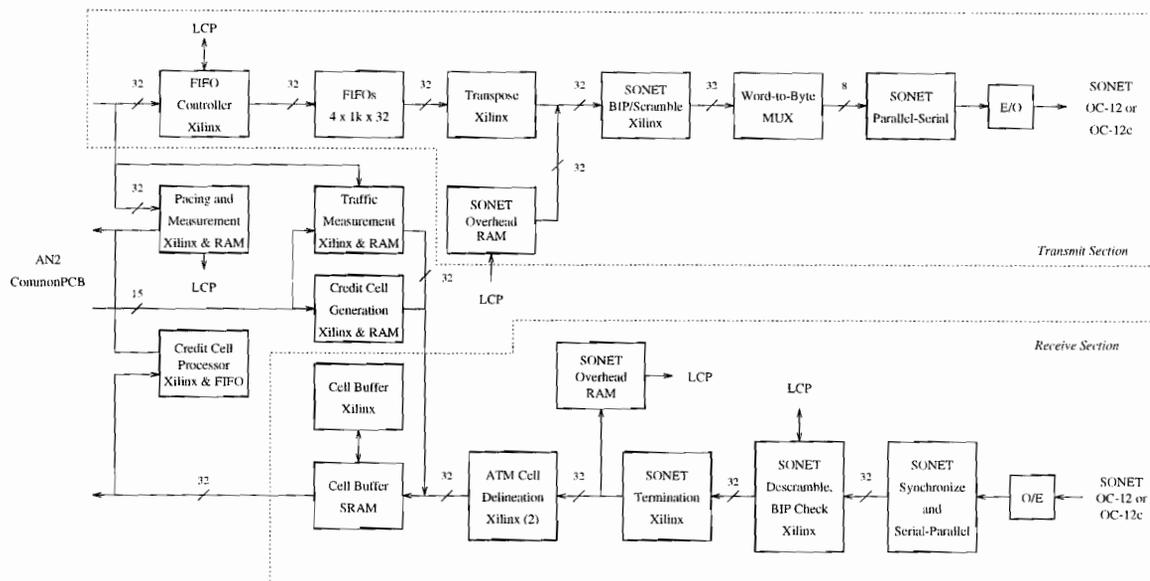


Figure 2-5: ATM/SONET Gateway Architecture

almost entirely in Xilinx FPGAs that are programmed according to the mode of operation determined at boot time. The gateway is actually one of two line boards, or daughter cards, that connects to the AN2 common board, or mother board. The two boards, shown in Figure 2-4, are connected by an 80-pin interface to form a single line card that plugs into an AN2 switch port[9].

A block diagram of the gateway card is shown in Figure 2-5. There are three sub-systems: the transmit section, the receive section, and the support system.

2.4.1 Transmit Section

The transmit section provides the assembly of SONET frames from payload and overhead and contains a high speed optical transmission section. Cells arrive from the common board are placed into high performance FIFO memories. Additionally, SRAMs contain SONET overhead information. The Transpose essentially creates SONET frames in the form of a 32 bit word stream containing SONET overhead and ATM cells. Next, the stream is scrambled according to SONET standards and the word stream is multiplexed into a high speed bit stream. The

last component is the optical-to-electrical converter or “laser”. The transmit laser of the Gateway is typically connected to Network SONET Termination gear. It can also be connected to the receiver of the Gateway for fiber loop-back testing.

The logic in the transmit section is created out of three Xilinx XC3195 FPGAs.

2.4.2 Receive Section

The receive section provides services for the disassembly of SONET frames and the extraction of ATM cells. A SONET stream arrives from a high speed optical receiver and is converted to an electrical signal. The bit stream is demultiplexed into a 32 bit SONET word stream which encounters the SONET Descrambler and Terminator. The SONET Section, Line and Path overhead is stripped and stored in SRAMs. ATM cell delineation is performed next, so that cell boundaries can be determined. The delineated cells are placed in SRAMs and forwarded to the motherboard by the Sram Controller. The motherboard or common board determines where the cells are destined depending upon the virtual circuit table. Cells either go out to the crossbar or are sent back to the Gateway. The logic in the receive section is created out of five Xilinx XC3195 FPGAs.

2.4.3 Support Section

The support chips provide capabilities such as measurement, flow control and pacing. While the support chips are not essential for sending traffic through the Gateway, they are accessories that make it unique. The Pacing Measurement and I/M controller are designed for rate based flow control. The BatchAck chips are designed for credit based flow control. Lastly, the Traffic Measurement chip allows the Gateway to generate cell level measurements.

The logic of the support section is created out of five Xilinx XC3195 FPGAs.

2.5 Line Card Processor

A Line Card Processor (LCP) interface is provided for the gateway in order to manage the transmit, receive and support systems. The LCP provides these services as well as the setup of circuits and the management of cells in and out of the Gateway. The LCP is composed of a general purpose RISC processor and support chips[10].

Chapter 3

Standard Version of the Traffic Chip

3.1 Specification

It is proposed that the Gateway Card will gather a VCI sequence in the transmit and receive cell streams. This provides the Gateway with cell level data measurement capabilities. The gathered VCI information will be transmitted on a specified VC back to a host for permanent storage. Additionally, AAL5 packet level sequence information will be gathered, providing interarrival statistics at the Adaptation layer[6]

3.2 Functional Description

The Traffic Measurement chip provides three probes into the transmit and receive ATM streams on the 622 Mb/s Gateway Card. Measurements are made in the form of recording ATM headers of the traffic on the above mentioned streams. The header information can be used to calculate per VC cell interarrival times, average rate, peak rates and burstiness. Additionally, AAL5 packet level header and trailer interarrival times and packet lengths can be observed.

The chip serves to probe two locations on the receive stream qabus, and one location on the transmit xData bus. The two positions on the qabus are located before the receive FIFO VRAM on the motherboard which corresponds to a latch

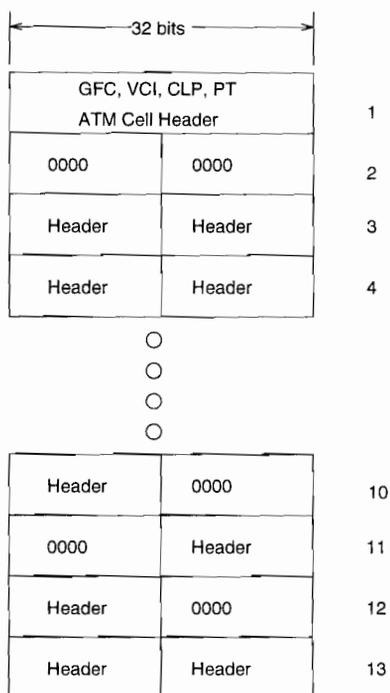


Figure 3-1: Traffic Measurement Cell

time of T8 and immediately after the FIFO VRAM which corresponds to T6. The T[12:0] designation is a convention describing word times for the AN2 line cards. An ATM cell is created out of 13 32 bit words. The word clock has a period of 40 nS and the cell occupies a total of 520 nS. Cells that arrive to the receive FIFO have come downstream from the high speed section. They have been striped of their SONET framing and sent through a Delineator to determine the cell boundaries. After a short time(520 nS) of being stored in Delineator SRAM, the cells are sent down to the motherboard where they reside in the receive FIFO VRAM. Cells leaving the FIFO are forwarded on to the crossbar destined for other cards. The probe on the xData bus observes traffic that comes from other cards across the crossbar, up through the motherboard and out the Gateway through the transmit section. Transmit data is latched on T2. Information generated by the chip exists in the form of a Traffic Measurement cell (see Figure 3-1), and is stored in local Delineator SRAM before being relayed to a host workstation.

The SRAM controller arbitrates requests for bus access when the Traffic chip has data to send. Note that this bus is shared by the Delineators whose output is real traffic.

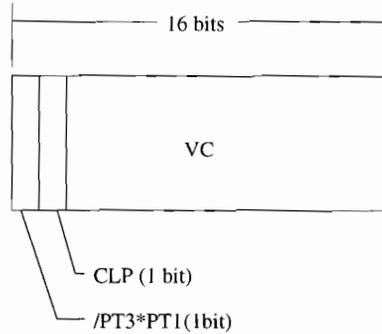


Figure 3-2: Traffic Measurement Payload Word Configuration

The data collected by the Traffic chip consists of 14 bits of VCI, the Cell Loss Priority bit and the reduction of Payload Type reflected in 1 bit, as seen in Figure 3-2(also see Table 2.1) Together, this forms a 16 bit half word, of which two can fit into one of the 12 32 bit words of the ATM Traffic Measurement cell payload. The VCI for the outgoing Traffic cell is written by the LCP on the LCP Data Bus.

The possibility of a cell not being valid on the xData bus, or not intended for the Gateway card brings the need for NULL VCI information to be placed in the payload in order to keep timing. The timing is inherent in the payload of Traffic cells and is explained as follows. A header (half word) designates a cell time. For a specific VC the count of the headers until the next occurrence of that VC provides the time between arrivals in ATM time slots. The AN2 word clock has a period of 40nS and AN2 ATM cells consist of 13 words minus the HEC. The result of the Traffic Measurement chip is an ATM cell.

3.3 Signals

The following signals are either available from or provided to the Traffic Measurement Chip

- AN2_CellSync_H IN : Is Asserted @ T10 every cell cycle and is used to get cell synchronization times.
- AN2_WordClkIn_H IN : AN2 word clock, period of 40nS
- LCP_Data_H(15 0) IN : Data interface to be able to write Traffic Measurement cell VCI and provides a way to turn chip on and off
- LCP_Address_H(3 0) IN : Address which is decoded to indicate that the corresponding data is a cell VC or other value that turns a given stream on and off
- VCI_Tx_H(13 0) IN : VCI from the XbarCellData stream available @ T2
- PT_CLP_Tx(3 0) IN : PT and CLP from XbarCellData stream available @ T2
- Outline_H IN : Signal that indicates ATM cells are intended for the Gateway to transmit
- XbarCell_Valid_H IN : Driven during T1 when valid cells are on crossbar
- CellForward_H IN : Asserted @ T6 to denote forwarded cell information is valid
- VCI_Rx(13 0) IN : VCI probe of the qabus providing two times to examine receive data , Rxo VCI @ T6, Rxi VCI @ T8
- PT_CLP_Rxo(3 0) IN : PT and CLP corresponding to VC data on qabus @ T8
- PT_CLP_Rxi(3 0) IN : PT and CLP corresponding to VC data on qabus @ T10
- CellValid_H IN : Asserted @ T8 to denote nonforwarded cell information is valid (before receive FIFO)
- LCP_CE_L IN : Asserted with LCP_Write_L
- LCP_Write_L IN : Asserted to latch address and data
- Request0_H IN : Asserted when the Measurement Stream 0 has a word ready
- Request1_H IN : Asserted when the Measurement Stream 1 has a word ready
- Request2_H IN : Asserted when the Measurement Stream 2 has a word ready
- ATM_Words(31 0) OUT : ATM Traffic Measurement Words

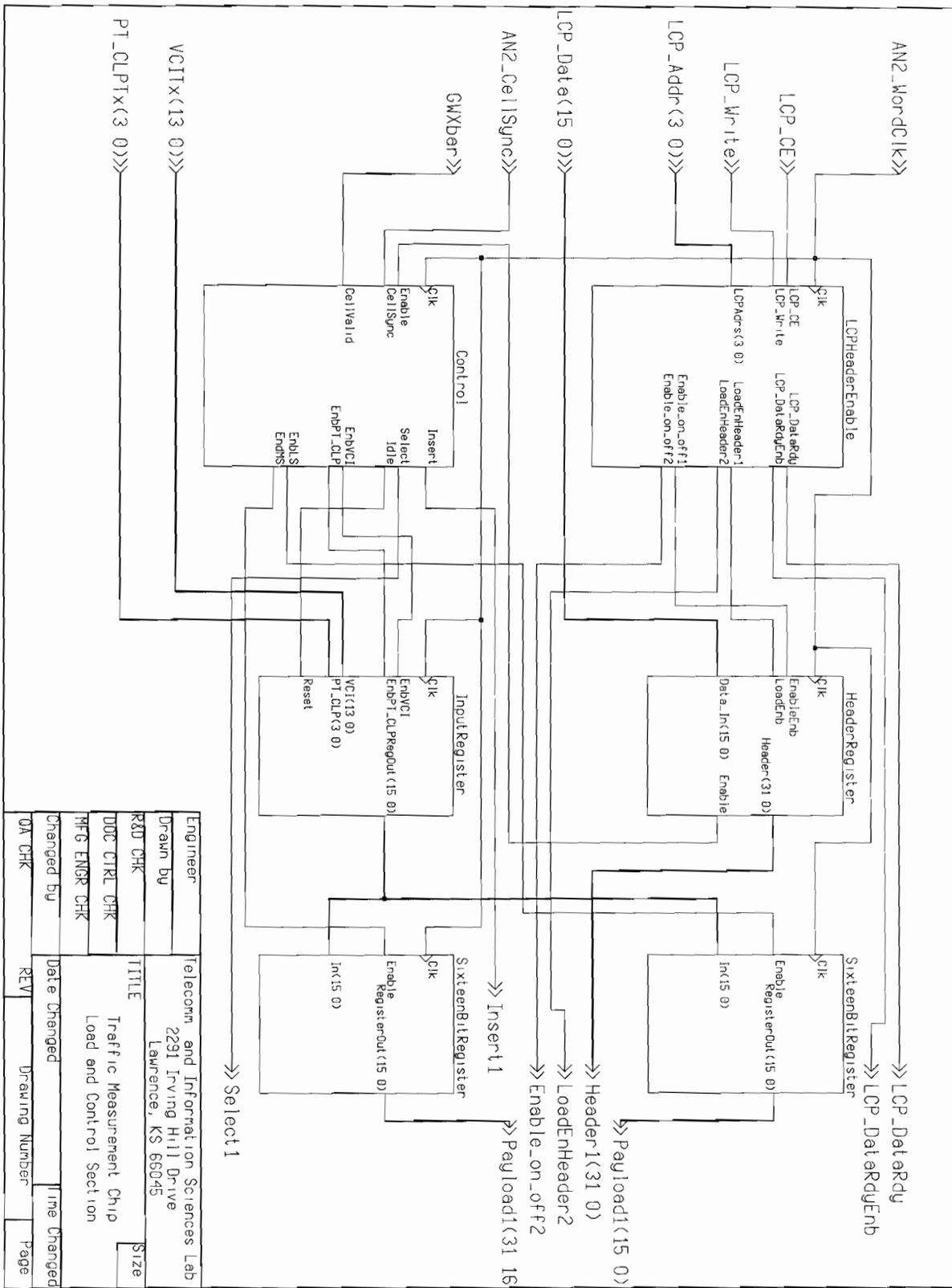
- LCP_DataRdy_H OUT : An acknowledge sent to LCP for receipt of LCP_Data
- Grant0_H OUT : Asserted by Sram Controller to signal Measure Stream 0 that it can drive the data word on bus
- Grant1_H OUT : Asserted by Sram Controller to signal Measure Stream 1 that it can drive the data word on bus
- Grant2_H OUT : Asserted by Sram Controller to signal Measure Stream 2 that it can drive the data word on bus
- Reset_L IN : Reset

The internal logic of the Traffic Measurement chip can be broken into several functions. These blocks are shown in Figure 3-4 and Figure 3-3 and will be described in turn.

