Dynamic Modelling of A Two-Stage

ATM Tandem Banyan Switch

THAM HON LOKE

Faculty of Computer Science and Information Technology

University of Malaya

2002

ABSTRACT

This dissertation studies the dynamic modelling of Two-State ATM Tandem Banyan Switch. First of all, ATM technology and ATM switch are introduced, then the ATM switch's architecture and its functions are discussed in detail. A Two-State ATM Tandem Banyan Switch has been selected as a model to be simulated. MATLAB's Simulink tool has been used to dynamically model the ATM switch. The ATM switch simulated is a simple two-stage Banyan Switch. It consists of 2 Inports x 2 Outports, with a switching algorithm included. The algorithm covers all cell flow patterns that are possible in a 2 Inports x 2 Outports Banyan Switch. The objective is to keep the cell loss rate at a minimum level. Lastly, this project presents an example of dynamic simulation, which will examine the cell flow patterns, cell loss rate and the switch performance. Discussion and overall conclusion are made based on the simulation results.

ACKNOWLEDGEMENTS

I have learnt a great deal in the process of completing this dissertation. A lot of people have contributed to it. I would like to take this opportunity to express my appreciation to all of them.

First of all, I would like to thank Assoc. Prof. Dr. Selvanathan Narainasamy, my supervisor for his invaluable guidance and advice throughout the writing of this dissertation. Special thanks to the exam panel for taking their time to go through this dissertation.

Thanks to my fellow course mates for sharing their knowledge generously and their company throughout the time in University Malaya.

Not forgetting, my dearest family members, I am grateful to them for their continuous support in my pursuit of this course and my endeavour.

Last but not the least, to my girl friend who always stands by me, giving me the strength I need to complete this dissertation successfully.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

ATM	Asynchronous Transfer Mode
ABR	Available Bit Rate
CAC	Connection Admission Control
CBR	Constant Bit Rate
CDV	Cell Delay Variation
CLR	Cell Loss Ratio
CTD	Cell Transfer Delay
DEMUX	Demultiplexer
FIFO	First In First Out
GUI	Graphical User Interface
HEC	Header Error Control
IM	Input Module
ISDN	Integrated Services Digital Network
ITU	International Telecommunication Union
LAN	Local Area Network
MINs	Multistage Interconnection Networks
MUX	Multiplexer
NIST	National Institute of Standards and Technology
NNI	Network-Node Interface
NPC	Network Parameter Control
OAM	Operation and Maintenance
OM	Output Module
OPNET	Optimised Network Engineering Tool
QoS	Quality of Services
SM	Switch Management
SONET	Synchronous Optical Network
STM	Synchronous Transfer Mode
TBSA	Tandem Banyan Switch Architecture
TDM	Time Division Multiplexing
UBR	Unspecified Bit Rate

UNI	Unified Network Interface
UPC	Usage Parameter Control
VC	Virtual Connection
VCC	Virtual Channel Connection
VCI	Virtual Channel Identifier
VoIP	Voice over Internet Protocol
VPI	Virtual Path Identifier
WWW	World Wide Web

Chapter 1: Introduction to ATM Switch and ATM Simulation Software

This chapter briefly explains ATM technology and the Simulation concepts. ATM technology and ATM Switch are introduced here, this is followed by the introduction of the simulation concepts, Simulink tools and some examples of ATM simulation application. Lastly, the objective and the organisation of this dissertation will be introduced.

1.1 Introduction to Asynchronous Transfer Mode (ATM) Technology

Asynchronous Transfer Mode (ATM) is based on the switching and multiplexing technique selected by the International Telecommunication Union (ITU) for the broadband access for ISDN. ATM transports information in fixed-size packets that are commonly known as ATM cells. ATM cell is fixed-size packet that is 53 octets. The 53-byte cell is divided into 5-byte header and 48-byte data. When the physical data transfer becomes extremely reliable, as in ATM, it makes more sense to do the relaying of packets in the lower layers. This reduces the extra overhead associated with the more traditional networking protocols designed for unreliable physical links.

By using a fixed-size cell, ATM protocol contributes to the following advantages [CHIO 99]:

- Simple hardware implementation for switching ATM cells.

  This is a big advantage, because the system only needs to take care about 53-byte cells with 5-byte fixed headers. The data contain inside the header will provide information to ATM switch whether to route that particular cell or not when it comes to a congestion situation.

- Efficient use of available bandwidth for active connections.

  Fixed size header and data enable ATM switch to efficiently plan the switch logic by comparing the priority bit in the 5-byte header.

- Fair sharing of the available bandwidth among applications.

  Because of the fixed short size cell, ATM does not take over the bandwidth availability too long when compared with the protocol that has variable size data packets. It is also easier to

implement the fair share of the bandwidth when queuing is needed, because each cell represents a fixed amount of data.

- Robust and efficient end-to-end data loss recovery.

  As each cell that is received at the destination is labelled uniquely, loss cell can be easily identified, only this loss cell needs to be sent back to sender, this reduces the amount of data need to be re-sent.

- Suitability for all kinds of information traffic.

  By using 53-octet cells, ATM can give fair access to information traffic likes voice, video and data. No one traffic type can take over the line to an extent that hinders other traffic types. Sequential deliveries of cells allow certain types of information transfer to continue operation, even though a particular cell was lost. Guaranteed rate can provide quality service to certain traffic types like voice, that requires a steady, constant transfer rate.

- Better accommodation of LAN Bursty Traffic.

  LAN traffic is bursty in nature and uses whatever bandwidth is available. ATM, with its fair use of the bandwidth, guaranteed rate, and priority mode, will benefit LAN to extend its performance beyond other traffic types.

1.1.1 Quality of Service and Service Categories

A set of parameters is negotiated when a connection is set up on ATM networks. These parameters are used to measure the Quality of Service (QoS) of a connection and quantify end-to-end network performance at the ATM layer. The network should guarantee the QoS by meeting certain values likes Cell Transfer Delay (CTD), Peak-to-peak Cell Delay Variation (CDV) and Cell Loss Ratio (CLR). ATM is able to provide desired QoS for different applications. For example, voice is delay-sensitive but not loss sensitive, data is loss-sensitive but not delay sensitive, while some other applications may be both delay-sensitive and loss-sensitive [LUFA 95]. Those ATM service categories such as Constant Bit Rate (CBR) and Variable Bit Rate (VBR) services allow ATM to accommodate fixed-rate, least-line types of connections (i.e. voice and video) as well as bursty-style LAN connections. VBR service allows a certain sustainable for LANs. The Unspecified Bit Rate

(UBR) and Available Bit Rate (ABR) provide the options to use available bandwidth in the network [CHIO 99].

1.1.2 ATM Cell Format

ATM layer supports cell switching based on ATM cells. An ATM cell contains 53 octets and has two parts: a 5-octets header and a 48-octet payload or information field. Some of the important bits like source bit, destination bit and priority bit is contained in the header. ATM protocol has two interface definitions: UNI and NNI. A minor difference exists between the cell header format for UNI and NNI [CHIO 99], as show in Figure 1.1.

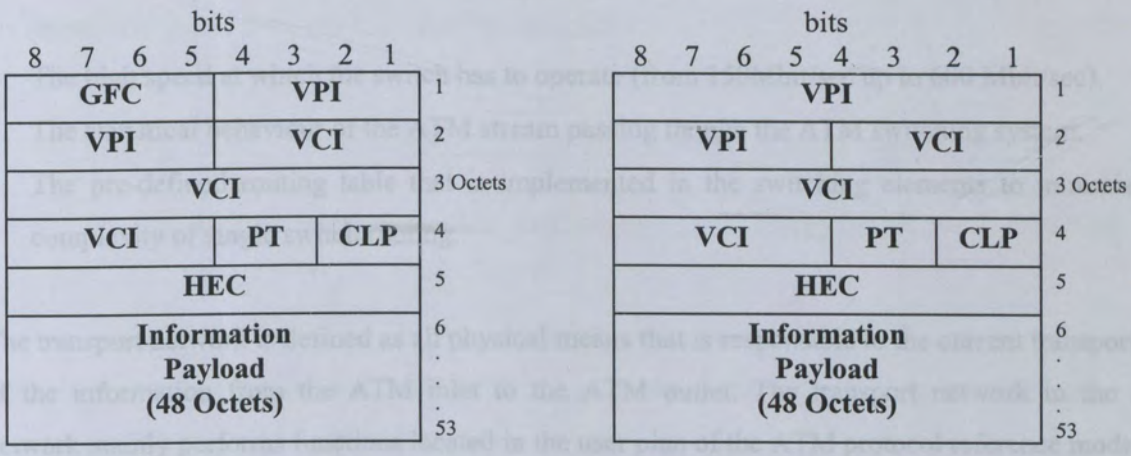


Figure 1.1 The cell format for UNI (left) and NNI (right)

ATM network service is connection-oriented by establishing a virtual connection (VC) between a source and a destination. The VC is dedicated to the pair of source and destination. More than one VC can be established through a physical link. The VCs are referred as virtual channel connections (VCCs). VCCs that have same end points are grouped together to become a single virtual path connection (VPC). Each VCC has virtual channel identifier (VCI) and pair with a virtual path identifier (VPI), the VCC will have a unique connection identifier [TAYS 00]. In this dissertation, VCI can be considered as the source bit for the Inport or destination bit for Outport.

1.2 ATM Switch

ATM switches are used to connect ATM host and networks. Each switch contains a routing table that consists of in/out pair of switch port and connection identifiers. Switching happens in the switches where incoming channels will be mapped to appropriate outgoing channels.

An ATM switching system is much more than a fabric that simply routes and buffers cells, rather it comprises an integrated set of modules. Switching systems not only relay cells, but also perform control and management functions. Furthermore, they must support a set of traffic control requirements Thus a good design of ATM switching architecture takes into consideration the following three factors [BIRA 94]:

- The high speed at which the switch has to operate (from 150Mbit/sec up to 600 Mbit/sec).
- The statistical behaviour of the ATM stream passing though the ATM switching system.
- The pre-defined routing table that is implemented in the switching elements to minimise the complexity of single switch routing.

The transport network is defined as all physical means that is responsible to the current transportation of the information from the ATM inlet to the ATM outlet. The transport network in the ATM network mainly performs functions located in the user plan of the ATM protocol reference model.

The control part of the switch is that which controls the transport network. It decides for instance, which inlet to connect to which outlet. The decision is based on incoming signalling information. The control network mainly performs functions located in the control plane of the ATM protocol reference model.

1.2.1 ATM Switch Functions

ATM is connection oriented. All cells belong to a virtual connection pre-established by the transport network. The ATM cell is 53 bytes long, built of 48 payload bytes and a 5-byte header. Each cell's header contains a VCI (Virtual Channel Identifier) that identifies the virtual connection to which the cell belongs.

ATM switch has several main tasks [BIRA 94]:

- VCI Translation

  The established connection on the ATM network defines the virtual path through different switches across the network. The VCI is local to each switch port. As each cell travels across an ATM switch, the VCI is translated into a new value. The switch has to build a new cell header containing the new VCI (any possibly new VPI – virtual path identifier).

- Switching – Cell transport from its input to its output.

  The transportation of the information (cell) from an incoming logical ATM channel (Inport) to an outgoing logical ATM channel (Outport) is also the responsibility of the ATM switch. Two identifiers characterise the logical ATM channel:

  a) The physical Inport/Outport which is characterised by a physical port number.

  b) The logical channel on the physical port, which is identified by the VCI and/or the VPI.

In order to provide the switching function, both physical and logical identifiers of the incoming cell have to related to physical and logical identifiers of the outgoing cell. Two functions have to be implemented in the ATM switching system.

The first function is the space switching function. The space switching is the one which allows the connection between inputs. An important aspect of space switching is the internal routing. This means how the information is routed internally in the switch. The internal structure of the switch must allow connect between input and output.

The second function is time switching. Since ATM is working in an asynchronous mode, cells, which had arrived in various time slots from the different inputs, can be delivered from different outputs in different time slots (There is no time identifier in ATM as it is in STM). Since there is no pre-assigned time slot connection, a contention problem arises if more than two logical channels are connected to the same output at the same time slot. However, implement a queuing function in the ATM switch system will solve this problem in the ATM switch.

1.3 ATM Switch Simulation

ATM Switch Simulation helps form an idea to the ATM Switch design in the complex network architectures and topologies. Even though the advent of high speed ATM switching solutions, ATM switch simulations allow designers to make informed decisions without need to invest in the physical hardware. Designers can carry out the performance related studies using simulation and therefore avoid the burden of the "trial and error" implementations.

1.3.1 Simulation

Simulation is the basic tool to help making a correct decision. Decisions are formulated based on the information resulting from the simulation A number of factor influences the probability of making a correct decision. These are summarised as below [OPNE 99]:

- The level of understanding of a problem - A problem should be well defined and manageable. A clear understanding of the problem is essential before a simulation model can be developed.

- Correct Model - It is paramount that the model correctly simulates the problem. A software model may be often be syntactically correct but might not accurately simulate the problem area being addressed. It is essential that the model being designed and simulated in such a way that allow the various experiments to be carried out and an appropriate simulation results to be generated.

- Interpretation of Results - The simulation model simply produces output data. This data must be manipulated and interpreted by the developer. The correct interpretation of this data is dependent on the usefulness of the output data and also the user's understanding of statistical methods.

1.3.2 Simulation Software

In the research field and software simulation market, quite a number of simulation software has been developed for ATM Network simulation, This simulation software allows designers or network planners to simulate network that they plan to set up. Simulation of ATM switch is normally part of the module in the simulation software. OPNET Simulator and ATM-NIST Network Simulator are the two examples.

OPNET, or Optimised Network Engineering Tool, can be described as a set of decision support tools, providing a comprehensive environment for the specification, simulation and performance analysis of communication networks, computer systems and applications, and distributed systems. OPNET has been designed to support the modelling and simulation of a large range of communication systems. Discrete event simulations are used as the means of analysing system performance and their behaviour. This sophisticated package comes complete with a range of tools which allows developers specify models in great detail, identify the elements of the model of interest, execute the simulation and analyse the generated output data [OPNE 99].

The key features of OPNET are summarised here as below:

- Objects Orientation - OPNET is object oriented, where each object has a defined set of attributes. These configurable attributes result in a highly flexible development environment.

- Hierarchical Modelling - OPNET perform modelling in a hierarchical approach, to describe any communication network. Each level of the hierarchy, to describes different aspects of the model being simulated. Models at the higher-level, to use model developed at one level lower of the hierarchy. This lead to a highly flexible simulation environment where generic models can be developed and it help to increase the reusability and expandability.

Another example of ATM network simulator is ATM NIST Network Simulator, it was developed at the National Institute of Standards and Technology (NIST) to provide a flexible test-bed for studying and evaluating the performance of ATM networks.

The ATM NIST Network Simulator allows users to create different network topologies, set the parameters of component operation, for example, users can select constant bit rate data to be generated, number of byte of data to be send, start time, end time and etc. All these parameters can be saved as different simulated configurations. While the simulation is running, various instantaneous performance measures, for example cell received and cell loss rate, can be displayed in graphical/text form or saved to files for subsequent analysis [NIST 94].

There are two major uses for the simulator. The first is as a tool for ATM network planning and the second is as a tool for ATM protocol performance analysis.

- As a planning tool, a network planner can run the simulator with various network configurations and traffic loads to obtain statistics such as utilisation of network links and throughput rates of virtual circuits.

- As a protocol analysis tool, a researcher or protocol designer could study the total system effect of a particular protocol. For example, one could investigate the effectiveness of various flow control mechanisms for ATM networks and address mechanisms such as: fair bandwidth allocation, protocol overhead, bandwidth utilisation, etc.

1.4 Introduction to Simulink

Simulink is an extension to Matlab that uses an icon driven interface for construction of block diagram representation of a process. Matlab integrates computation, visualisation, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Matlab provide an interactive system that allows many technical-computing problems, such as mathematics formulation in the block function to be solved.

Simulink is an interactive tool that has a capability for modelling, simulating, and analysing dynamical system. Using Simulink, virtual prototypes can be built and tested quickly. This will help to minimise the effort of exploring the design concepts at any level of detail. Simulink provides an interactive, block diagram environment for modelling and simulating dynamic system. It includes an extensive library of predefined blocks that can be used to build graphical models of the system. Supported model type includes linear, non-linear, continuous time, discrete-time, multi-rate, conditionally executed, and hybrid system. Simulink provides a graphical user interface (GUI) for building models as block diagram, using click-and drag mouse operations. A comprehensive block library of sinks, sources, linear and non-linear components, and connectors has been included for the building of block diagram. Models are hierarchical, thus, model can build using both top-down and bottom-up approaches to create a simplified view of components or subsystem [SIMU 99].

Simulink provides a complete set of modelling tools that can be used to develop details block diagram model of a system. Features such as block libraries, hierarchical modelling, signal labelling, and subsystem "masking" provide a powerful set of capabilities for creating, modifying, and

maintaining block diagrams. The Simulink block library consists of over one hundred blocks, providing an extensive set of standard components. After studying various network simulation tools, it was concluded that Simulink would be an ideal tool for dynamically modelling the ATM switch operation.

1.5 Objectives of the Dissertation

The main objective of this dissertation is to dynamically model a simple ATM Tandem Banyan Switch. Different bit rate cell is flowed through the switch, and cell received at the switch's outport are being monitored for analysis. Throughout this dissertation, we have to achieve the objectives below:

- To discuss various ATM switching functions and requirements.
- To focus in depth, the cell switch fabric functionality.
- To study the switch design principle.
- To study the ATM Tandem Banyan Switch, focusing on the cell switch fabric and its sub components.
- To model the cell fabrics sub component using MATLAB Simulink software.
- To develop a dynamic modelling of a two-state ATM Tandem Banyan Switch simulation software.
- To run the simulation software and monitor the simulation result, change the process parameters and monitor the cell flow pattern change, compare and analyse the result.
- To discuss the results and conclude on the overall performance of the dissertation.

1.6 Scope of the Project

The followings are the scope of the project.

- The ATM switch consists of 2 Inports x 2 Outports.
- The ATM switch is Two-state Tandem Banyan type.
- Constant Bit Rate type and Variable Bit Rate type data are generated to pass through the ATM switch.
- Cell received at Outport can be monitored for its contents. For example, type of data, cell bit rate and Inport address.