3.4 Load and Control Section

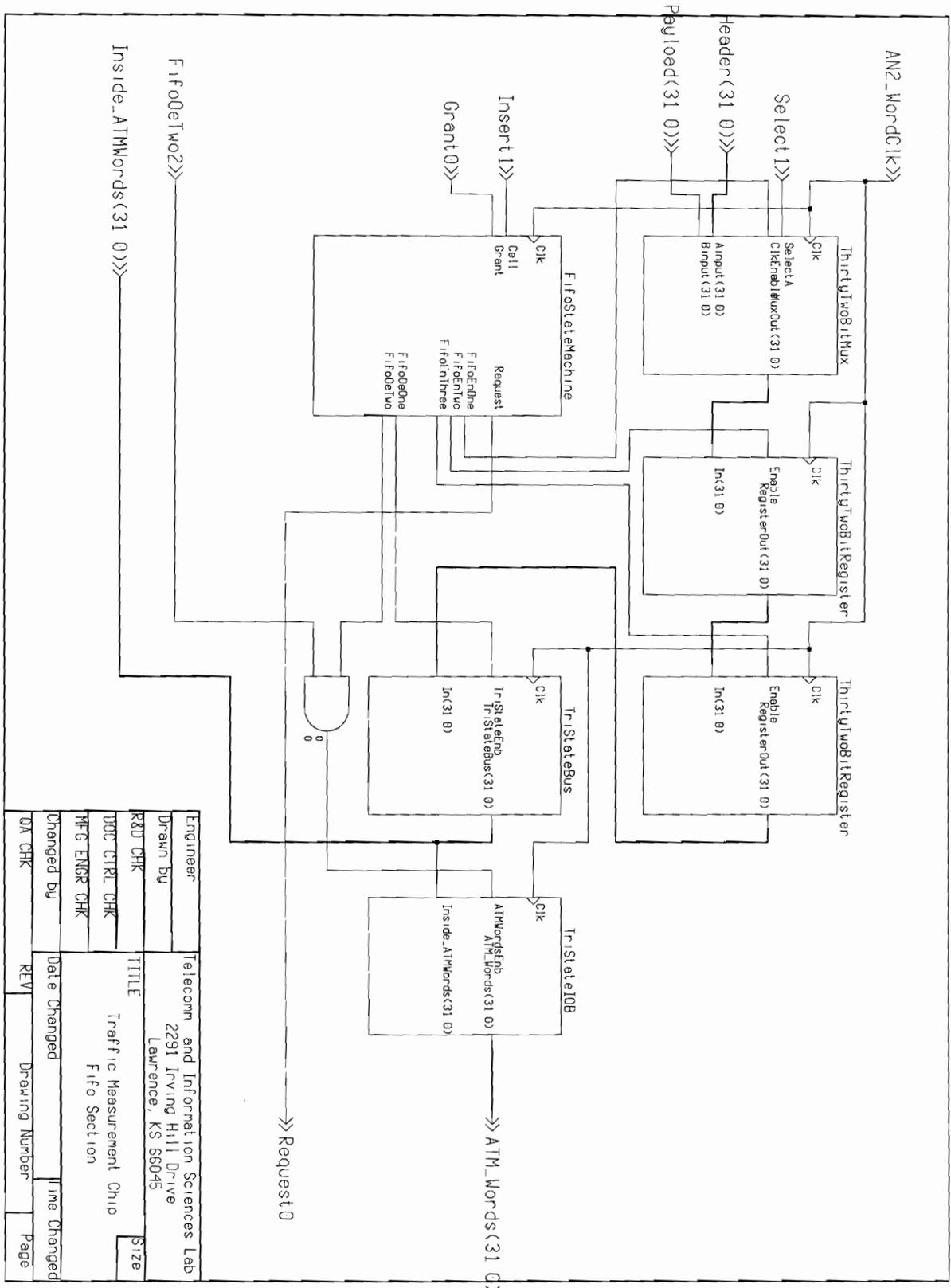
3.4.1 Module LCPHeaderEnable

Given an LCP address has been provided by the LCP, the LCPHeaderEnable module serves to decode the address. The LCP address, LCP chip enable, and LCP write enable are shared with the Descrambler and FIFO Controller. Therefore, the Traffic chip must decode the address and respond only to those intended for it. The Traffic chip address table shows the address ranges that are valid 3.1. A valid address means that data is present on the LCP data bus that corresponds to a one of the three probes. The LCPHeaderEnable module generates an enable signal and load signal to be used by the HeaderRegister. The enable signal tells the HeaderRegister when to latch bit 0 of LCP_Data and the load signal indicates when to load data from the LCP_Data bus. Additionally, the LCP_DataRdy indicates that the Traffic Measurement Chip has placed valid data on the LCP_Data bus.



Engineer	Telecomm and Information Sciences Lab	
Drawn by	2291 Irving Hill Drive Lawrence, KS 66045	
R&D CHK	TITLE	Size
DOC CTRL CHK	Traffic Measurement Chip	
MFG ENGR CHK	Load and Control Section	
Changed by	Date Changed	Time Changed
DA CHK	REV	Drawing Number
		Page

Figure 3-3: Traffic Measurement Load and Control



Engineer	Telecomm and Information Sciences Lab		
Drawn by	2291 Irving Hill Drive Lawrence, KS 66045		
R&D CHK	TITLE		Size
DOC CTRL CHK	Traffic Measurement Chip		
MFG ENDR CHK	Fifo Section		
Changed by	Date Changed	Time Changed	
DA_CHK	REV	Drawing Number	Page

Figure 3-4: Traffic Measurement FIFO
 CHAPTER 3. STANDARD VERSION OF THE TRAFFIC CHIP 20

Table 3.1: Traffic Chip Address Table

lcp addr offset	xilinx addr	operation
0x0C	0x3	Load Header 1
0x10	0x4	Load Header 2
0x14	0x5	Load Header 3
0x34	0xD	Enable stream 1
0x38	0xE	Enable stream 2
0x3C	0xF	Enable stream 3

Notes: - cached lcp base addr = 0x1EC38000 - uncached lcp base addr = 0xBEC38000 - chip enable (LCP_CE) signal is shared with Descrambler and FIFO Controller.

3.4.2 Module HeaderRegister

The basic function of the HeaderRegister is to produce the Header signal which becomes the VCI of the outgoing Traffic Measurement Cell. An enable signal is generated and passed to the Control register to indicate the chip should start measuring. The register simply latches input data during the LoadEnb signal.

3.4.3 Module Control

The Control module provides most of the signals for accessing cell stream data. The module houses a 4 bit counter and 5 bit counter which provide counts for the words of an ATM cell and the number of cells observed on the ATM cell stream. The Enable is used to create a start signal which is asserted for the entire length of the the Measurement Cell. This allows only complete cells to be emitted from the Traffic Measurement Chip. EnbLS and EnbMS are used to enable 16 bit registers. The signals indicate the most significant and least significant 16 bits of the 32 bit word payloads of the Traffic Measurement Cell. Enables alternate to create a 32 bit word of VCI/PT/CLP information. A Select signal is generated to allow a

selection between payload and Traffic cell header information at the input to the internal FIFO. Idle is generated to indicate that no valid information is available or intended for the Gateway Card during a cell time. An Insert signal is generated to signal the FIFOStateMachine to insert a word into the internal FIFO. EnbVCI and EnbPT_CLP are signals generated to indicate that the input data should be latched by the Input Register.

3.4.4 Module InputRegister

This module serves to obtain the cell stream header information which consists of 14 bits of VCI, the PT bits and CLP bit. One bit of the Payload Type is thrown away. The remaining 3 bits of PT and CLP are mapped into 2 bits. The resulting header information is 16 bits wide. The reset signal is used when no valid cell is observed in a cell time. This idle condition is indicated by resetting all D-FlipFlops resulting in a VC value of 0000.

3.4.5 Most and Least Significant SixteenBitRegisters

The 16 bit words from the Input Register are submitted to 16 bit registers for the creation of 32 bit word payload for the Traffic Measurement Cell. The EnbLS and EnbMS signals created by the Control Module select which of the 16 bit registers receive the Input Register contents.

3.5 FIFO Section

3.5.1 Module FIFOStateMachine

The state machine consists of logic derived from the Insert and Grant signals. When a valid word is inserted into the FIFO with the Insert signal, a Request is asserted. Upon a Grant, the state machine advances the 32 bit words residing

in the FIFO. The last word is advanced to the the TriStateBus, upon which the Output enable signals are asserted to send the word out on the ATM_Words bus. The result of the Traffic Measurement Chip is a Measurement Cell streamed out one word at a time.

The AN2 word clock has a 40nS/period, which is a clock "frequency" of 25 Mcycle/sec. A 16 bit piece of cell information is gathered every cycle. Twenty-four pieces of information fit into one measurement cell. Thus, one measurement cell is generated every 12.48 μ s and the expected receive cell rate at the host is 4.25 Mbytes/sec or 80128 cells/sec.

3.5.2 Module ThirtyTwoBitMux

A 32 bit mux is provided so that a selection can be made between the Header and Payload signals. For every 32 bit Header word selected, 12 32 bit Payload words are chosen.

3.5.3 Module ThirtyTwoBitRegister 2 and 3

Two 32 bit registers are provided which creates a small internal FIFO. A FIFO is needed in the chip because the Grant that advances the FIFO obeys a schedule where priority is given to other chips on the Gateway.

3.5.4 Module TriStateBus

A Tri-state bus exists inside the Traffic chip so that the information for two probes can be sent out one output port. A multiplexer would perform the same duty but occupy more configurable logic blocks.

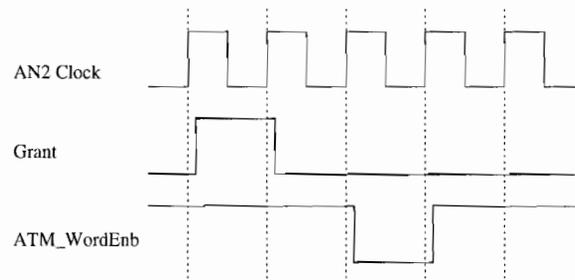


Figure 3-5: TriState IOB Enable Timing

3.5.5 TriStateIOB

The Tri-state Input/Output Block allows the final output word to be placed on the ATM_Word bus. The IOB passes a word according to the ATM_WordEnb signal. The enable essentially always follows the Grant signal for a given probe after one clock cycle as shown in Figure 3-5 [5].

3.6 Receive Probes

The schematics and blocks described above exist in the same chip for each of the two chosen probes. The only exceptions are the LCPHeader module and TriStateIOB which services both probes. Only two probes can be chosen to exist in the chip at any one time. The configuration of the Traffic chip is determined at compile time where the user is prompted for two probe selections. A traffic_configuration file is created every time the Modula-3 code is compiled. It describes the selections made for current version of the chip. Logic block placement inside the Xilinx is statically defined for the Traffic chip. It has been guaranteed that delays never exceed the 40 nS word clock boundary.

Chapter 4

Run Length Encoded Version of the Traffic Chip

4.1 Specification

The Traffic chip should stream data indicating cell arrival times. The chip should be capable of probing three locations on the Gateway reporting any cell VC intended for the Gateway. The chip should also include a method by which to report cell data in as little bandwidth as possible. The hardware needed should still fit into one Xilinx XC3195

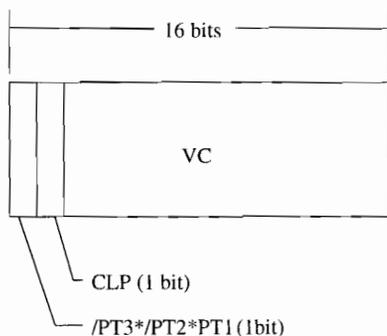


Figure 4-1: Traffic Measurement Payload Word Configuration

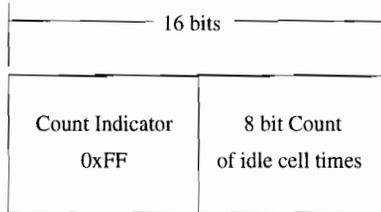


Figure 4-2: Traffic Measurement Payload Encode Word Configuration

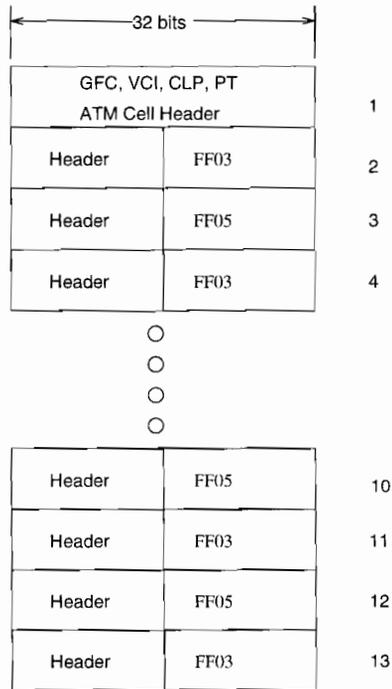
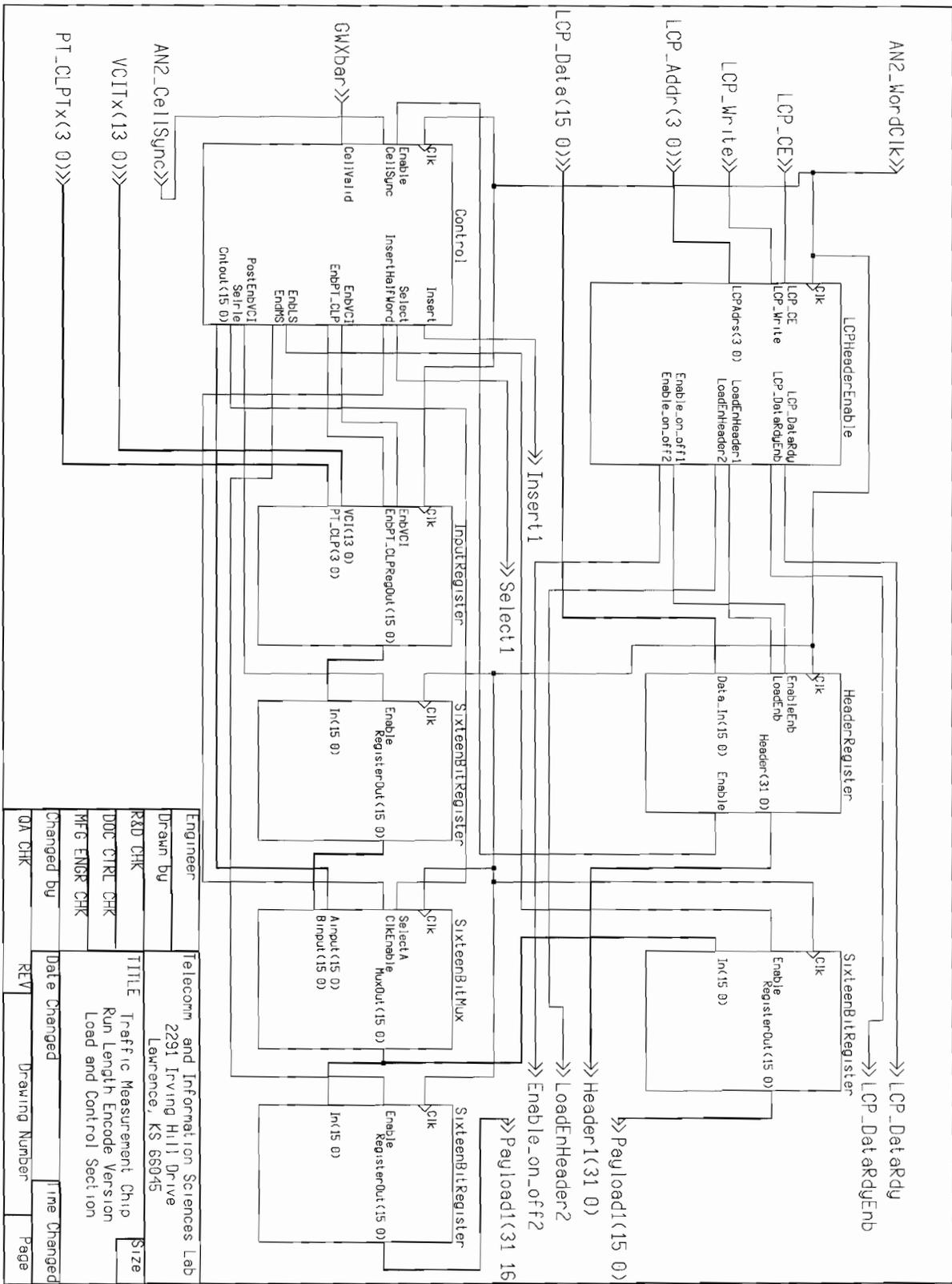


Figure 4-3: Traffic Measurement Cell, RLE Version

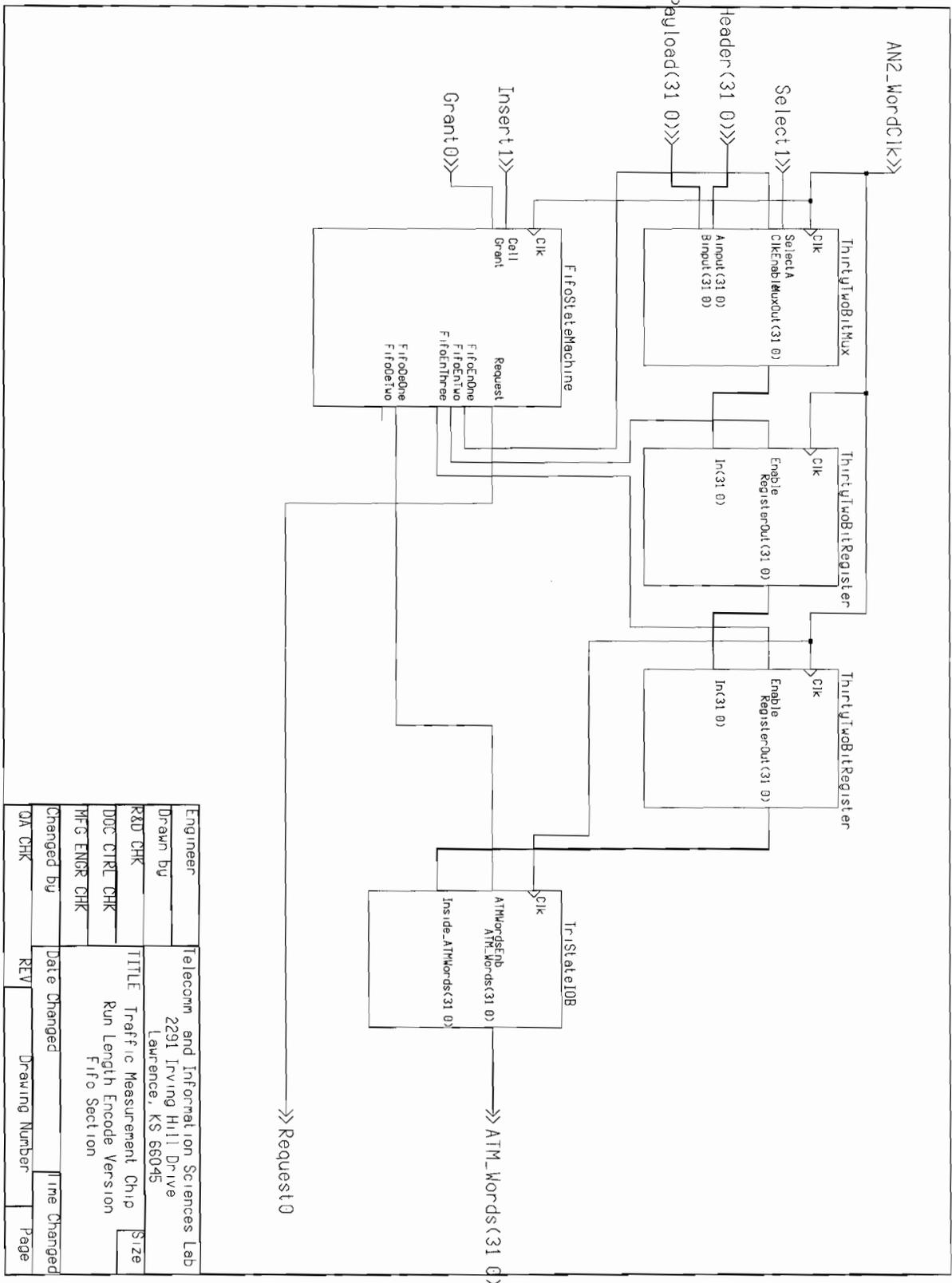
4.2 Functional Description

The proposed Traffic chip that completes the needs of the specification is essentially the same as the Standard version of the Traffic chip. The chip still samples the data at the proposed T cell times however the idle information in between cell interarrivals can be reported differently. When the Traffic chip is used to report information on low bandwidth streams the information in the measurement cells contains much idle information. Only when the Gateway bandwidth is fully used do we see little idle information in the measurement cells. Since many times single OC-3 streams are observed, a proposed Run Length Encoded chip was designed. It provides a count of all idle cell times observed. The count words have an indication byte in the most significant byte and an eight bit count in the least significant byte, see Figure 4-2. The indicator byte is 0xFF and chosen to be this because no VC is allowed to have a value of 0xFF00 (65280 decimal) or greater on the AN2. The VC report words are much the same as in the Standard Version as shown in Figure 4-1 however, the Payload Type mapping is done differently. The three bits of PTI are examined for the value 100. For OC-3 traffic on the Gateway, the bandwidth savings is considerable. A rough approximation that the Traffic Chip reports is 1 Mbyte/sec as opposed to 4.25 Mbytes/sec in the Standard version. The result of the Run Length version is a Traffic Measurement cell as shown in Figure 4-3. The modules of this version are much the same as those of the Standard version except for the following described in turn. Additionally, due to time constraints and available routing facilities, only one of three probes can be programmed into a Xilinx XC3195. The schematics for the improved chip are shown in Figure 4-4 and Figure 4-5.



Engineer	Telecom and Information Sciences Lab
Drawn by	2291 Irving Hill Drive Lawrence, KS 66045
R&D CHK	TITLE Traffic Measurement Chip
DOC CTRL CHK	Run Length Encode Version
MFG ENDR CHK	Load and Control Section
Changed by	Date Changed
QA CHK	REV
	Drawing Number
	Time Changed
	Page

Figure 4-4: Traffic Measurement Load and Control



Engineer	Telecomm and Information Sciences Lab
Drawn by	2291 Irving Hill Drive Lawrence, KS 66045
R&D CHK	TITLE Traffic Measurement Chip
DOC CTRL CHK	Run Length Encode Version
WFG ENDR CHK	Fifo Section
Changed by	Date Changed
DA_CHK	REV
	Drawing Number
	Time Changed
	Page

Figure 4-5: Traffic Measurement FIFO
 CHAPTER 4. RUN LENGTH ENCODED VERSION OF THE TRAFFIC CHIP 29

4.3 Load and Control Section

4.3.1 Module Control

The Control module provides most of the signals for accessing cell stream data. The module houses 4 bit, 5 bit, and 8 bit counters. The 4 bit counter provides counts to maintain the T designated cell times of an ATM cell. The 5 bit counter provides the number of halfwords observed in the Traffic Measurement cell payload. The 8 bit counter is used to keep count of the number of consecutive idle cell times observed. The Enable is used to create a start signal which is asserted for the entire length of the the Measurement Cell. This allows only complete cells to be emitted from the Traffic Measurement Chip. EnbLS and EnbMS are used to enable 16 bit registers. The signals indicate the most significant and least significant 16 bits of the 32 bit word payloads of the Traffic Measurement cell. Enables alternate to create a 32 bit word of VCI/PT/CLP information. A Select is generated to allow a selection between payload and and Traffic cell header information, at the input to the internal FIFO. An Insert signal is generated to signal the FIFOStateMachine to insert a word into the internal FIFO. EnbVCI and EnbPT_CLP are signals generated to indicate that the input data should be latched by the Input Register. The CntOut signal is the count that comes directly from the 8 bit counter of observed idle times. The Selrle signal is asserted when a count needs to be inserted, which occurs when a CellValid signal is asserted. The PostEnbVCI signal controls one stage of the pipeline made of 16 bit registers. The InsertHalfWord signal is generated so that a 16 bit mux can advance either an idle count or VC value.

4.3.2 Module InputRegister

This module serves to obtain the cell stream header information which consists of 14 bits of VCI, the PT bits and CLP bit. One bit of the Payload Type is

thrown away. The remaining 3 bits of PT and CLP are mapped into 2 bits. The resulting header information is 16 bits wide and is submitted to 16 bit registers for the creation of 32 bit word payload for the Traffic Measurement cell. Unlike the standard version, no asynchronous reset is provided to the InputRegister since idle information is reflected in a count.

4.3.3 Module SixteenBitRegister

This register is provided so that any valid VC coming from the InputRegister can be stored. This storage allows the idle count to be advanced into the multiplexer before the insertion of the VC value. This register creates a small input FIFO necessary for maintaining VCI and PT/CLP information while there is a count insertion downstream.

4.3.4 Module SixteenBitMux

The multiplexer provides a simple means to place real VC data and run length idle counts in the same word stream. In the data stream, a VC value is always followed by an idle count which immediately followed by another VC value.

4.4 FIFO Section

4.4.1 Module FIFOStateMachine

The state machine consists of logic derived from the Insert and Grant signals. When a valid word is inserted into the FIFO with the Insert signal, a Request is asserted. Upon a Grant, the state machine advances the 32 bit words residing in the FIFO. The last word is advanced to the the TriStateBus, upon which the Output enable signals are asserted to send the word out on the ATM_Words bus. The result of the Traffic Measurement Chip is a Measurement Cell streamed out

one word at a time.

The amount of information that the chip generates in the Run Length mode is variable and depends on the data rates of the observed streams. When no data is being transmitted and the chip is streaming idle information; it generates about 17 Kbytes/s. When four OC-3 streams are routed through the Gateway, the measurement stream rate is 4 Mbytes/s.

4.4.2 Module TriStateBus

A Tri-state bus is not provided in the Run Length Encode version since only one probe can be fitted into a Xilinx XC3195. The output of the third 32 bit register of the FIFO section is fed directly into the Tri-state I/O Blocks.

Chapter 5

Design Verification

5.1 Digital Simulation

The chip-sets designed to perform Traffic measurements were verified in several ways. When the code was described in M3 (Modula3), it was then compiled to create an hdl (high level description language) file and xnf (Xilinx netlist) file.

The hdl is a description of the chip in high level description language, and simply describes the logic the chip performs. Mentor Graphics Quicksim was used to examine the timing diagrams and logic of the hdl file. A number of simulations were performed where the chip was given a set of inputs and the chip responded by creating output signals. In other words, the virtual operation of the chip can be performed and all signals can be observed for anticipated correctness. It must be kept in mind that the delay characteristics of the real device were not incorporated into these simulations.

The xnf file is used to create, in the end, the actual raw bit file. It is first used to create a map file. The map file is used to create an lca file that actual contains the routing for the chip. The lca is used to make an rbt file which is the final code that is included in the boot code and transmitted serially through the Gateway to the program the XC3195 FPGAs.

The lca file is often times examined for total delay characteristics. The XACT tool `xdelay` can exposed cumulative routing delays that can exceed the word

clock cycle that the chip logic is based on. It does not include any delays that input signals encounter on the way to the chip. Boundary delays involve special examination of other chip-sets that signals are generated from. This involves the use of XACT where the signals come from other Xilinx FPGAs. Complete timing diagrams including delay can be built manually in this way.

5.2 Hardware Testing

Hardware testing was performed by using several tools and making many observations. The chip was first examined with an oscilloscope. The Start signal and Enable_on_off signals were routed to unused pins to assure the LCP interface was working. The digital scope and logic analyzer were used countless times to ensure signal were either arriving to the chip as assumed, and that the generated signals and bus activity was as expected.

Tools as part of the LCP software interface were useful for examining the operation of the chip. The `ku_ShowVCs` command displays the traffic on a per VC basis. When the Traffic chip was operating correctly, the VC would appear in the list of active VC's. This tool was also useful when any spray of VC's occurred. Spraying of VC's occurs when the header of a cell coming out of the Delineator or the Traffic chip is misinterpreted. This can happen when either the Traffic chip or Delineators output a piece of payload when the SRAM Controller expects a header with a VC value. It could also happen if the words on the bus are not being written into the Delineator SRAM correctly, by violating SRAM timing conditions[7].

Another method used to verify the chip was the use of the host workstations. The Alpha workstations were used to send traffic through the Gateway. It was known that the workstations were capable of fully inundating an OC-3 stream with TCP level data[12]. The Traffic chip should report that the Bandwidth occupied

by the stream is the total possible available at an OC-3 rate[8]. In other words, 135.6 Mb/s should be observed to be available to the adaptation layer. This corresponds to 149.7 Mb/s available to the ATM layer after SONET overhead is stripped.

Additionally, the host workstations could be setup to send Constant Bit Rate cell streams. One particular investigation involved the use of an Alpha station to stream data a 5 Mb/s on a certain VC. The Traffic chip was used to observe the stream and reported 5.2 Mb/s

Yet another method used to verify the chip was programming the chip with both a Run Length version and a Standard version. This allowed a type of cross correlation and verification. The actual design was very tricky to fit into one Xilinx. The two versions were turned on at the same time to observe the same data streams. If the two versions reported the same information, the chip was deemed valid in both versions.

Lastly, the ARTS project was used to examine validity of the TX probe at the AAL5 layer. Packet data had not been verified up until this point.

5.3 Findings

Early on, the problems found were varied. After fixing several problems in both the Traffic chip and other chips, a measurement stream of considerable accuracy resulted at the host workstation. One problem involved the dropping of traffic measurement cells. At first, the SRAM Controller was simulated heavily and found to neglect the measurement stream. After the Sram Controller chip was fixed the problem lessened. Again however, the stream occasionally lost cells, on the order of one per 1000 measurement cells. Extreme attention was paid to this problem of determining why measurement cells were dropped. In the end, the problem was that the SRAM write enable and chip enable signals did not

conform to the specifications dictated by the manufacturer. This only occurred when the Delineators were feeding heavy traffic and the Traffic chip was streaming measurement words. To this point, all problems have been fixed.

Chapter 6

Data Retrieval and Operation

6.1 Chip Operation

To operate the chip, several preparation steps need to be taken. Since the chip resides on an assembly that contains a motherboard, a telnet session is established so that commands can be issued to the card assembly. Several routines were created for the Traffic chip and are a part of the kernel, or “application” that is transferred via tftp on boot. Additionally, the application can reside in Flash ROM suitable for booting when no tftp session is available. The utility routines specific to the Traffic chip are described below.