1.7 The Structure of the Dissertation

The structure of this dissertation shall organise as below:

Chapter 1 Introduction to ATM switch and ATM simulation software

The ATM technology, ATM switch and ATM Simulation Software will be explored in this chapter, so that basic knowledge and concepts can be gained before go in depth at the following chapter.

Chapter 2 ATM switch architecture, functionality, and design principle.

ATM switch architecture, functionality and design principle are discussed here. The parameter that determines the performance of the switch is introduced here. Two type of Banyan Network is introduced, that is Tandem Banyan Switch and Knockout Switch.

Chapter 3 Introducing to MATLAB Simulink tool.

The MATLAB Simulink tool is explained in this chapter. This includes the introduction to the Simulink tool, the advantage of using Simulink tool, the modelling concept, the Simulink toolbox and its basic Simulink block that isused in this project.

Chapter 4 Development of ATM switch simulation using MATLAB Simulink.

Development of ATM switch simulation using MATLAB-Simulink is discussed in detail. This includes determination of the simulation requirement and assumption, structure of the ATM switch functional block, determines the simulation parameter, builds the function block using Simulink tool, and test runs the simulation.

Chapter 5 Discussion and Conclusion

The testing result of the switch simulation is discuss and analyse here. Conclusion will be made on the overall performance of the switch simulation, which is run by MATLAB-Simulink. Area of enhancement and suggestion will be make if there is any.

Chapter 2: ATM Switch Architecture, Functionality and Design Principle

In the first chapter, basic knowledge of ATM technology, ATM switch, simulation and Matlab-Simulink Tool have been introduced. Chapter 2 will explain the ATM switch in more detail. This includes ATM switch function, various ATM switch architecture, cell switch fabric and designs principle.

2.1 Switching Functions

An ATM switch contains a set of input ports and output ports, through which it is interconnected to users, other switches, and other network elements. It might also have other interfaces to exchange control and management information with special purpose networks. In general, the switch is assumed to perform cell relay and support control and management functions.

It is useful to examine the switching functions in the three-plane contexts, which is mentioned in the B-ISDN model [FAHM 95]:

2.1.1 User Plane

The main function of an ATM switch is to relay data cells from input ports to the appropriate output ports. The switch processes only the cell headers and the payload is carried transparently. As soon as the cell comes in through the input port, the Virtual Path Identifier/Virtual Channel Identifier (VPI/VCI) information is derived and used to route the cells to the appropriate output ports. This function can be divided into three functional blocks: the input module at the input port, the cells switch fabric that performs the actual routing, and the output modules at the output ports.

2.1.2 Control Plane

This plane represents functions related to the establishment and control of the VP/VC connections. Unlike the data cells, information in the control cells payload is not transparent to the network. The switch identifies signalling cells, and even generates some itself. The Connection Admission Control

(CAC) carries out the major signalling functions required. Signalling information may not pass through the cell switch fabric, or maybe exchanged through a signalling network such as SS7.

2.1.3 Management Plane

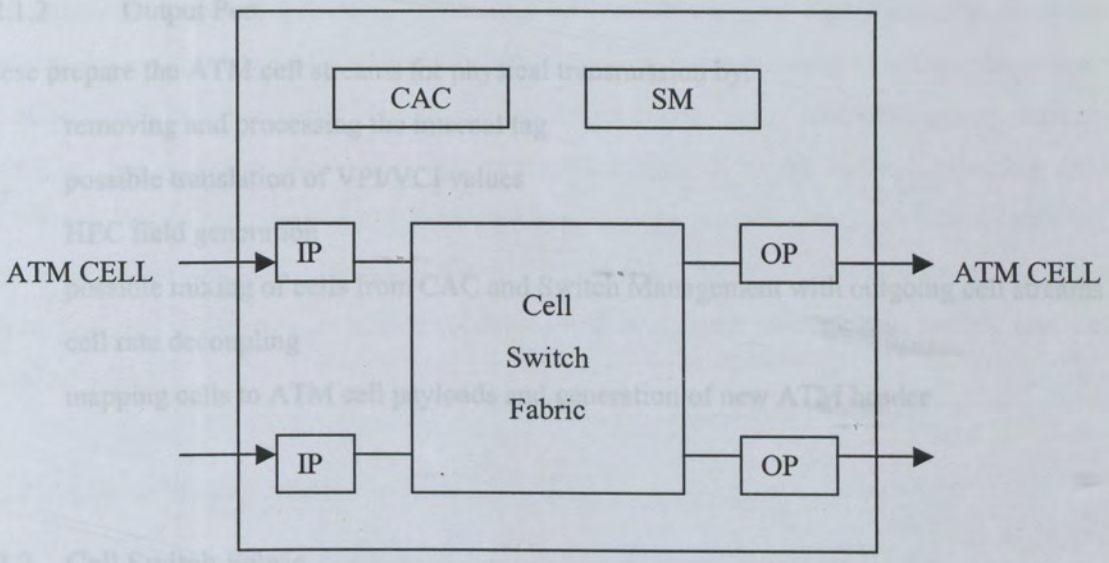
The management plane is concerned with monitoring and controlling the network to ensure its correct and efficient operation. These operations can be subdivided as fault management functions, performance management functions, configuration management functions, security management functions, accounting management function and traffic management function. These functions can be represented as being performed by the functional block Switch Management. The Switch Management is responsible for supporting the ATM layer Operations and Maintenance (OAM) procedures. OAM cells may be recognised and processed by the ATM switch. The switch must identify and process OAM cells, OAM cells may/may not pass through cell switch fabric.

2.1.4 Traffic Control Functions

The switching system may support connection admission control (CAC), usage/network parameter control (UPC/NPC), and congestion control. UPC/NPC functions are handled by the input modules, congestion control functions as handled by the Switch Management, while special buffer management actions (such as cell scheduling and discarding) are supervised by the Switch Management, but performed inside the cell switch fabric where the buffers are located.

2.2 A Generic ATM Switching Architecture

It will be useful to adopt a functional block model to simplify the discussion of various design alternatives. Switch functions defined previously can be divided according to its functional blocks such as input modules, output modules, cell switch fabric, connection admission control, and switch management. Figure 2.1 illustrates this switching model. These functional blocks are service-independent, and the partitioning does not always have well-defined boundaries between the functional blocks [CHEN 95].



IM = Input Port
OM = Output Port
CAC = Connection Admission Control
SM = Switch Management

Figure 2.1 A generic switching model

2.2.1 Switch Interface

2.2.1.1 Input Port

The Input Port or Inport terminates the incoming signal and extracts the ATM cell stream. These involve signal conversion and recovery, processing ATM cell header, cell delineation and rate decoupling. After that, for each ATM cell the following functions should be performed:

- error checking the header using the Header Error Control (HEC) field
- validation and translation of VPI/VCI values
- determination of the destination output port
- passing signalling cells to CAC, and OAM cells to Switch Management
- usage/network parameter control (UPC/NPC) for each VPC/VCC
- internal tag containing internal routing and performance monitoring information for use only within the switch

2.2.1.2 Output Port

These prepare the ATM cell streams for physical transmission by:

- removing and processing the internal tag
- possible translation of VPI/VCI values
- HEC field generation
- possible mixing of cells from CAC and Switch Management with outgoing cell streams
- cell rate decoupling
- mapping cells to ATM cell payloads and generation of new ATM header

2.2.2 Cell Switch Fabric

The cell switch fabric is primarily responsible for routing of data cells and possibly signalling and management cells as well. The next section (2.3) will focus on the cell switch fabric.

2.2.3 Connection Admission Control (CAC)

Establishes, modifies and terminates virtual path/channel connections. More specifically, it is responsible for:

- high-layer signalling protocols
- signalling ATM Adaptation Layer (AAL) functions to interpret or generate signalling cells
- interface with a signalling network
- negotiation of traffic contracts with users requesting new VPCs/VCCs
- renegotiations with users to change established VPCs/VCCs
- allocation of switch resources for VPCs/VCCs, including route selection
- admission/rejection decisions for requested VPCs/VCCs
- generation of UPC/NPC parameters

If the CAC is centralised, a single processing unit would receives signalling cells from the input modules, interpret them, and perform admission decisions and resource allocation decisions for all the connections in the switch. CAC functions may be distributed to blocks of input modules where each CAC has a smaller number of input ports. This is much harder to implement, but solves the connection control processing bottleneck problem for large switch sizes, by dividing this job to be performed by parallel CACs. Some of the distributed CAC functions can also be distributed among output modules which can handle encapsulation of high-layer control information into outgoing signalling cells.

2.2.4 Switch Management

Handles physical layer OAM, ATM layer OAM, configuration management of switch components, security control for the switch database, usage measurements of the switch resources, traffic management, administration of a management information base, customer-network management, interface with operations systems and finally support of network management.

Switch Management must perform a few basic tasks. It must carry out:

- specific management responsibilities
- collect and administer management information
- communicate with users and network managers
- supervise and co-ordinate all management activities

Management functions include:

- fault management
- performance management
- configuration management
- accounting management
- security management
- traffic management

Carrying out these functions entails a lot of intra-switch communication between the switch management and other functional blocks.