6.1.1 `trafficvc(vc,stream)`

`Trafficvc` is used to set the VC for the outgoing traffic measurement cells. The first argument is the VC in hex. The second argument is the stream number that creates cells with the given VC. Table 6.1 describes the possible probe numbers, however, downloading the correct chip that was compiled with a specified probe is essential for correct operation. If this is done wrong, the Gateway will most likely reboot.

Table 6.1: Traffic Chip Probes

probe	function
0	transmit probe
1	cell forward after recv FIFO
2	cell valid before recv FIFO

6.1.2 traf_pulse(stream, time)

traf_pulse is a program that pulses the chip on and off. The first argument is the probe number to pulse, and the second argument specifies the duration in milliseconds. The value is approximate since the LCP is not a perfect real time system.

6.1.3 enable_traffic(stream)

Turns the chip on. Issue disable_traffic to stop the chip.

6.1.4 disable_traffic(stream)

Turns the chip off. Issue enable_traffic to start the chip.

6.2 Raw Cell Traffic Capture

The Traffic measurement chip generates raw ATM cells that contain traffic VCs in the relative positions that they were observed in the cell stream on the Gateway. The Traffic chip does not generate AAL5 packets with a 32 CRC. Therefore, a host ATM entity was needed that could capture raw cells of a certain VC and store them. The Alpha workstations are generally considered to be high performance workstations and have turbo channel OTTO ATM cards. Since the device driver code for the OTTO cards was available, it was deemed a sensible choice to modify the driver code to recognize and generate ATM cells with no Adaptation layer

encapsulation. The work for this driver was performed by rmenon et al. The basic modifications include the addition of system calls that allow the read and write of the OTTO device in raw cell mode. The size of the receive buffer can be designated and the status of the VC mode and buffer size can be queried. The code for the OTTO tools is found in Appendix A.

6.2.1 Setdev

Setdev is a program that tells the OTTO card to recognize various packet level formats for a specified VC. If the card is told to operate in raw cell mode, it stores all cells on a certain VC regardless of whether they are part of a adaptation layer packet. The arguments for setdev are:

- VC to identify for setting adaptation layer format.
- AAL type of adaptation layer the OTTO card should expect on the specified VC. A value of 6 tells the OTTO that it should operate in raw cell mode for this VC.
- rxbufsize the size of the buffer in bytes the card should allocate to the given VC. Large values can be requested since the OTTO card can use main memory by use of turbo channel.

6.2.2 Readtraffic

Readtraffic is a program to read the OTTO receive buffer. The arguments are

- VC buffer to read from. The buffer contains consecutive ottorw structs which in themselves contain 52 bytes of raw cell information minus the HEC.
- output filename in which the data will be placed. The outfile is ascii and contains one entire cell per line with blanks in between.

6.2.3 Statusdev

Statusdev is a program that allows for the examination of the current state of the receive buffers for a specified VC.

- VC value to examine. Information reported back is size of the receive buffer, amount filled, number of cells received and location of buffer. The number of cells received is valuable since it tells us whether the traffic from the chip actually showed up. The readtraffic tool reads out all the cells in a buffer.

6.3 Post-processing

The post processors assume that the input file is one generated by readtraffic. The readtraffic routine is used in the same way whether the Standard or Run Length version of the chip is used to generate measurements. After all, ATM cells are all the same at the host, only the payload differs. Use of the correct post processor depends on the type of Traffic chip used to generate measurements. All post processors generate the same type of output, that being MATLAB¹ files, ready for use by MATLAB. The only stipulation is that MATLAB expects files with an extension of .m. The code for the post processors is found in Appendix A. All post processors consist of a read_file, parse, and print routines. The read_file routine is the same for all programs and is as simple as reading each line of the input file and extracting the payload of the cells. The information for the cells is conveniently placed in a large two dimensional array. The array elements are 4 bytes wide. The VC in a measurement cell payload is only 16 bits wide, but when converted to ASCII is 32 bytes wide. Each nibble reported in hexadecimal becomes an unsigned char.

All post processors expect a set of arguments as follows:

- input filename generated from readtraffic
- name of time series output file for use by MATLAB(give it a .m extension)
- name of interarrival output file for use by MATLAB(give it a .m extension)
- VC to look for in the input file. The time series and interarrival information will be given based on the this VC. All data is looked at on a per VC basis. An input file can be processed

¹MATLAB V4.2a (c) Copyright 1984-94 The MathWorks, Inc.

several times over looking for different VC information. For instance, to use the maximum Gateway bandwidth, four different OC-3 stream with different VCs have to be generated. The output file can be processed four different times with the same post processor and the only argument to change is the VC value to look for.

6.3.1 Run Length Processing

`postrle_cell`

A post processor for the Run Length Encoded version with attention to the cell level. The parse routine takes a large two dimensional array by pointer and a target VC value as arguments. The array is traversed while looking for run length counts and the target VC. A count called timecount is maintained while comparisons are made. If the value 'FF' is seen in the first two bytes of an array element, it means there is a run length count that follows in the next two bytes. The maximum count reported by one array element is 255 which would be an FFFF. This means that a residual count will follow in the next array element and must be added to the current value. It is not unusual to see interarrival times on the order of 1000 cell times, especially on low bit rate streams. The timecount is increased by the total run length counts observed. In between counts, the array element will contain a VC value which does not necessarily have to correspond to the target VC value. If it does not, a cell time is added to the current timecount. If the array element does correspond to the target VC, the timecount is stored in an integer array called timeseries. One slight detail about the target VC is that, if for instance, it is 0x0064, the parse routine will also look for 0x8064 since this value indicates an AAL5 packet trailer for the VC 0x0064. The so called end of packet marker qualifies as a target VC.

The print routine simply generates MATLAB files from the timeseries arrays. Cell interarrivals are generated by subtracting successive timeseries values. By default, a plot of the cell level time series is generated and a histogram of cell interarrivals for the target VC. The names of the MATLAB variables are:

- trfi for traffic cell interarrivals
- trft for traffic cell level time series

postrle_pkt

A post processor for the Run Length Encoded version with attention to the packet level. It is assumed that the Traffic chip is observing AAL5 type packets since it is the only packet type that includes packet boundary information in the head of the ATM cell. The parse routine takes a large two dimensional array by pointer and a target VC value as arguments. The array is traversed while looking for run length counts or the target end of packet marker, while all the time keeping a time count. The counts are treated the same as in the cell level code described above. The major difference between the packet level and cell level code is the detection of the end of packet markers for the specified VC. For example, if as before 0x0064 is the target VC, then the parse routine will look for 0x8064. When the first end of packet marker is found, then the next 0x0064 seen is the head of a packet. From this technique, when packet boundaries are found, the time count array is updated.

The print routine simply generates a MATLAB file from the timeseries arrays. Head of packet and end of packet interarrivals are generated by subtracting time-series values. The first time series value is a timestamp of a trailer, the next is of header, then trailer and so on. The first packet is discarded and not reported in the output files, since often times the chip starts up in the middle of observing a packet. The last packet is thrown away for the same reason.

By default, a plot of the packet level time series is generated for both headers and trailer and a histogram of packet interarrivals for both header and trailers is generated for the target VC. The names of the MATLAB variables are:

- t.head for packet level header time series
- i.head for interarrival data of packet headers
- t.trail for packet level trailer time series

- `i_trail` for interarrival data of packet trailers

6.3.2 Standard Processing

`postnonrle_cell`

A post processor for the Standard version with attention to the cell level. The parse routine takes a large two dimensional array by pointer and a target VC value as arguments. This routine is much more simple than the run length count version. The array is traversed while looking for the target VC all the time keeping a time count. If a value of 0000 is seen, then it means that a single idle time was observed by the chip. There may be several consecutive idle values which must be counted until a target VC is seen. If the array element contains a VC value which does not correspond to the target VC, it is added as an idle cell time. If the array element does correspond to the target VC, the timecount is stored in an integer array called timeseries. As with the run length version, the parser qualifies end of packets markers.

The print routine simply generates MATLAB file from the timeseries arrays. Cell interarrivals are generated by subtracting successive timeseries values. By default, a plot of the cell level time series is generated and a histogram of cell interarrivals for the target VC. The names of the MATLAB variables are:

- `trfi` for traffic cell interarrivals
- `trft` for traffic cell level time series

`postnonrle_pkt`

A post processor for the Standard version with attention to the packet level. It is assumed that the Traffic chip is observing AAL5 type packets since it is the only packet type that includes packet boundary information in the head of the ATM cell. The parse routine takes a large two dimensional array by pointer and a target VC value as arguments. The array is traversed while looking for target

end of packet markers, all the time keeping a time count. The idles are treated the same as in the cell level code described above. The major difference between the packet level and cell level code is the detection of the end of packet markers for the specified VC. For example, if as before 0x0064 is the target VC, then the parse routine will look for 0x8064. When the first end of packet marker is found, then the next 0x0064 seen is the head of a packet. From this technique, when packet boundaries are found, the time count array is updated.

The print routine simply generates MATLAB file from the timeseries arrays. Head of packet and end of packet interarrivals are generated by subtracting time-series values. The first time series value is a timestamp of a trailer, the next is of header, then trailer and so on. The first packet is discarded and not reported in the output files, since often times the chip starts up in the middle of observing a packet. The last packet is thrown away for the same reason.

By default, a plot of the packet level time series is generated for both headers and trailer and a histogram of packet interarrivals for both header and trailers is generated for the target VC. The names of the MATLAB variables are:

- t.head for packet level header time series
- i.head for interarrival data of packet headers
- t.trail for packet level trailer time series
- i.trail for interarrival data of packet trailers

6.4 VC setup

Setup of VCs can be done in any number of ways. The Out of Band manager on the AN2 is sufficient, however, TISL has created a server that allows socket connections by client software. An2setup, an2teardown are preferred tools for VC creation and teardown. They allow a user to execute a script containing the appropriate commands.

When the chip has been setup correctly and the VC's have been setup to route the traffic cells to a host Alpha station, the cells arrive to the OTTO and reside in the receive buffer. The cells wait there until they are read with readtraffic. The output file is then post processed and the data is analyzed with MATLAB.

Chapter 7

Verification of ATM Reference Traffic Source

7.1 Description of system

The Traffic chip is being used in a research project to examine the ATM traffic created by a Pentium based computer with an OC-3 interface running a Linux operating system. The objective of the project is to generate AAL5 level packets accurately according to a specified distribution. The Linux operating system has been modified to have a personality that resembles a real time system[2]. The operating system is responsible for scheduling the departure of packets, making it dedicated to the control of its ATM card and device driver. The end result of the project is an inexpensive and reliable ATM traffic source. It can be very flexible in terms of the type of traffic distributions that can be created.

7.2 Method

The standard Linux kernel schedules events based on a timer. The timer for the system counts down to zero every 10mS and creates a signal at that time called "interrupt 0". Interrupt 0 is special in nature and is given precedence over other interrupts in the system but like all interrupts, it initiates the execution of a

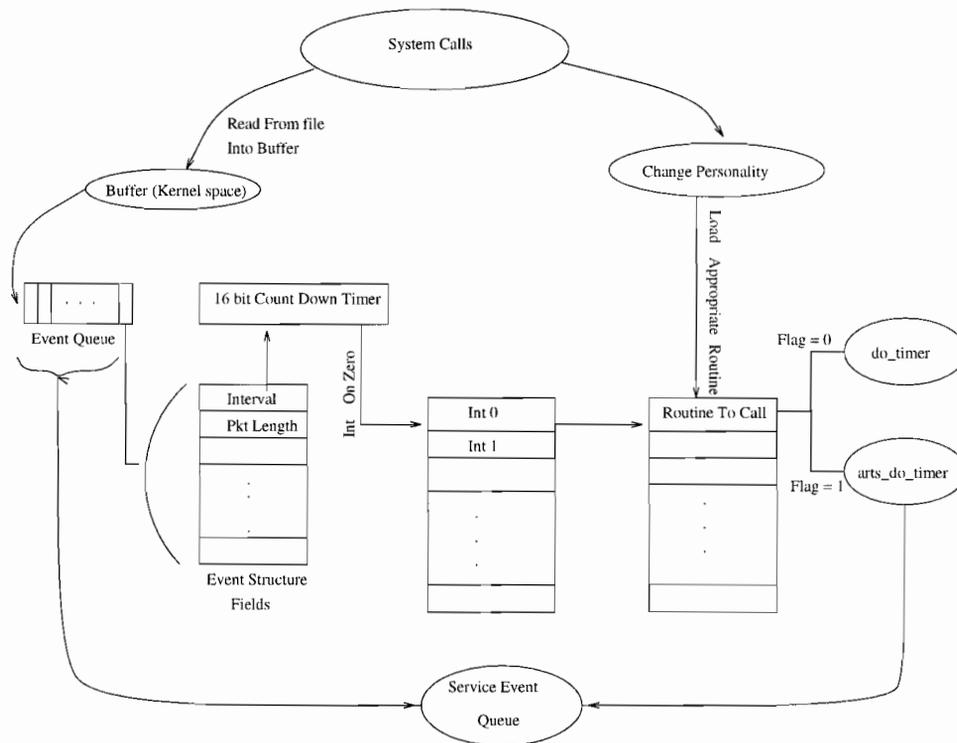


Figure 7-1: ATM Reference Traffic Source, System Design

service routine. In the case of the ARTS (ATM Reference Traffic Source) project, the routine places an event in the event queue which ultimately results the release of an AAL5 packet. Since the initial value loaded into the timer can be modified, then interrupts can occur at a time specified by a user. This is the essence of a programmable reference traffic source. The system is shown in Figure 7-1.

The system is subject to some variations since the event queue can take some time to service. To be successful, the operation of the traffic source must be consistent through all ranges and since the AAL5 packet size can be altered, the system must perform well for different packet sizes. This work is ongoing at the University of Kansas.

7.3 Evaluation

The Traffic chip was used in the early test stages of the project. A connection from the host PC was setup directly to a quad-line card in an AN2. Permanent Virtual Circuits were setup on the switch which routed the traffic through the Gateway. It should be stressed that no other switch of any type was ever introduced between the quad-line card and the PC. This would only add variance to the stream. The measurements would no longer be representative of the actual source. The ultimate measurement would be the examination of data at the output port of the PC, however, the Traffic chip can do a sufficient job in timestamping the packet interarrivals. The transmit probe Run Length Encode version of the chip was used to gather the measurements for this experiment. At maximum, three memories are encountered before a stream coming into a quad-line card is examined by the Tx probe of the Traffic chip. These are a proposed FIFO inside the SUNI chips, two cell buffers, and the VRAM on the quad-line motherboard[15].

For the first tests, the data gathered by the Traffic chip showed very periodic cell data being generated by an ARTS PC. However at this stage, the Traffic chip had undergone only simple AAL5 testing to verify its operation. The only tests performed to calibrate packet level data measurements involved a version of the chip that had both a Run Length and Standard design compiled into one chip. The ARTS project helped to reveal a critical timing problem that the Traffic chip possessed on one of its input PT/CLP bits. The PT/CLP bits are latched by CLB's and not IOB's due to efficiency concerning timing specifications[16]. A situation existed where the PT data was early with respect to the latch enables and thus the data was not latched properly. The PT bits contain the end of packet marker for the AAL5 packet.

Later tests proved to be promising for the ARTS project. The packet data in Figure 7-2 and Figure 7-3 shows header of packet and end of packet interarrivals. A value was loaded into the count down timer that generated interrupts every 4000

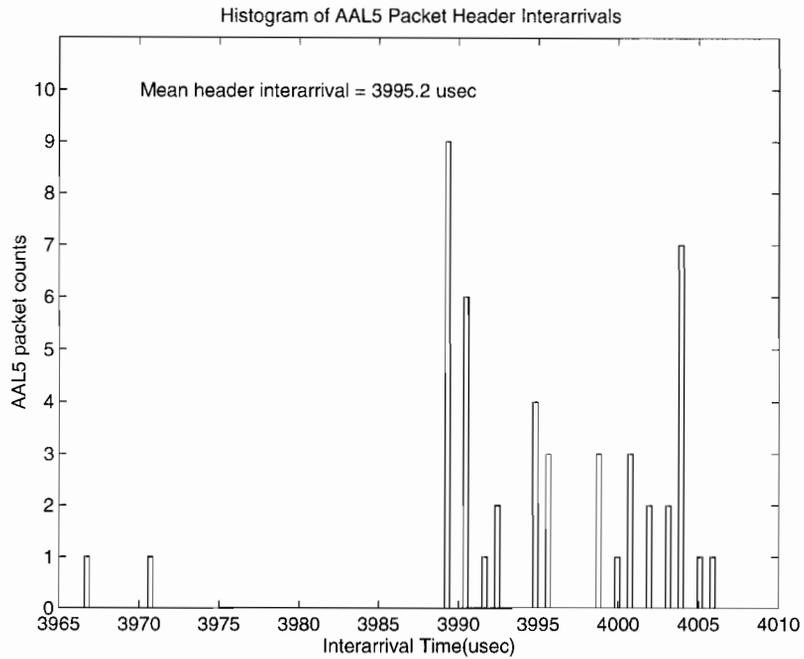


Figure 7-2: AAL5 Header Interarrivals for $T=4000\mu S$

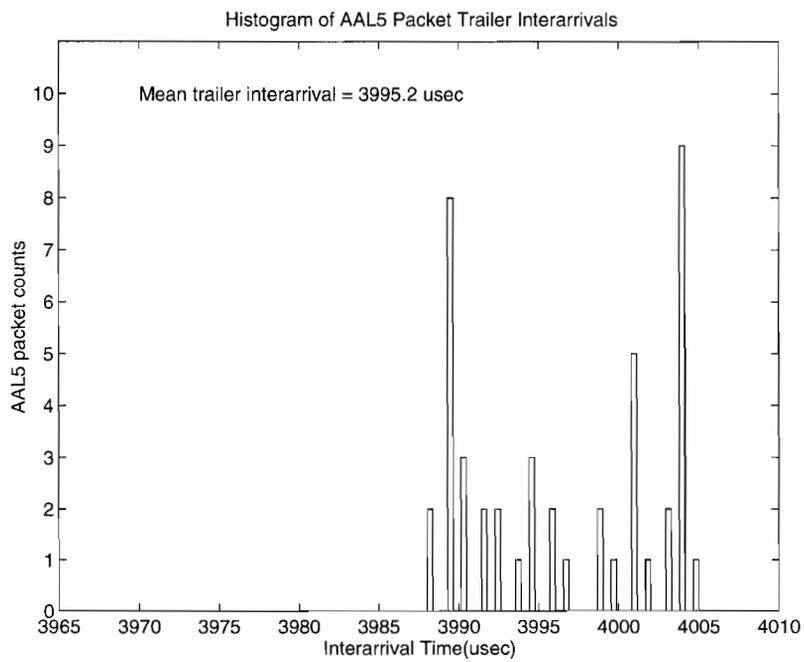


Figure 7-3: AAL5 Trailer Interarrivals for $T=4000\mu S$

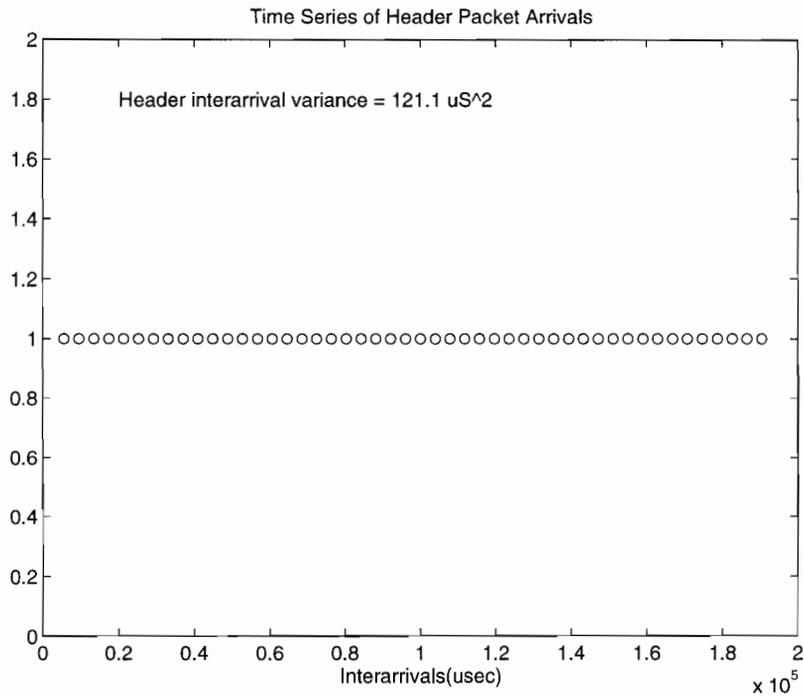


Figure 7-4: AAL5 Header Time Series for $T=4000\mu S$

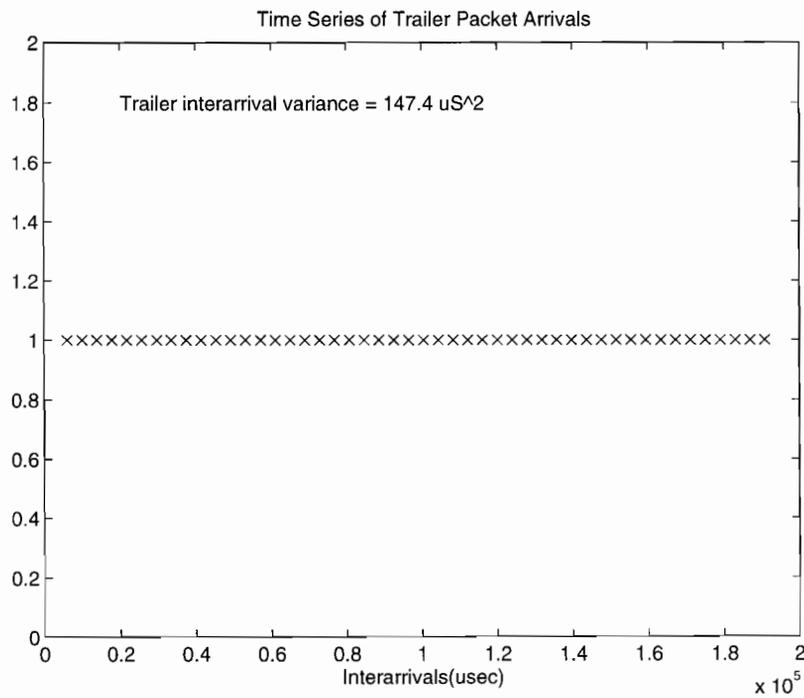


Figure 7-5: AAL5 Trailer Time Series for $T=4000\mu S$

μs and packets were 8192 bytes in size. The plots show that the mean for both the header and trailer are the same at 3995.2 μs . The result for this experiment is within two-tenths of a percent of the target. Additionally, Figure 7-4 and Figure 7-5 show a time series representation of the packet header and trailer arrivals. The data is very clean and the calculated variance of interarrivals is 121 μs^2 for headers and 147 μs^2 for trailers. The difference between head of packet and end of packet interarrivals can be attributed to the fact that the amount of time the ATM card takes to transmit the cells in a packet actually contains some variance. In other words, the time to clock a packet out on the fiber is slightly variable. Some variance can also be attributed to the memories a stream encounters in the switch that lie between the Traffic chip and the PC. For this test, the mean time to transmit a packet is 472 μs and the approximate bandwidth of the stream is 17 Mb/s available to the Adaptation layer.

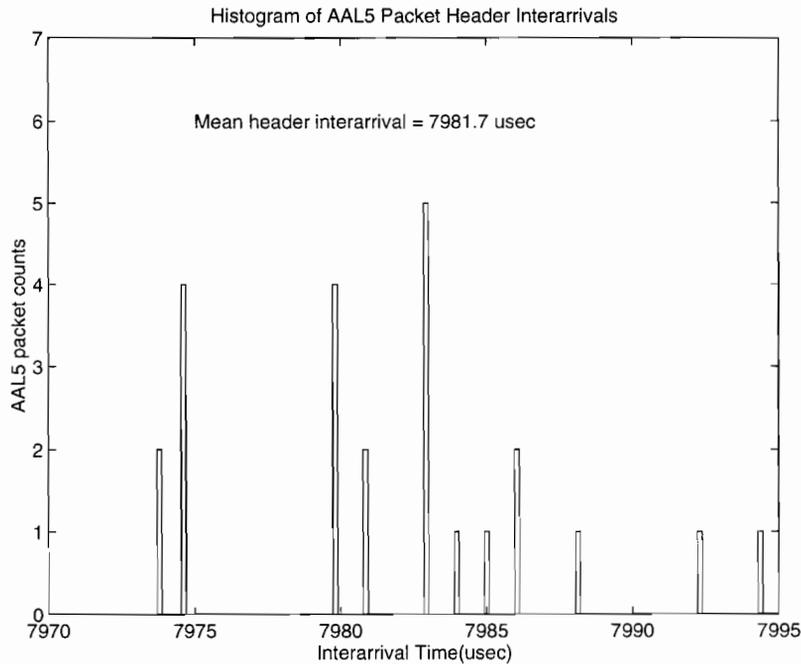


Figure 7-6: AAL5 Header Interarrivals for T=8000uS

Another test was performed where interrupts were generated every 8000 μs and packets were 8192 bytes in size. Figures 7-6 and 7-7 show that the mean for

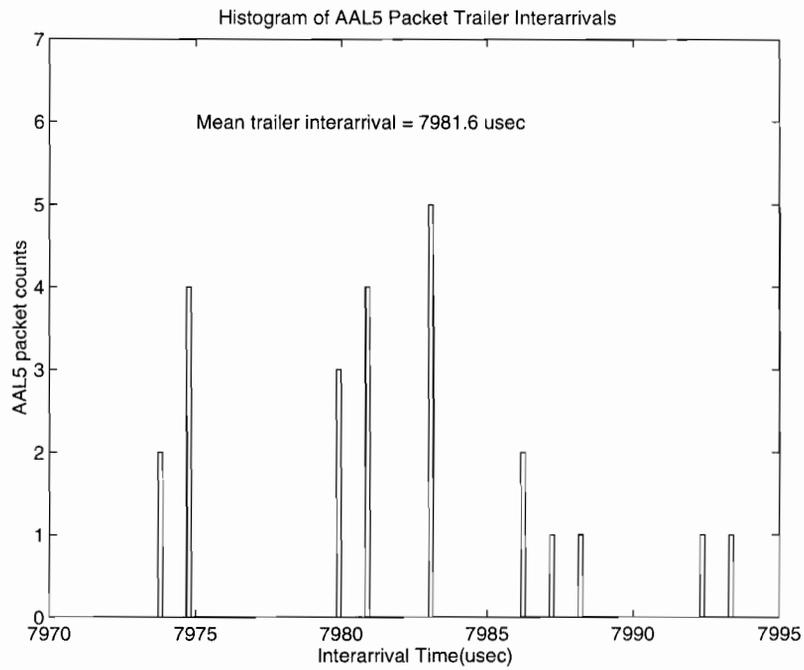


Figure 7-7: AAL5 Trailer Interarrivals for T=8000uS

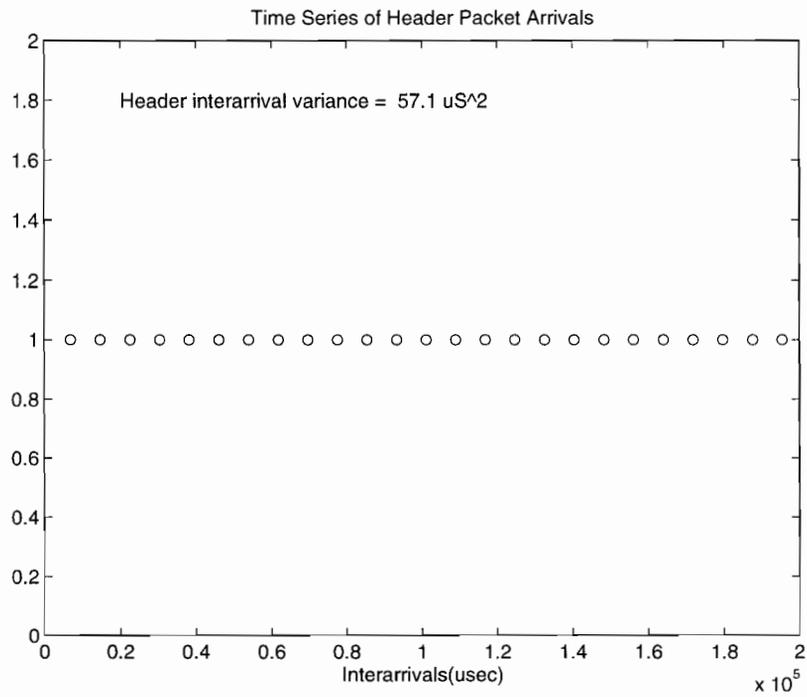


Figure 7-8: AAL5 Header Time Series for T=8000uS

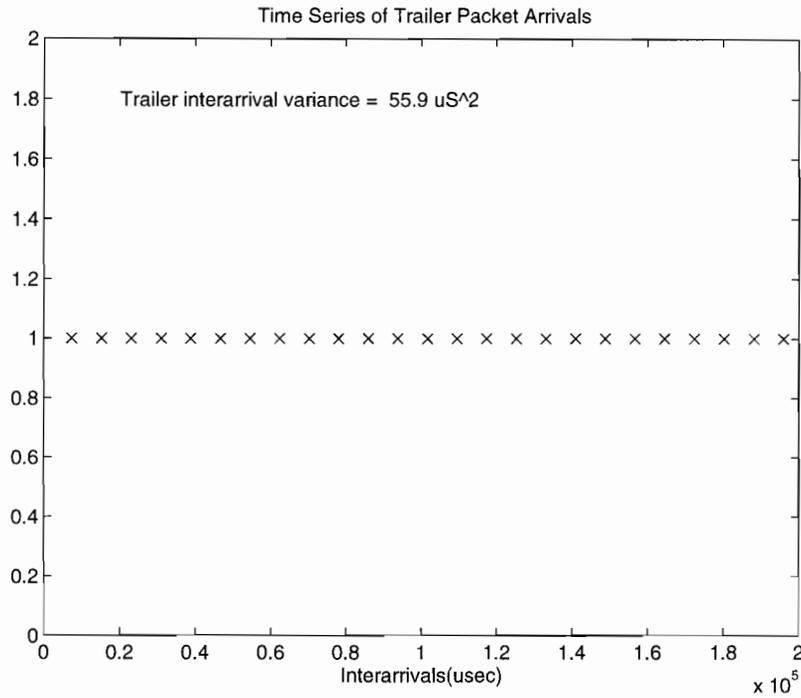


Figure 7-9: AAL5 Trailer Time Series for T=8000uS

both the header and trailer interarrivals are very close to the same at 7981 μ s. The results are within one-fourth of a percent of the target packet generation period. Figures 7-8 and 7-9 show the time series representation of the packet arrivals. The data is clean and the variance in this case is very small; between 55 and 57 μ s². For this test, the mean time to transmit a packet is 474 μ s which is similar to what was seen with 4000 μ s test. Again slight differences can be attributed to the ATM card and FIFOs in the switch. The approximate bandwidth of the stream is 8 Mb/s available to the Adaptation layer. This is half of what was seen in the previous test which is sensible since the packets are streamed out on the fiber every 8000 μ s instead of every 4000 μ s.