A centralised switch management can be a performance bottleneck, if it is overloaded by processing demands. Hence, switch management functions can be distributed among input modules, but a lot of co-ordination would be required. Each distributed input module switch management unit can monitor the incoming user data cell streams to perform accounting and performance measurement. Output module switch management units can also monitor outgoing cell streams [CHEN 95].

Below is shown a literature study on the ATM Switching Technique. It is presented at Section 2.3 (Cell Switch Fabric function) and Section 2.4 (Switch Design Principle). This study is based on the article, "A survey of ATM Switching Techniques" presented by Sonia Fahmy. The detail of the article can be obtained at http://www.cis.ohio-state.edu/~jain/cis788-95/atm_switching/ [FAHM 95].

2.3 The Cell Switch Fabric

The cell switch fabric is primarily responsible for transferring cells between other functional blocks, this include routing of data cells and possibly signalling and management cells as well. Other possible functions include:

- cell buffering
- traffic concentration and multiplexing
- redundancy for fault tolerance
- multicasting or broadcasting
- cell scheduling based on delay priorities
- congestion monitoring

2.3.1 Concentration, Expansion and Multiplexing

Traffic needs to be concentrated at the inputs of the switching fabric to better utilise the incoming link connected to the switch. The concentrator aggregates the lower variable bit rate traffic to higher bit rate for switching matrix to perform the switching at standard interface speed. The concentration ratio is highly correlated with the traffic characteristics, so it needs to be dynamically configured. The concentrator can also aid in dynamic traffic distribution to multiple routing and buffering planes, and duplication of traffic for fault tolerance. At the outputs of the routing and buffering fabric, traffic can be expanded and redundant traffic can be combined.

2.3.2 Routing and buffering

The routing and buffering functions are the two major functions performed by the cell switch fabric. The input module attaches a routing tag to each cell, and the switch fabric simply routes the arriving cells from its inputs to the appropriate outputs. Arriving cells may be aligned in time by means of single-cell buffers. Because cells may be addressed to the same output simultaneously, buffers are needed. Several routing and buffering switch designs have aided in setting the important switch design principles. All current approaches employ a high degree of parallelism, distributed control, and the routing function is performed at the hardware level.

Before examining the impact of the various design alternatives, we need to consider the essential criteria for comparing among them. The basic factors are:

1. throughput (total output traffic rate/input traffic rate)
2. utilisation (average input traffic rate/maximum possible output traffic rate)
3. cell loss rate
4. cell delays
5. amount of buffering
6. complexity of implementation

Traditionally switching has been defined to encompass either space switching or time switching or combinations of both techniques. The classification adopted here is slightly different in the sense that it divides the design approaches under the following four broad categories:

1. shared memory
2. shared medium
3. fully interconnected
4. space division

For simplicity, the ensuing discussion will assume a switch with N input ports, N output ports, and all port speeds equal to V cells/s. Multicasting and broadcasting will be addressed with the other issues in the next section, so they will be temporarily ignored in this discussion.

2.3.2.1 Shared Memory Approach

Figure 2.2 below illustrates the basic structure of a shared memory switch. Here incoming cells are converted from serial to parallel form, and written sequentially to a dual port Random Access Memory. A memory controller decides the order of cells read out of the memory, based on the cell headers with internal routing tags. Outgoing cells are de-multiplexed to the outputs and converted from parallel to serial form.

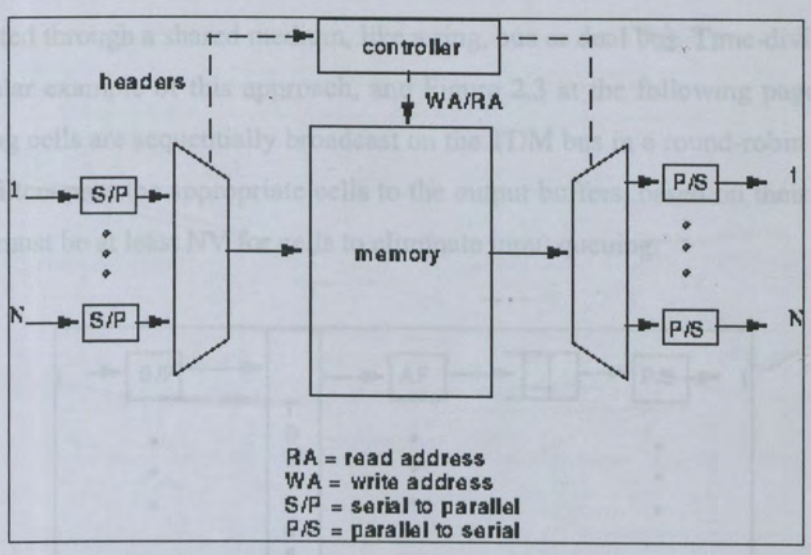


Figure 2.2 Basic structure of a shared-memory switch

This approach is an output queuing approach, where the output buffers, all physically belong to a common buffer pool. This approach is attractive because it achieves 100% throughput under heavy load. The buffer sharing minimises the amount of buffers needed to achieve a specified cell loss rate. This is because if a large burst of traffic is directed to one output port, the shared memory can absorb as much as possible of it.

This approach, however, suffers from a few drawbacks. The shared memory must operate N times faster than the port speed because cells must be read and written one at a time. As the access time of memory is physically limited, the approach is not very scalable. The product of number of port multiplier port speed NV is limited. In addition, the centralised memory controller must process cell headers and routing tags at the same rate as the memory. This is difficult for multiple priority classes, complicated cell scheduling, multicasting and broadcasting.

2.3.2.2 Shared Medium Approach

Cells may be routed through a shared medium, like a ring, bus or dual bus. Time-division multiplexed buses are a popular example of this approach, and Figure 2.3 at the following page illustrates their structure. Arriving cells are sequentially broadcast on the TDM bus in a round-robin manner. At each output, address filters pass the appropriate cells to the output buffers, based on their routing tag. The TDM bus speed must be at least NV for cells to eliminate input queuing.

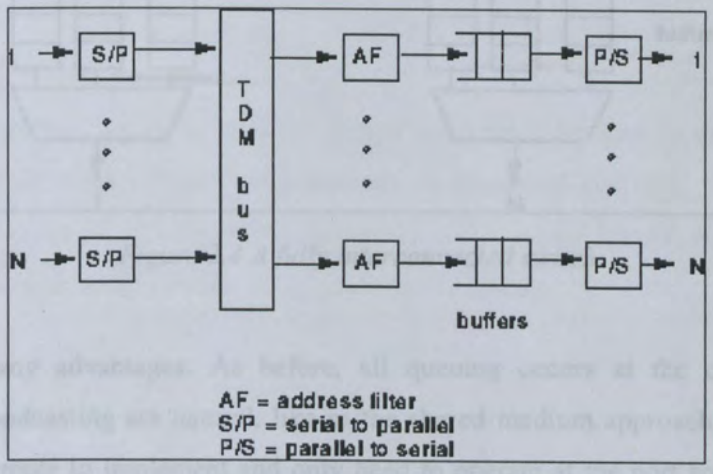


Figure 2.3 A shared bus switch

The outputs are modular, which makes address filters and output buffers easy to implement. Also the broadcast-and-select nature of the approach makes multicasting and broadcasting straightforward. As a result, many such switches have been implemented, such as IBM's Packetized Automated Routing Integrated System (PARIS) and PLANET, NEC's ATM Output Buffer Modular Switch (ATOM), and Fore Systems' ForeRunner ASX-100.

However, because the address filters and output buffers must operate at the shared medium speed, which is N times faster than the port speed, this places a physical limitation on the scalability of the

approach. In addition, unlike the shared memory approach, output buffers are not shared, which requires more total amount of buffers for the same cell loss rate.

2.3.2.3 Fully Interconnected Approach

In this approach, independent paths exist between all N squared possible pairs of inputs and outputs. Hence arriving cells are broadcast on separate buses to all outputs and address filters pass the appropriate cells to the output queues. This architecture is illustrated in Figure 2.4 as show in the next page.

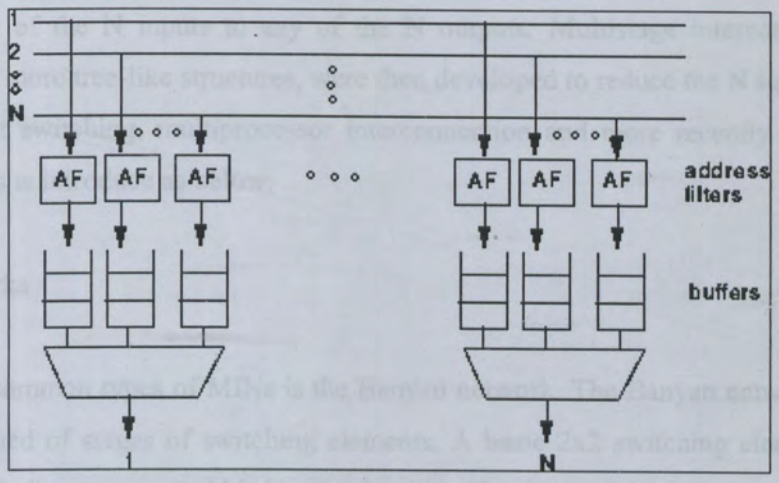


Figure 2.4 A fully interconnected switch

This design has many advantages. As before, all queuing occurs at the outputs. In addition, multicasting and broadcasting are natural, like in the shared medium approach. Address filters and output buffers are simple to implement and only need to operate at the port speed. Since all of the hardware operates at the same speed, the approach is scalable to any size and speed. Fujitsu's bus matrix switch and GTE Government System's SPANet are examples of switches in which this design was adopted.

Unfortunately, the quadratic growth of buffers limits the number of output ports for practical reasons. However, the port speed is not limited except by the physical limitation on the speed of the address filters and output buffers.

The *Knockout* switch developed by AT&T was an early prototype where the amount of buffers was reduced at the cost of higher cell loss. Instead of N buffers at each output, it was proposed to use only a fixed number of buffers L for a total of NxL buffers. This technique was based on the observation that it is unlikely that more than L cells will arrive for any output at the same time. It was argued that selecting the L value of 8 was sufficient for achieving a cell loss rate of 1/1 Million under uniform random traffic conditions for large values of N [ROBE 93].

2.3.2.4 Space Division Approach

The *crossbar* switch is the simplest example of a matrix-like space division fabric that physically interconnects any of the N inputs to any of the N outputs. Multistage interconnection networks (MINs) which are more tree-like structures, were then developed to reduce the N squared cross points needed for circuit switching, multiprocessor interconnection and more recently packet switching. Two type of MINs is introduce as below:

(I) Banyan networks

One of the most common types of MINs is the Banyan network. The Banyan network is constructed of interconnectioned of stages of switching elements. A basic 2x2 switching element can route an incoming cell according to a control bit (destination bit). If the control bit is 0, the cell is routed to the upper port address, otherwise it is routed to the lower port address.

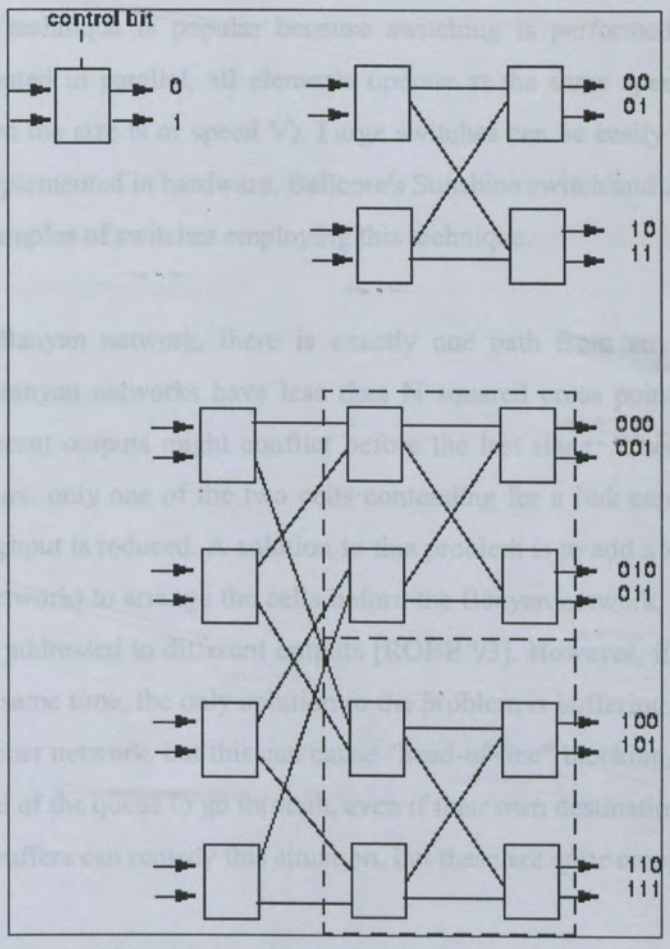


Figure 2.5 Switching element, 4x4 Banyan network and 8x8 Banyan network

To better understand the composition of Banyan networks, consider forming a 4 by 4 Banyan network. Figure 2.5 shows the step-by-step interconnection of switching elements to form 4x4, and then 8x8 Banyan networks. The interconnection of two stages of 2x2 switching elements can be done by using the first bit of the output address to denote which switching element to route to, and then using the last bit to specify the port. 8x8 Banyans can be recursively formed by using the first bit to route the cell through the first stage, either to the upper or lower 4x4 network, and then using the last 2 bits to route the cell through the 4x4 network to the appropriate output port.

In general, to construct a NxN Banyan network, the n^{th} stage uses the n^{th} bit of the output address to route the cell. For N = 2 to the power of n, the Banyan will consist of n = log to the base 2 of N stages, each consisting of N/2 switching elements [ROBE 93].

The Banyan network technique is popular because switching is performed by simple switching elements. Cells are routed in parallel, all elements operate at the same speed (so that there is no additional restriction on the size N or speed V). Large switches can be easily constructed modularly and recursively and implemented in hardware. Bellcore's Sunshine switch and Alcatel Data Networks' 1100 are just a few examples of switches employing this technique.

It is clear that in a Banyan network, there is exactly one path from any input to any output. Unfortunately, since Banyan networks have less than N squared cross points, routes of two cells addressed to two different outputs might conflict before the last stage. When this situation, called internal blocking, occurs, only one of the two cells contending for a link can be passed to the next stage, so overall throughput is reduced. A solution to this problem is to add a sort network (such as a Batcher bitonic sort network) to arrange the cells before the Banyan network. This will be internally non-blocking for cells addressed to different outputs [ROBE 93]. However, if cells are addressed to the same output at the same time, the only solution to the problem is buffering. Buffers can be placed at the input of the Batcher network, but this can cause "head-of-line" blocking, where cells wait for a delayed cell at the head of the queue to go through, even if their own destination output ports are free. First-In-Random-Out buffers can remedy this situation, but these are quite complex to implement.

Alternatively, buffers may be placed internally within the Banyan switching elements. Thus if two cells simultaneously attempt to go to the same output link, one of them is buffered within the switching element. This internal buffering can also be used to implement a backpressure control mechanism, where queues in one stage of the Banyan will hold up cells in the preceding stage by a feedback signal. The backpressure may eventually reach the first stage, and create queues at the Banyan network inputs. It is important to observe that internal buffering can cause head-of-line blocking at each switching element, and hence it does not achieve full throughput. Chen, Xing have designed a delta-based ATM switch with backpressure mechanism capable of achieving a high throughput while significantly reducing the overall required memory size [CHXI 94].

A third alternative is to use a recirculating buffer external to the switch fabric. This technique has been adopted in Bellcore's Sunshine and AT&T's Starlite wideband digital switch [ROBE 93]. Here output conflicts are detected after the Batcher sorter, and a trap network selects a cell to go through, and recirculates the others back to the inputs of the Batcher network. Unfortunately, this approach

requires complicated priority control to maintain the sequential order of cells and increases the size of the Batcher network to accommodate the recirculating cells.

As discussed before, output buffering is the most preferable approach. However, Banyan networks cannot directly implement it, since at most one cell per cell time is delivered to every output. Possible ways to work around this problem include:

- increase the speed of internal links
- route groups of links together
- use multiple Banyan planes in parallel
- use multiple Banyan planes in tandem or adding extra switching stages

2.4 Switch Design Principles

From the preceding section, it can be seen that each design alternative has its own merits, drawbacks, and considerations. The general design principles and issues exposed in the last section are analysed in more detail here.

2.4.1 Internal Blocking

A fabric is said to be internally blocking if a set of N cells addressed to N different outputs can cause conflicts within the fabric. Internal blocking can reduce the maximum possible throughput. Banyan networks are blocking, while TDM buses where the bus operates at least N times faster than the port speed are internally nonblocking. By the same concept, shared memory switches which can read and write at the rate of NV cells per second are internally non-blocking, since if N cells arrive for N different outputs, no conflicts will occur. Hence, to prevent internal blocking, shared resources must operate at some factor greater than the port speed. Applying this to Banyan networks, the internal links need to run square root of N times faster than the highest speed incoming link [ONVU 95]. This factor limits the scalability and throughput of the switch.

2.4.2 Buffering Approaches

Buffering is necessary in all design approaches. For instance, in a Banyan network, if two cells addressed to the same output successfully reach the last switching stage at the same time, output

contention occurs and must be resolved by employing buffering. The location and size of buffers are important issues that must be decided.

There are four basic approaches to the placement of buffers. These basic approaches are illustrated in Figure 2.6. A lot of comparative studies of these approaches together with numerous queuing analyse and result has been examined although each approach has its own merits and drawbacks, output queuing is the preferred technique so far.