7.4 Further Evaluation

A new system has been proposed that uses a different device driver for the same ATM card. The new features of the device driver include an implementation of signaling for ATM networks. While this does not directly affect or improve the performance of the kernel modifications, it does introduce a better coherence with the ATM protocol. This is important for a traffic source deemed to be a reference in an ATM network.

7.5 Conclusion

While the work of the ARTS project is ongoing, the results seen so far indicate that the project is successful. Extensive testing needs to be done to ensure that the scheduling performance is not platform specific. The role of the Traffic chip in the project was the verification of the stream at an AAL5 packet level. The accuracy of measurements is due to the fact that the chip can examine AAL5 packet information at a cell level. Having cell level insight is also educational since the operation of ATM card can be observed. It may not always be correct to assume that a card will stream packets of equal size in the same amount of time for each individual packet.

Chapter 8

Examination of Flow Control

Flow control in an ATM network is essential for guaranteeing a Quality of Service. Various flow control methods exist, such as, credit based flow control and rate based control, however, the ATM Forum has settled on a rate based flow control for the standard. A question to pose: Is the Quality of Service achieved in manner that does not have adverse affects on the switch or violates any bandwidth contract?

An examination of flow control was conducted with the Traffic chip in hopes to expose the real behavior and performance of control mechanisms on several manufactured ATM cards. The tests were performed at different rates of 25, 50 and 100 Mb/s. Only the 25 Mb/s rate will be examined here since it is only one half of an OC-1 rate, and represents a reasonable bandwidth that might be commonplace in an ATM network with paying customers. The transmit probe of the Traffic chip was used and the focus of the experimental data is the ATM cell level.

8.1 Method

The setup for the tests was the same for each manufacturer. The source machine was connected to a quad-line card on an experimental AN2 switch that contained a quad-line card and the KU Gateway. The Gateway was situated to the be

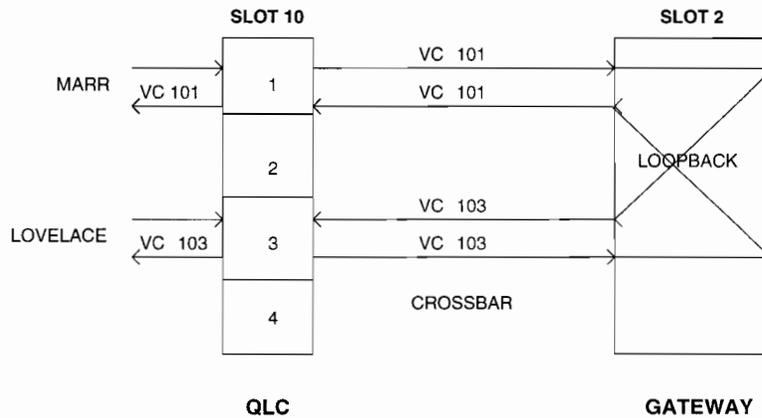


Figure 8-1: Permanent Virtual Circuit Configuration

the master of the switch, and permanent virtual circuits were setup from the quad-line card to the Gateway and back to the quad-line card. An example circuit configuration is shown in Figure 8-1, between two Alphas called marr and lovelace(actual addresses are marr-atm and lovelace-atm). TTCP was used to create traffic from the source. TTCP is as traffic program that essentially allocates large amounts of memory with no significant content to represent data. The data is encapsulated in either TCP/IP or UDP/IP packets then sent to a IP designated workstation. At a lower level, the host ATM device driver configuration for each test associates a particular IP workstation with its output port so that data is not sent out onto the ethernet, but rather out the lasers of the ATM card. TTCP was used to send UDP/IP traffic over the two workstation test network. The UDP datagram protocol was used since it was desirable to have no acknowledgments reported from the far end receive workstations. If TCP were used, it would result in additional flow control at layers above the ATM layer.

8.2 DEC OTTO Cards

The DEC OTTO card is a turbo channel ATM card manufactured for DEC workstations. It is an OC-3 card and has ST type input/output lasers, very much

suitable for connection to an AN2 or AN3 ATM switch. The `pvc` tool was used to establish the host workstation configuration which includes a setting for the desired bandwidth. The OTTO is in CBR mode when an integer bandwidth is specified and works as follows. The OTTO card maintains its own scheduling mechanism. Every cell slot time, the device sees if the current entry in the CBR transmit schedule is valid and the indicated VC can send a cell. The VC can send a cell if its disassembly queue is nonempty, if it is marked as sending according to the schedule, and if it is not on the transmit ready queue. The VC does not need to have a credit. The device then advances to the next entry in the schedule.

Deviations from the bandwidth setting can occur when the OTTO is lightly loaded so the schedule transmits slightly too fast. The driver must reserve extra capacity when it opens a VC to make sure the lightly loaded schedule will not overrun the switches' buffers.

8.2.1 DECstation 3000 model 600, "Alpha"

Machines named `marr` and `lovelace` were setup for this experiment. Both machines, have experimental DEC REV B(no fc capability) OC-3 OTTO cards. Both machines are DEC 3000 series 600 with 175 Mhz Alpha processors running OSF/1 V3.0. `Marr` contains 48 MB of memory and `lovelace` contains 96 MB of memory. The tool `pvc` was used to setup the host workstation configuration which includes a desired bandwidth setting of 25 Mb/s and virtual circuit settings. TTCP UDP was used in this case to create traffic at the source.

Figure 8-2 shows a histogram of the cell interarrivals for the experiment. While the mean calculated bandwidth is 24.33 Mb/s, very close to the target, the interarrival variance is very large. Figure 8-3 shows the complete data where cell interarrivals were observed at approximately 1700 cell slots(520 nS per cell slot).

The time series shown in Figures 8-4 and 8-5 is regular. Figure 8-4 shows only a small section of the data, the series is very regular in this region. Figure 8-5

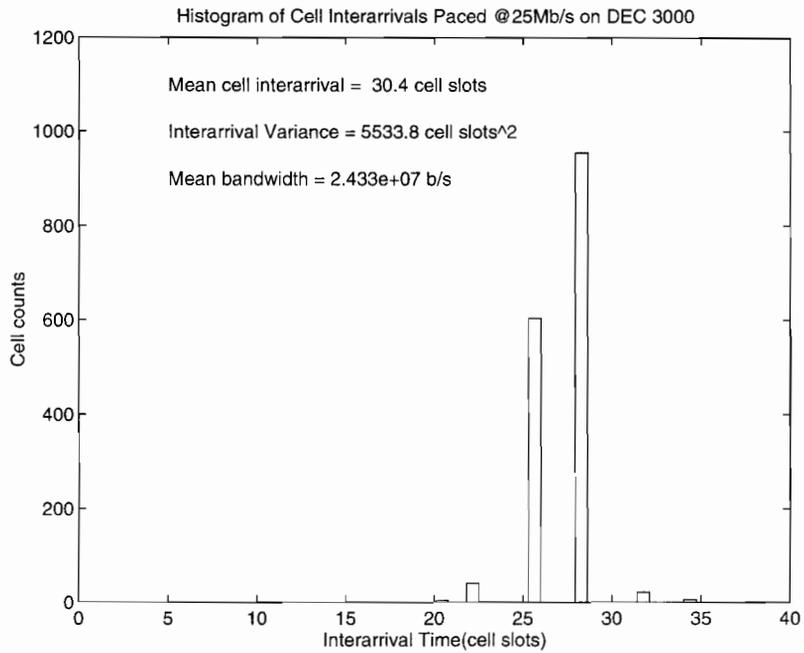


Figure 8-2: Histogram of cell interarrival times, pacing @25 Mb/s

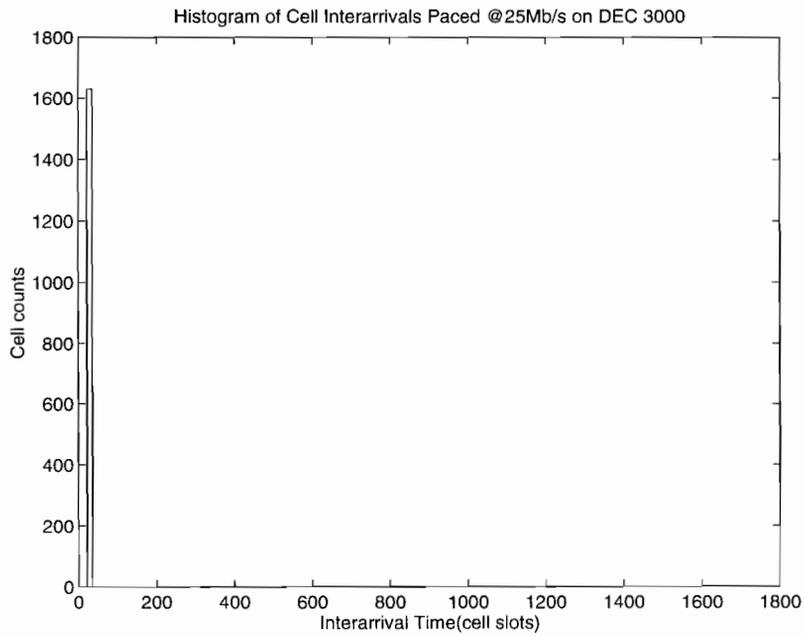


Figure 8-3: Histogram of cell interarrival times, pacing @25 Mb/s, all data

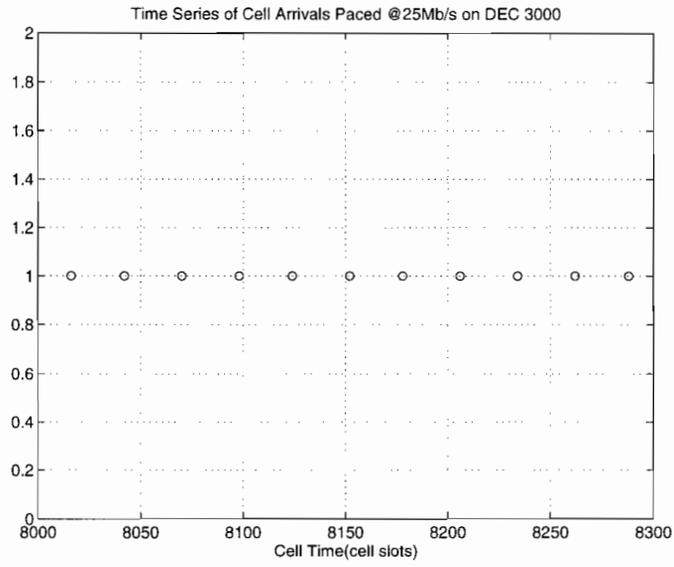


Figure 8-4: Time series of cell arrival times, pacing @25 Mb/s

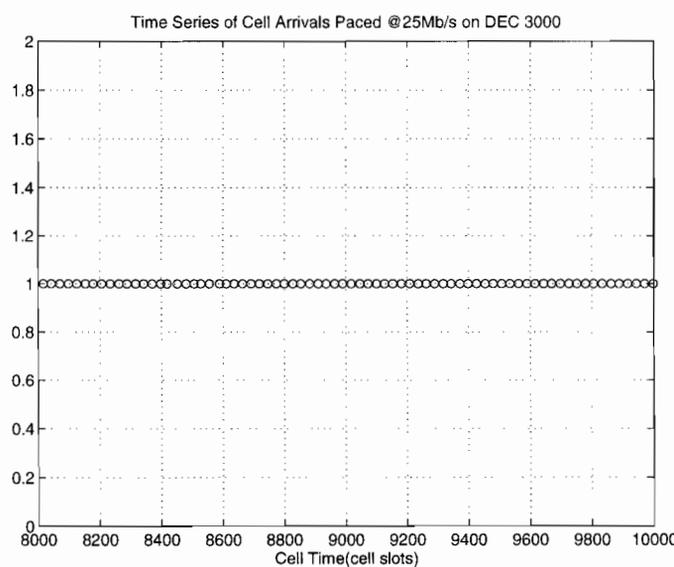


Figure 8-5: Time series of cell arrival times, pacing @25 Mb/s, all data

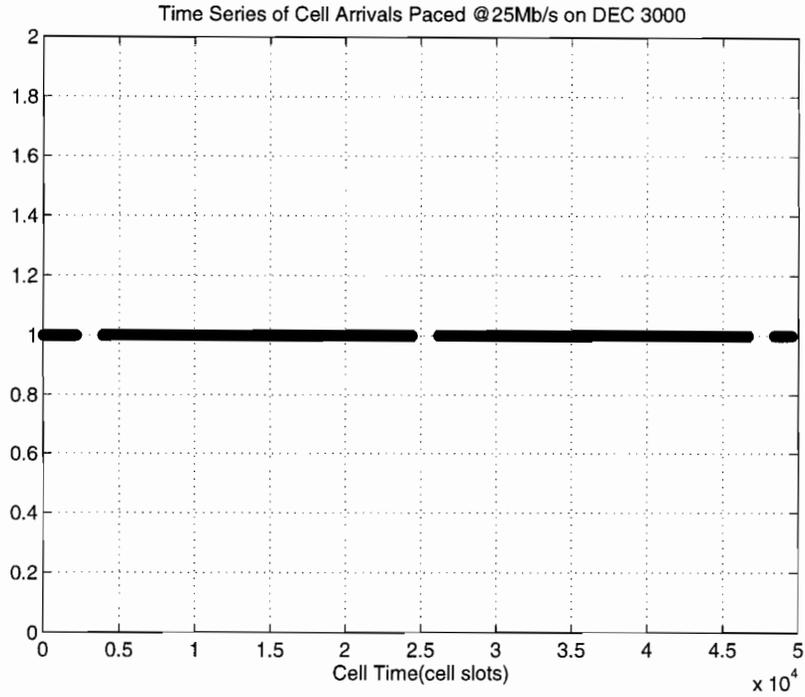


Figure 8-6: Time series of cell arrival times, pacing @25 Mb/s

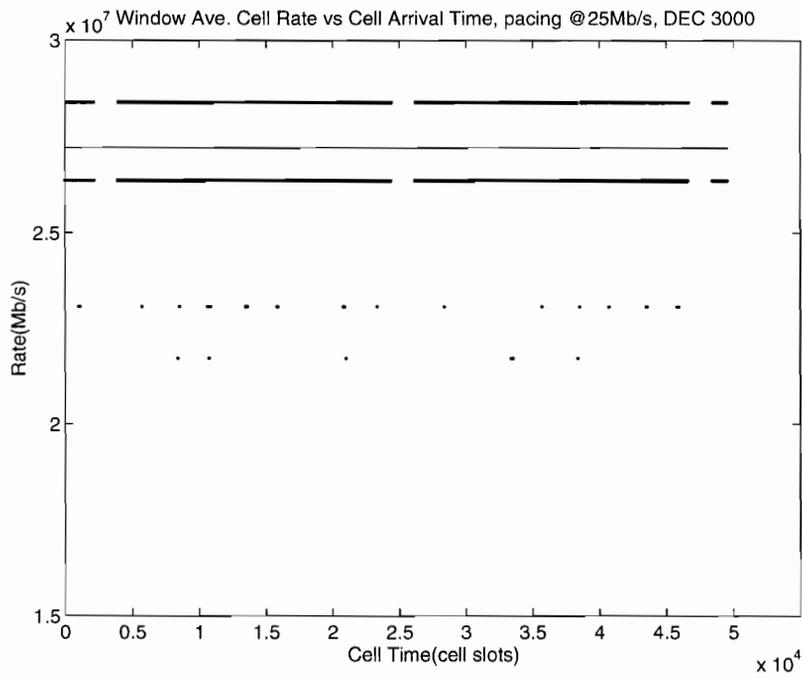


Figure 8-7: Time series of cell arrival times, pacing @25 Mb/s, all data

shows a larger view of the same data again indicating regularity in this region. Figure 8-6 is a complete look of all data taken, totalling approximately 25 mS in duration. The time series plot shows a periodic bursting of cells, where the gaps are 1700 cell slots in length. A question of concern is; while the average bandwidth is close to that of the target, why does the OTTO burst traffic? To help answer this question, another test was performed with an OTTO card in a special setup. Figure 8-7 shows a window averaged usable data rate versus cell slot plot.