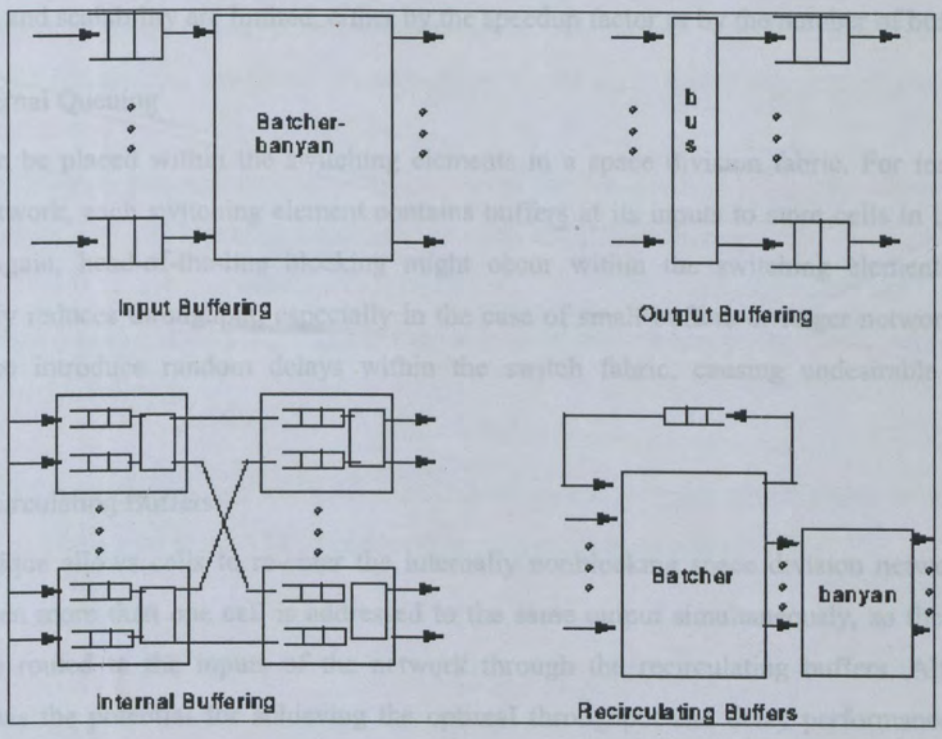


Figure 2.6 The various buffering approaches

2.4.2.1 Input Queuing

Buffers at the input of an internally nonblocking space division fabric (such as Batcher Banyan network) illustrate this type of buffering. This approach suffers from head-of-the-line blocking. When two cells arrive at the same time and are destined to the same output, one of them must wait in the input buffers, preventing the cells behind it from being admitted. Thus capacity is wasted.

Several methods have been proposed to tackle the head-of-the-line blocking problem, but they all exhibit complex design. Increasing the internal speed of the space division fabric by a factor of four, or changing the First-In-First-Out (FIFO) discipline are two examples of such methods.

2.4.2.2 Output Queuing

This type of buffering can be evident by examining the buffers at the output ports of a shared bus fabric. This approach is optimal in terms of throughput and delays, but it needs some means of delivering multiple cells per cell time to any output. Hence, either the output buffers must operate at some factor times the port speed, or there should be multiple buffers at each output. In both cases, the throughput and scalability are limited, either by the speedup factor or by the number of buffers.

2.4.2.3 Internal Queuing

Buffers can be placed within the switching elements in a space division fabric. For instance, in a Banyan network, each switching element contains buffers at its inputs to store cells in the event of conflict. Again, head-of-the-line blocking might occur within the switching elements, and this significantly reduces throughput, especially in the case of small buffers or larger networks. Internal buffers also introduce random delays within the switch fabric, causing undesirable cell delay variation.

2.4.2.4 Recirculating Buffers

This technique allows cells to re-enter the internally nonblocking space division network. This is needed when more than one cell is addressed to the same output simultaneously, so the extra cells need to be routed to the inputs of the network through the recirculating buffers. Although this approach has the potential for achieving the optimal throughput and delay performance of output queuing, its implementation suffers from two major complexities. First, the switching network must be large enough to accommodate the recirculating cells. Second, a control mechanism is essential to sequentially order the cells.

2.4.3 Buffer Sharing

The number and size of buffers has a significant impact on switch design. In shared memory switches, the central buffer can take full advantage of statistical sharing, thus absorbing large traffic bursts to any output by giving it as much as available shared buffer space. Hence, it requires the least

total amount of buffering. For a random and uniform traffic and large values of N, a buffer space of only 12 N cells is required to achieve a cell loss rate of 1/10 to the power of 9, under a load of 0.9.

For a TDM bus fabric with N output buffers, and under the same traffic assumptions as before, the required buffer space is about 90 N cells. Also the other output buffers cannot absorb a large traffic burst to one output, although each output buffer can statistically multiplex the traffic from the N inputs. Thus buffering assumes that it is improbable that many input cells will be directed simultaneously to the same output.

Neither statistical multiplexing between outputs nor any output can be employed with fully interconnected fabrics with N squared output buffers. Buffer space grows exponentially in this case.

After going through this section, the design of ATM Switch needs to take into consideration the design issue likes internal blocking, buffering approach and buffer sharing.

Banyan Switch model and output buffering approach could become an optimum solution to overcome the above design challenge.

In the next section, Banyan Switch model will be discussed in more detail where it brings to the decision to simulate Two State ATM Tandem Banyan Switch in this dissertation.

2.5 Banyan Network

A Banyan network is a blocking self-routing switch as shown below:

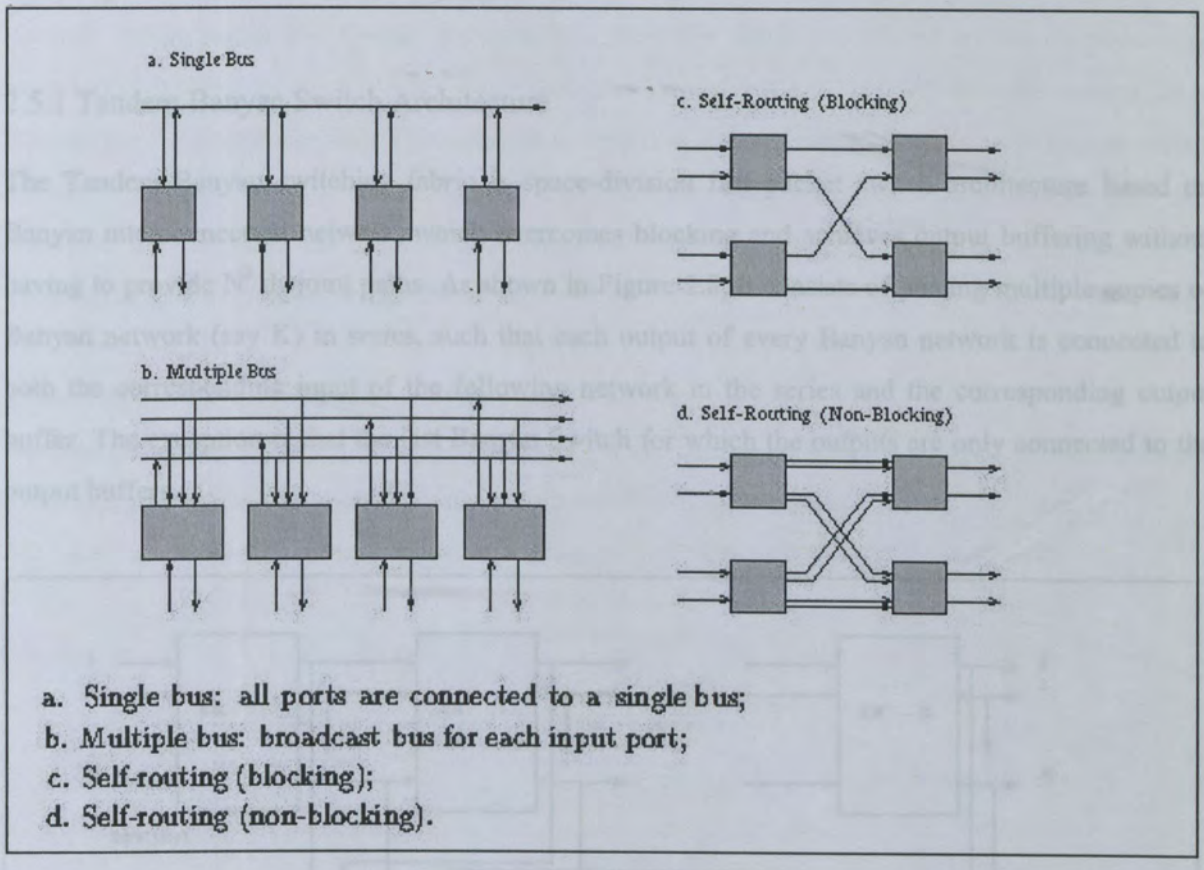


Figure 2.7 Example of Switch Architecture

It consists of $k = \log_2 N$ stages (assuming N, the number of ports, is a power of 2). Each stage consists of N/2 binary switching elements. The switch elements can assume Two-state, cross and bar, and interconnection lines between the stages are placed in such a way as to follow a unique path from each input to each output.

In all of these networks, the establishment of a path from any input line to an output line is accomplished in a distributed fashion, using a self-routing procedure. To route a packet to a destined output port (for example j, with binary representation $[d_1 d_2 \ldots d_k]$), the state of the switch element at stage s should be set such that the packet gets routed to its upper output if $d_s = 0$ and to its lower output if $d_s = 1$. Thus, the setting of an elementary switch is a function of a single bit of the

destination address. Up to N concurrent paths between inputs and outputs may be established simultaneously. When two packets arrive at inlets requesting the same output port there is a conflict and the throughput is decreased. In order to avoid this, a new scheme called *Tandem Banyan Switching Architecture* (TBSA) was proposed [VIST 99].

2.5.1 Tandem Banyan Switch Architecture

The Tandem Banyan switching fabric is space-division fast packet switch architecture based on Banyan interconnection network, which overcomes blocking and achieves output buffering without having to provide N^2 disjoint paths. As shown in Figure 2.8, it consists of placing multiple copies of Banyan network (say K) in series, such that each output of every Banyan network is connected to both the corresponding input of the following network in the series and the corresponding output buffer. The exception is that the last Banyan Switch for which the outputs are only connected to the output buffers.

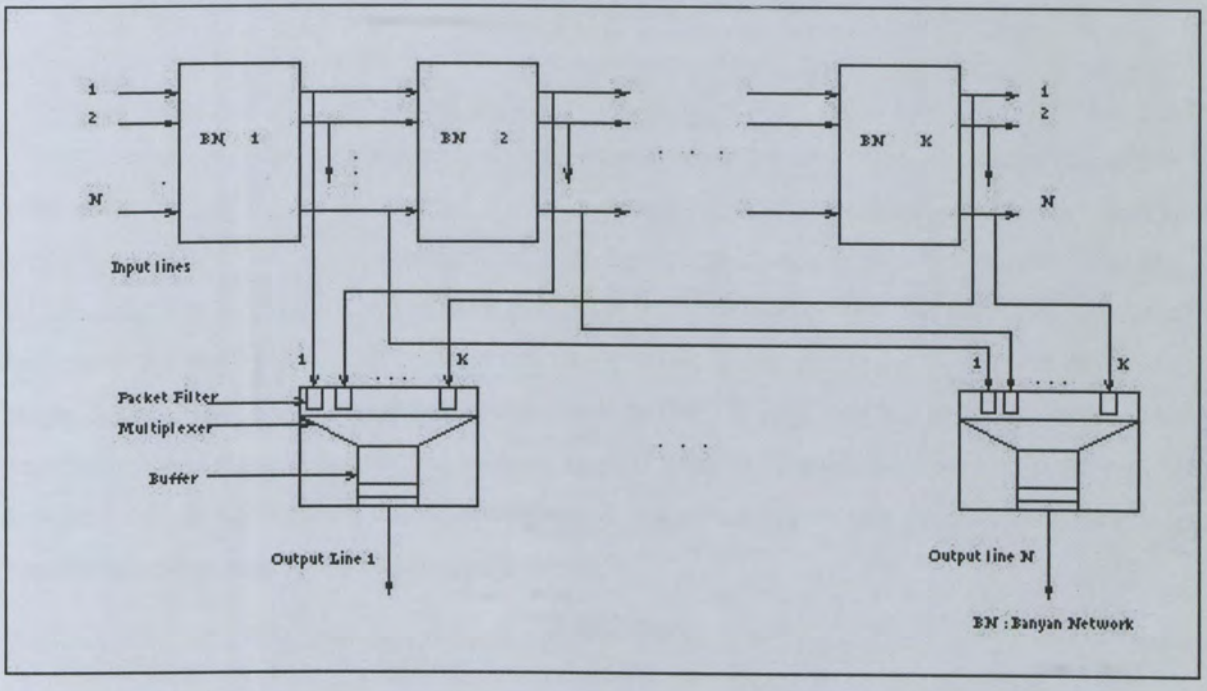


Figure 2.8 ATM Tandem Banyan Switch Architecture

In this arrangement, upon a conflict between two packets at some switching element, one of the two packets is routed properly, while the other is routed in to the next stage Banyan network.

Furthermore, whenever a packet is routed to the next stage of the Banyan network, it is marked as such and whenever there is a conflict between a properly routed packet and a misrouted packet, the former is always honoured. At the output of the 1st stage Banyan network, the packets that have reached their respective destinations distinguished from those that get route to the next stage Banyan network output ports. The former are extracted from the fabric and placed in the corresponding output port buffers, while the latter have their mark removed, and are fed into the second stage Banyan for further processing. This process is repeated through the K Banyan networks in series. Unsuccessful packets at the last Banyan are lost. Note that the load on successive Banyan networks decreases and so does the likelihood of conflicts. With a sufficiently large K, it is possible to decrease the packet loss to the desired level.

The novelty of the TBSA lies in two features:

(i) The placement of multiple Banyan networks in a series arrangement, so as to provide multiple paths from each input to each output and to achieve output buffering, and

(ii) Their operation in a novel way so as to achieves high performance.

2.5.2 Knockout Switching Architecture

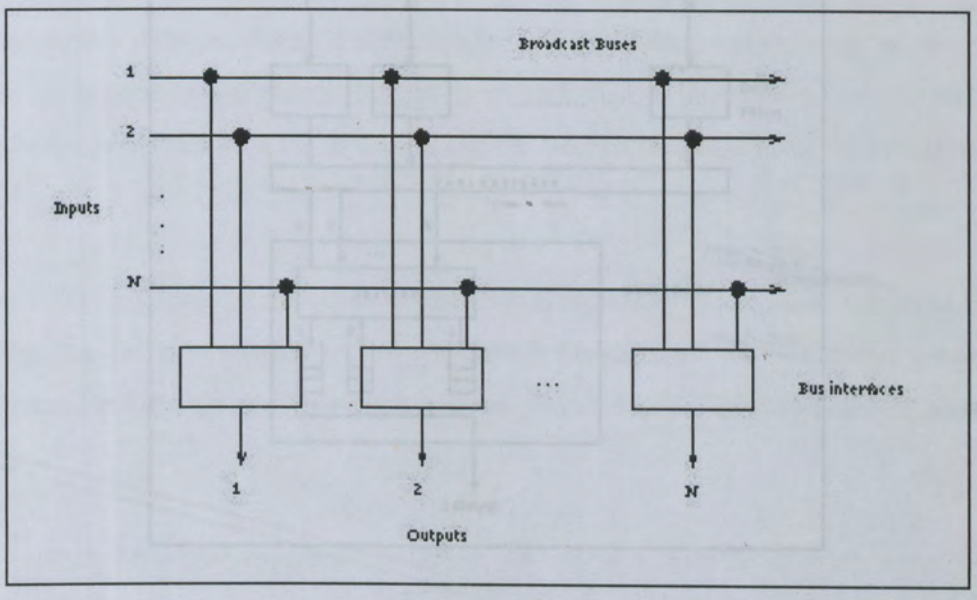


Figure 2.9 Knockout Switching Element

The knockout switch has N inlets and outlets, each operating at an equal speed. As shown in Figure 2.9, the transfer medium is composed of N broadcast buses for each inlet. An outlet has access to cells arriving on all inputs. Each of the N inlets puts cells on a separate broadcast bus on which each outlet gets access via a bus interface with N inputs. This means that the transfer medium is nonblocking and no cells are lost at the input of the bus interface. At one bus interface, several cells may arrive simultaneously at all destined to a single outlet. In the worst case, N cells are destined to a single outlet. Thus, the bus interface requires cell buffers. If zero loss has to be guaranteed in the transfer phase to the cell buffer, the memory must operate at N times the speed of each inlet. The knockout switch has reduced this operating speed, by an intelligent bus interface, which acts as a concentrator, with a non-zero cell loss probability.

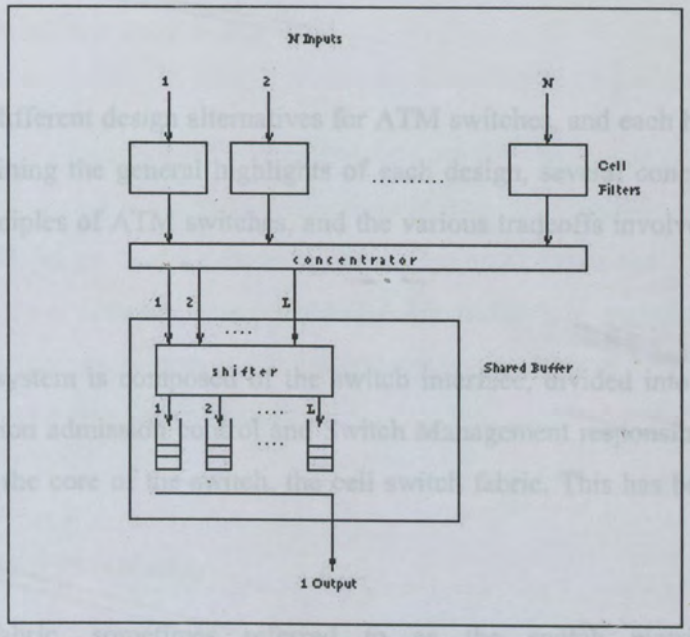


Figure 2.10 Knockout bus interface

The bus interface associated with each output is shown in Figure 2.10. At the top of the diagram, N cell filters are shown each connected to one of the N broadcast buses. These cell filters examine the address of each incoming cell. If destined to that specific outlet, the cell is passed to the concentrator, otherwise the cell is discarded. The next part of the bus interface is the concentrator from N inputs to L outputs (L less than or equal to N). Suppose k cells arrive simultaneously for the same outlet. After the concentrator, these k cells will arrive on outputs 1 to k of the concentrator, if k is less than or equal to L. If k is greater than L, then all L outputs of the concentrator will have cells, and k-L cells will be lost in the concentrator. The concentrator itself can be built from a very simple switch in which two inputs contend for a winner output.

2.6. Summary

There are numerous different design alternatives for ATM switches, and each has its own merits and drawbacks. By examining the general highlights of each design, several conclusions can be drawn about the design principles of ATM switches, and the various tradeoffs involved in selecting among them.