8.2.2 DECstation 5000 model 240

A machine called aiken running Ultrix V4.3 was used in this particular experiment. It contains an a KNO3 rev 48 processor and has 65 MB of memory. It houses an experimental DEC REV B(no fc capability) OC-3 OTTO card. A special device driver was written for its OTTO card. A DEC 5000 is a dated machine and is not generally capable of flooding an OC-3 bandwidth without some help. The driver has some special tools available, namely the `link` program (proto version) which can stream data at high rates. TTCP is not needed in this case since `link` packages data. This helps avoid some protocol stack delays and process time. `Link` provides an option to let the user specify the how many cell slot times can be occupied with data.

Figure 8-8 shows the interarrival histogram for the experiment where the schedule was set so that every 6 slots contained data. The total capacity of 135.6 Mb/s is achieved when every slot is used to send data. If one sixth of the slots are used then

$$135.6Mb/s * \frac{1}{6} = 22.6Mb/s$$

Using the slot convention, a bandwidth of 25 Mb/s cannot be used. However, 22.6 Mb/s is acceptable and was seen to present useful results. The variance is small, and the bandwidth calculation from the Traffic chip indicates the stream achieved an average rate of 22.67 Mb/s. This is slightly over the target, but is

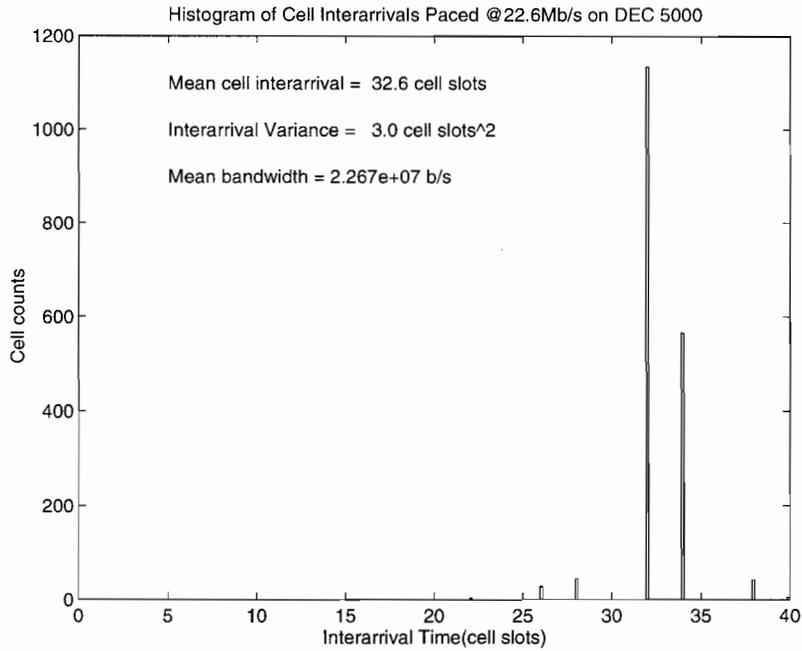


Figure 8-8: Histogram of cell interarrival times, pacing @22.6 Mb/s

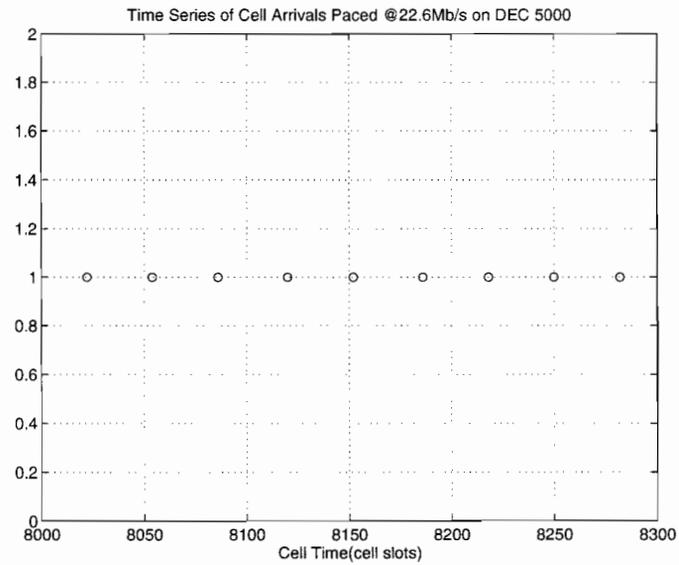


Figure 8-9: Time series of cell arrival times, pacing @22.6 Mb/s

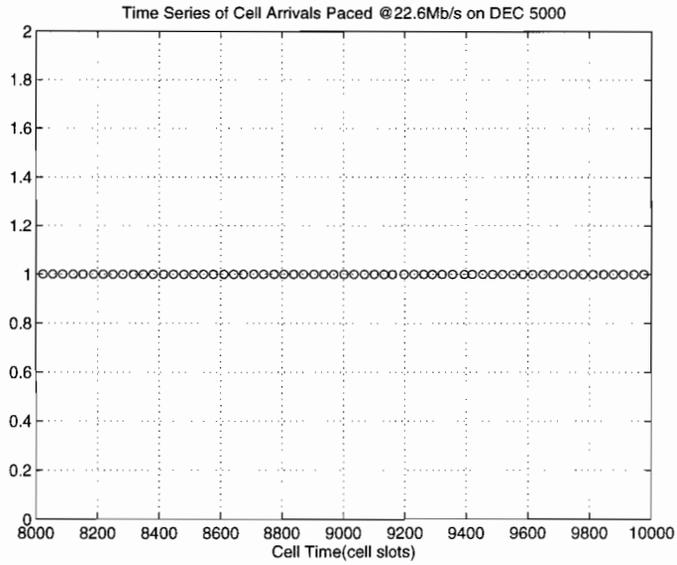


Figure 8-10: Time series of cell arrival times, pacing @22.6 Mb/s

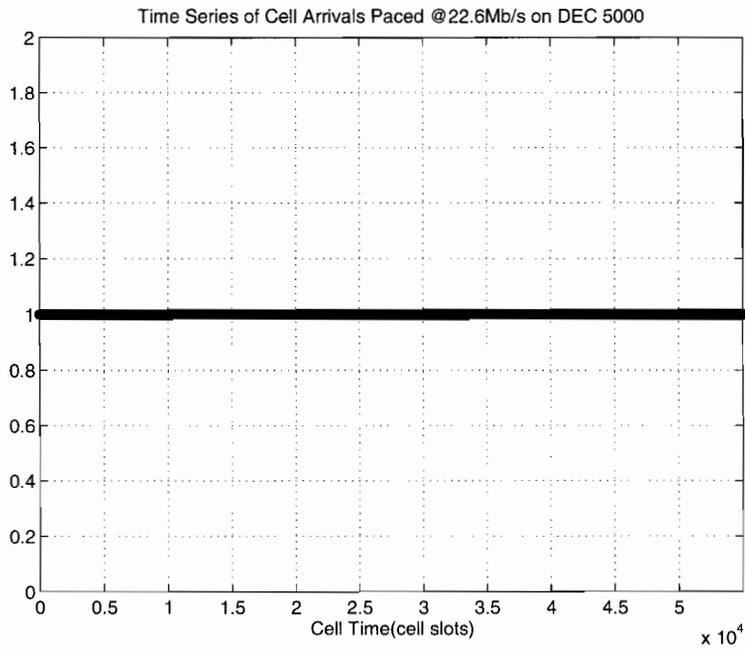


Figure 8-11: Time series of cell arrival times, pacing @22.6 Mb/s

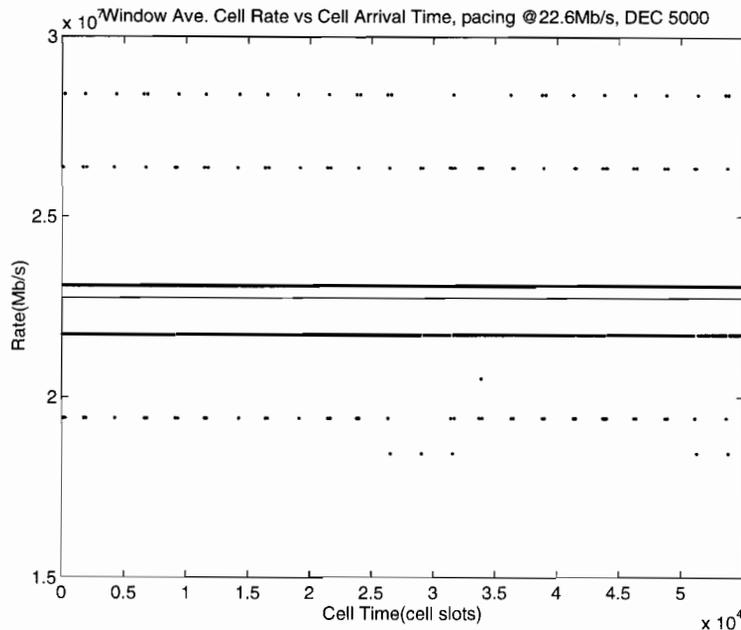


Figure 8-12: Time series of cell arrival times, pacing @22.6 Mb/s

consistent with some of the premises of the OTTO driver. Figure 8-9 shows a magnified time series of the cell arrivals. The data shows a very consistent cell stream. Figure 8-10 shows a larger time series of the same data. Figure 8-11 shows a time series of all of the data gathered. There are no gaps as seen in the DEC 3000 case. Figure 8-12 shows a window averaged usable data rate versus cell slot plot.

A DEC 3000 Alpha station is typically considered to be faster than a DEC 5000. Without careful consideration of the data taken in the suite of experiments, one might believe that a DEC 5000 out-performed a DEC 3000. Why does the DEC 5000 stream data consistently, and DEC 3000 burst traffic? The answer lies in the the upper layers of the of the data transmission process. The UDP packet length was 8192 bytes, but the socket buffer size was set to the 32768 bytes. Bursting occurs when the OTTO card has emptied its transmit queue. Under CBR mode, the socket buffer size at the UDP/IP level does not supply enough data to the buffers of the OTTO card. When the cell level transmit queue

is emptied, a gap in transmission is experienced. The problem was not an issue with the DEC 5000 since TTCP was not used, and data was assembled at a lower level, avoiding the protocol stack delays.

8.3 Fore SBA Cards

A Sun workstation named wigner is a HyperSPARC20 dual 125 Mhz processor workstation with 64 MB memory running Sun Solaris 2.4. It houses an OC-3 Fore SBA 200 ATM card. The input/output lasers have SC type jacks. The software for configuring the host is called **atmarp**. Since wigner was going to be connected to a DEC AN2 switch, classical IP PVCs had to be used. Fore has proprietary signaling built into the driver for use with Fore switches. Proper operation mandated classical PVCs and not switched virtual circuits maintained through signaling. The receive station was lovelace, as used in a previous experiment. There is no real concern that the far end station was not a Sparc station with a Fore card. Since UDP is used, the far end station serves only as a receptacle, not affecting transmission performance. TTCP was used with settings similar to those used in the Alpha test and wigner was setup to have a peak cell rate of 25 Mb/s. The Fore literature claims the cell rate will not exceed this value[14].

Figure 8-13 shows the interarrival histogram for the experiment. The variance is small and the mean bandwidth of 22.3 Mb/s is below the peak rate setting of 25 Mb/s. Figure 8-14 shows the cell stream time series which is fairly constant, indicating a well maintained scheduling algorithm. Figure 8-15 shows a larger time series of the same data where again the cell arrivals is very regular. Figure 8-16 shows a time series of all of the data gathered and there are no gaps.

Figure 8-17 shows a window averaged usable data rate versus cell slot plot, that sometimes exceeds 25 Mb/s. The time averaged rate is performed by taking the cell payload size and dividing by the interarrival time for that cell.

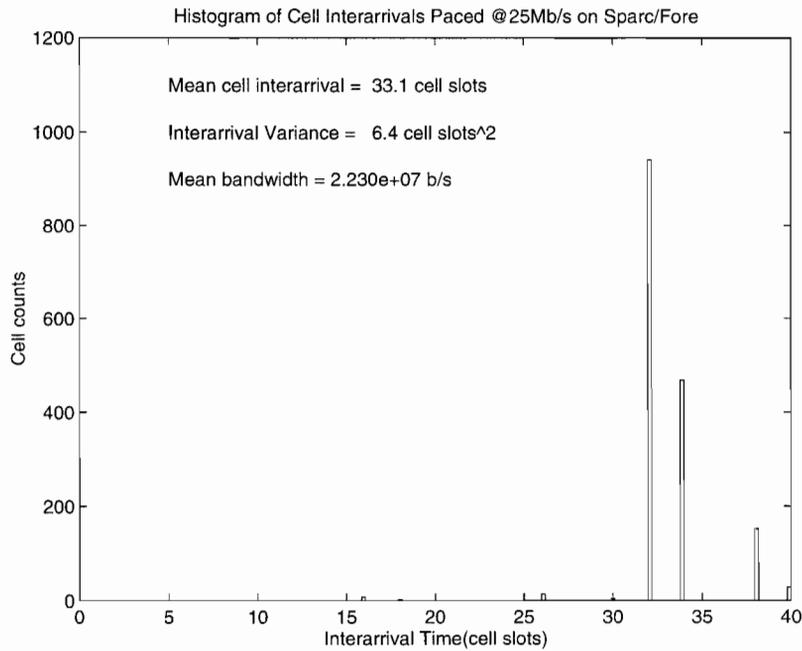


Figure 8-13: Histogram of cell interarrival times, PCR = 25 Mb/s

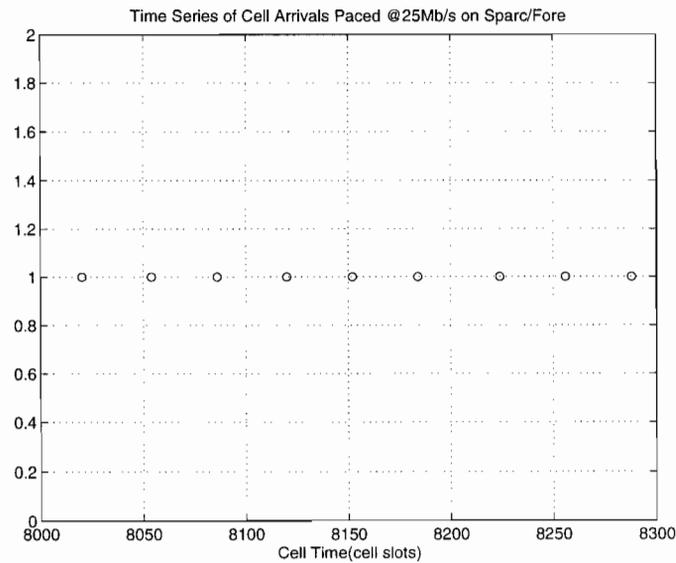


Figure 8-14: Histogram of cell interarrival times, PCR = 25 Mb/s

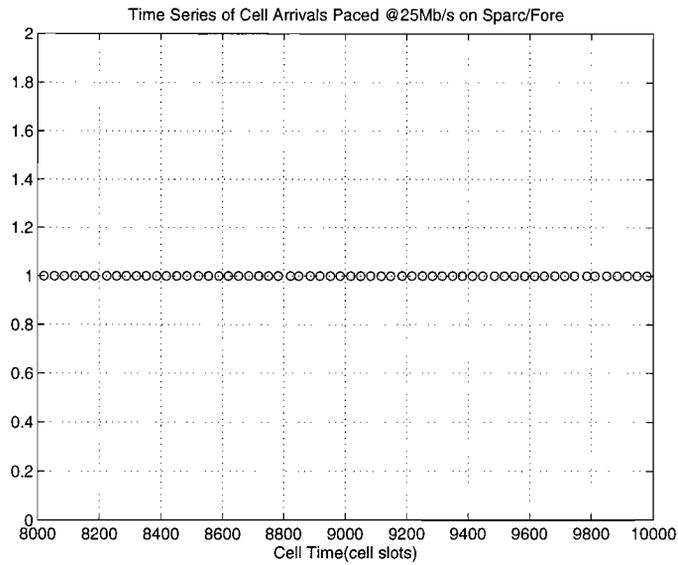


Figure 8-15: Window Averaged Usable Cell Rate vs Arrival Time, PCR = 25 Mb/s

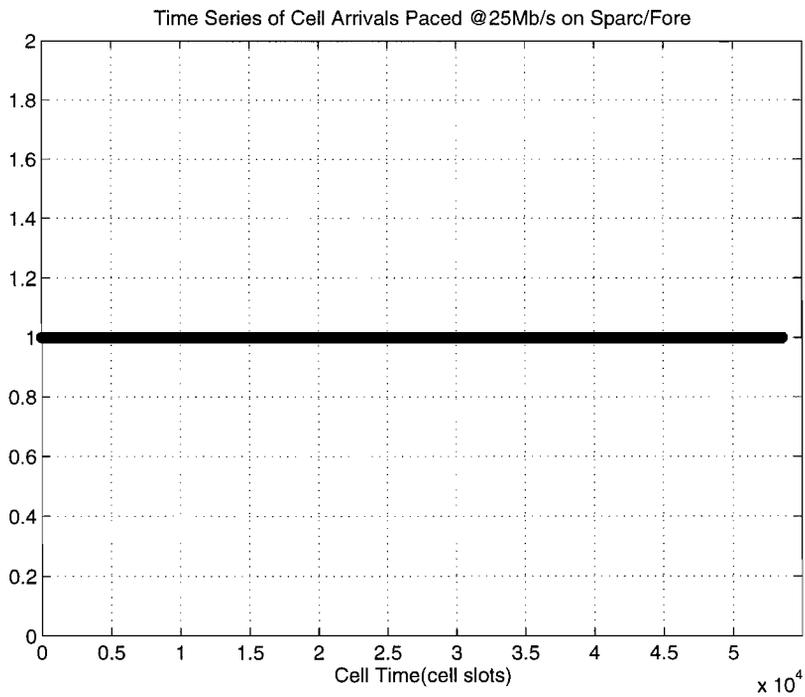


Figure 8-16: Histogram of cell interarrival times, PCR = 25 Mb/s

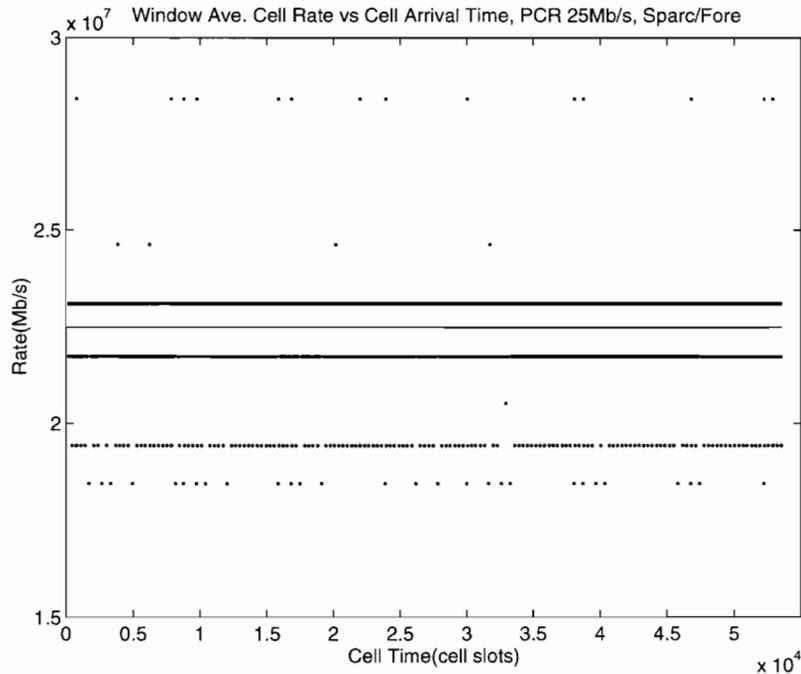


Figure 8-17: Window Averaged Usable Cell Rate vs Arrival Time, PCR = 25 Mb/s

$$\text{time averaged cell rate} = \frac{48 * 8\text{bits}}{\text{cell interarrival time in seconds}}$$

The usable cell data size is 48 bytes in ATM cells. According to this calculation, the Fore card violates the Peak Cell Rate Quality of Service it claims to maintain. Just how rigid a network will be on instantaneous bandwidth violations will be an interesting area of debate.

8.4 Examination of ATMTIMES on an Alpha

Atmtimes is a tool that is often used by researchers to collect ATM packet level data with a host workstation. It runs on an Alpha with an OTTO card and basically records a timestamp when a packet is clocked in or out. However, this type

of measurement is subject to scheduling delays and has some inherent problems, essentially tainting the measured data. An exaggerated Heisenberg uncertainty principle comes to mind, which says that nothing can be measured with absolute certainty. The question is: How accurate does `atmtimes` perform packet level data measurements?

The Traffic chip, while not perfect, is a hardware measurement device that was used in an experiment to determine just how well `atmtimes` performs. A flood of traffic from an Alpha setup in ABR mode, was examined by `atmtimes`. With `atmtimes`, the machine that is streaming the data is also the same machine taking measurements. Figure 8-18 shows how the traffic chip perceived the data stream as it passed through the Gateway. The bandwidth is very much where it should be for a full OC-3 stream. The limit is 135.6 Mb/s available to the Adaptation layer. A standard deviation calculation is shown in bits per second. Now, Figure 8-19 shows the same data collected by `atmtimes`. According to `atmtimes`, the total bandwidth exceeded the theoretical maximum. Additionally, the standard deviation is a magnitude larger than the value associated with the Traffic chip.

8.5 Conclusions

Flow control mechanism were observed with the Traffic chip and some interesting results were obtained. The exact nature of the cell level traffic can depend on the upper layer protocols, as with TTCP on the Alphas. However, the overall target bandwidth was closely maintained with the Alpha experiment, obscuring the burstiness of the cell level traffic. A Fore card was examined and found to behave well in terms of periodic scheduling. An exact definition of Peak Cell Rate is required before any conclusions can be made about the flow control mechanism. However a question might be asked, what could be the impact of flow control

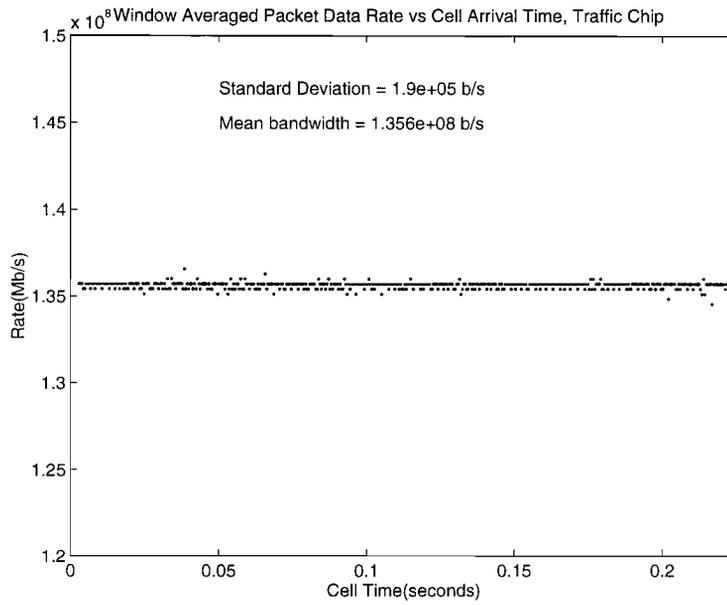


Figure 8-18: Rate vs time plot of ABR stream, observed by Traffic chip

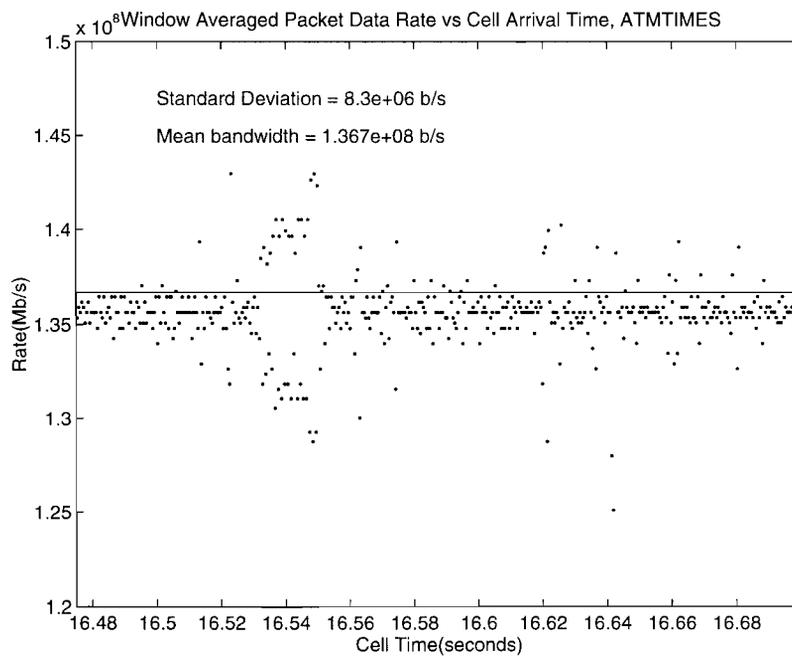


Figure 8-19: Rate vs time plot of ABR stream, observed by atmtimes

violations on ATM traffic contracts?

Lastly, the program **atmtimes** was examined with a “microscope” and found to be less accurate than hardware dedicated to the measurement of packet level traffic. This experiment defends the premise and motivation for the Traffic chip since switch level insight is needed.

Chapter 9

Conclusion

9.1 Summary and Conclusions

Two versions of the Traffic Measurement chip have been successfully designed and implemented on the KU Gateway running in OC-12 mode. The Standard version can operate with selections for two of three probes, allowing simultaneous measurements at two locations on the Gateway. This paper concerned itself with the use of the transmit probe, however, this does not mean that the receive probes have not been tested. The Run Length version of the chip provides a selection for only one of three probes, in exchange for a savings in measurement data bandwidth. The Run Length version is suitable for capturing long sequences of data. Several times, the chip was pulsed for one to ten seconds and the amount of data was still manageable.

9.2 Future Work

The design of the 4xOC-3 mode of the chip still needs to be done. The amount of attention given to the Traffic chip has been considerable. With the exception of a few chips, as much attention has been given to it as any other chip in the transmit and receive sections. The Traffic chip even mandated custom routing.

A useful experiment would have been to examine cell and packet level data

at the transmit probe, and then at the input to the receive FIFO. Then it would be interesting to rerun the experiment and examine data at the receive probe before the the FIFO, and the receive probe after the FIFO. This would be an investigation into some practical queueing theory and cell delay variation. The potential experiments the Traffic chip can perform are many and varied.

Other future work could include the measurement of Credit Based flow control mechanisms on the AN2 and multiplexing effects from other switches.

Bibliography

- [1] ATM Forum. ATM User-Network Interface Specification, Version 3.0, June 1993.
- [2] F. Ansari, J. Keimig. ATM Reference Traffic Source October 1995.
- [3] R. Ballart and Y. Ching. SONET: Now it's the standard optical network. *IEEE Communications Magazine*, 27(3):8-15, March 1989.
- [4] Bellcore Technical Reference TR-NWT-000253. Synchronous Optical Network (SONET) Transport Systems: Common Generic Criteria, Dec 1991.
- [5] B. Ewy Sram Controller Implementation TISL Technical Memo, October 1994.
- [6] M. Bog Proposal for Gateway Measurement Gathering, September 1992.
- [7] Cypress Semiconductor TTL SRAMs, Mar 1992.
- [8] P. Cavanaugh Capacity, June 1992.
- [9] H. Uriona Development and Verification of a Reconfigurable 4xOC-3/OC-12 ATM/SONET Gateway, June 1992.
- [10] G. Minden, J. Evans, D. Petr, and V. Frost. An ATM WAN/LAN gateway architecture. In *2nd IEEE Symp. High Perf. Dist. Comp.*, pages 136-143, July 1993.

- [11] M. Prycker. *Asynchronous Transfer Mode: Solution for B-ISDN*. Ellis Horwood, 1991.
- [12] R. Jonkman. Netspec Documentation.
- [13] R. Menon, M. Swink, R. Jonkman. Raw Cell Driver kernel found as `vmunix.OTTO+JV+LOFI*` in `eckert/usr/sys/OTTO+JV+LOFI`.
- [14] Fore Systems, Inc. ATM SBus Adapter User's Manual, Rev D, 2.3.X Oct 1994
- [15] C. Thacker, M. Schroeder. AN2 Switch Overview. Digital Equipment Corporation Systems Research Center - Palo Alto, California, July 18, 1994.
- [16] Xilinx. The Programmable Logic Data Book, 1993.

Appendix A