An ATM switching system is composed of the switch interface, divided into the input and output modules, the connection admission control and Switch Management responsible for connection and OAM functions, and the core of the switch, the cell switch fabric. This has been the main focus of this survey.

The cell switch fabric, sometimes referred to as the switch matrix, may consist of concentration/duplication, expansion/combination and buffer management. Buffer management is quite complicated due to the varying requirements of different QoS classes, which affect the cell scheduling and discarding policies, as well as the congestion control indication. The routing and buffering fabric, however, constitutes the heart of the switch.

The design principles of the routing and buffering fabric were developed after analysing the main switch design categories, namely the shared memory, shared medium, fully interconnected and space division approaches and their variations.

Two types of Banyan Switch architectures are introduced here, both architectures provide advantage of minimising the cell loss by routing the cell through a few stages of the Banyan network before arriving at the output port.

ATM Tandem Banyan Switch, which consists of Banyan network and output buffering, has been choose as a model to be simulated in this project. This model will enable reader to have an idea of how the Banyan network and its switching algorithm mechanism works. More detail will be discussed in Chapter 4.

Chapter 3: Simulation and Simulink

In Chapter 2, ATM switch functionality, switch architecture, Banyan Switch and design principle are discussed. Chapter 3 discusses some basic concepts in simulation. Simulation can be divided into 3 divisions, that is model design, model execution, and execution analysis. The ATM switch simulation will be taken through these three stages using MATLAB Simulink tool. Simulink tool is an ideal tool to perform a dynamic cell modelling and it has extensive components both reusable and custom made. Its graphic facility makes the process visible.

3.1 Introduction to Simulation

Decisions are formulated based on the information resulting from the simulation. With the advent of high-speed network solution likes ATM switch, simulation allows designers make-informed decisions without the need to invest in the new technologies. Designer can test their new designs and carry out performance related studies using simulation, therefore free from the burden of "trial and error" implementations.

3.1.1 Basic Concepts

Simulation has been defined by Shannon as "the process of designing a computerised model of a system (or process) and conducting experiments with this model for the purpose either of understanding the behaviour of the system or of evaluating various strategies for the operation of the system" [POOC93]. The foundation of simulation is the concept of modelling. Modelling involves building a simpler representation of phenomena. Modelling is an old concept. The invention of computer has changed the approach of model building by changing the model form. Algorithms replace equations and computer programs are written to mimic the real phenomena instead of developing the mathematical theory. Model building becomes the core process of simulation, without building a model, no simulation can be done.

Central to the simulation study is the idea of a system. Before modelling a system, one must understand what a system is. A system can be defined as "an orderly collection of logically related principles, facts or objects" or simply "a collection of objects and interactions". In the context of a

simulation study, a system is generally referred to as "a collection of object with a well-defined set of interactions among them". In view of these definitions, we can conclude that a model, which is static in nature, is only approximately real and accurate in a limited range. A model will not produce reliable information on every aspect of the modelled system. On the other hand, simulation, which involves digital computer programming, will give a more accurate simulated result. This is due to the fact that programming is able to take into consideration, various interactions of a modelled system effectively [TAYS 00].

3.1.2 Continuous versus Discrete

The term continuous and discrete applied to a system refer to the nature or behaviour of changes with respect to time in the system state. The states of continuous systems are changing continuously over the time. Systems that state changes occurring at precise points in time are called discrete system. Different simulation method is required for these two kinds of system. In simulating discrete system, the variable values change, just at a time of some discrete event or at a certain time point in simulation.

Example of the continuous system is the inflow and outflow of water in a reservoir. Example of discrete system is the arrival and departure of cars at a car park. There are systems called hybrid system where some of the system state may vary continuously, while others may vary discretely. Discrete event simulation can serve as an approximation of continuous simulation, if the time range of event occurred is set to be small.

3.1.3 Stochastic versus Deterministic

A deterministic system, is a system where its new state is completely determined by previous states and by the activity. A given activity will transit a state from one to another in a completely deterministic manner. A stochastic system is a system in which the effects of the activity varies randomly and hence, contains a certain amount of randomness in its transitions from one state to another.

3.1.4 Discrete Event Simulation

The following is the terminology frequently used in discrete event simulation:

- State: A variable characterising the system such as level of stock in inventory or number of jobs waiting for processing.
- Event: An occurrence at a point of time that may change the state of system such as arrival of a customer or start of working on a job.
- Entity: An object that passes through the system such as cars in an intersection or orders in a factory. Often an event (e.g., arrival) is associated with an entity (e.g., customer).
- Queue is not only a physical queue of people, it can also be a task list, a buffer of finished goods waiting for transportation or any place where entities are waiting for something to happen for any reason.
- Creating is causing an arrival of a new entity to the system in some future time.
- Scheduling is to assign a new future event to an existing entity.
- Random Variable is a quantity that is uncertain such as inter-arrival time between two incoming flights and number of defective parts in a shipment.
- Random variant is an artificially generated random variable.
- Distribution is the law that governs the probabilistic features of a random variable.

3.1.5 Event-Driven versus Time-Driven in Discrete Event Simulation

In event-driven simulation, simulation time is advanced in fixed increment called ticks. Time values are an increasing arithmetic sequence, $\Delta s = c$, where s denotes simulation time and c is a constant greater than zero. Short ticks can result longer simulation time in order to ensure accuracy. Longer simulation is needed because nothing may happen in short tick interval. Time-driven simulation is particularly appropriate for modelling continuous time system such as digital computer system. To make time driven simulation more efficient, the tick interval can be tailored to the level of activity occurring in the system. For example lengthen the tick interval when there is little activities.

In event-driven simulation, time is moved from one event to the next. The time sequence still increases monotonically but not in an arithmetic sequence as in time-driven simulation, Δs is greater than zero. The increment time of event-driven simulation is based on the occurrence of an event that represents a change in state. Hence, a great potential for acceleration exists in event-driven simulation

than in time-driven simulation. Event-driven simulation is more efficient for modelling even-oriented system such as telephone system.

3.1.6 Computer Simulation

Computer Simulation is a fundamental discipline for studying complex system [FSIH94]. It is the discipline of designing model of an actual or theoretical physical system, executing the model on a digital computer, and analysing the execution output. It can be subdivided into three areas:

- Model design
- Model execution
- Execution analysis

Model design involves creating a representation of the physical object in mathematical model. The mathematical model can be in a form such as declarative, functional, constraint, spatial or multimodel. Then the model will be programmed and operated on a computer by running the executable computer program. The program will update the state and event variables of the mathematical model as it steps through time. The end result of the operation will be gathered and analysed to produce a meaningful outcome.

3.2 Modelling and Simulation

3.2.1 Modelling and Simulation Cycle

There are five phases in the modelling and simulation cycle [OPNE 99] as show in below Figure 3.1.

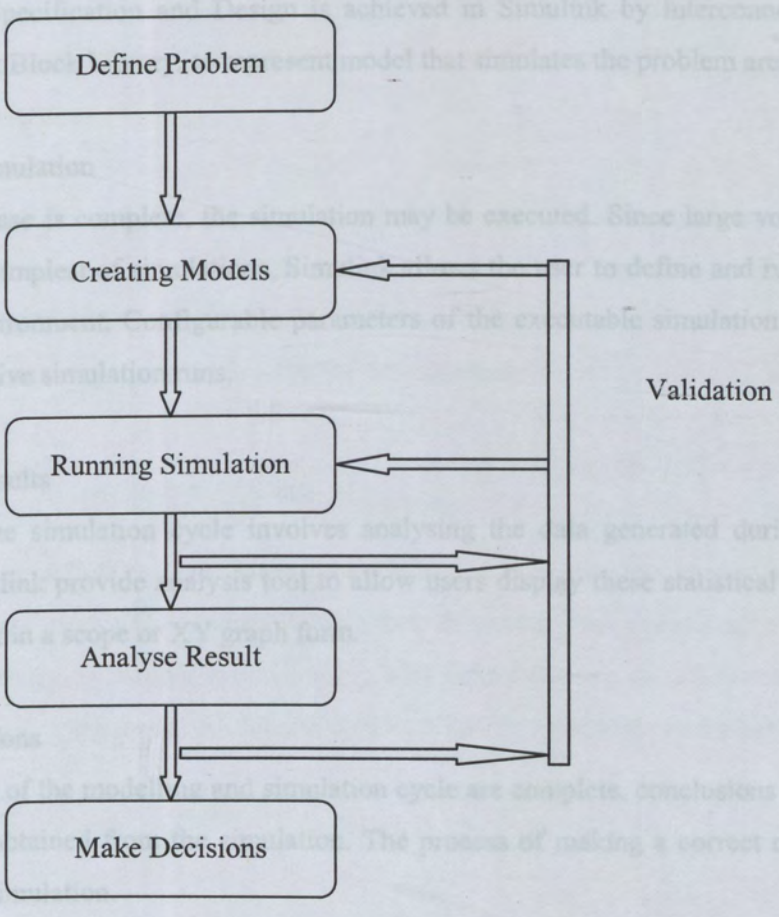


Figure 3.1 Modelling and Simulation Cycle

Simulink provides specific tool that assists the user through three of the five phases (e.g. the creating of model, the running of a simulation and the analysis of the output data).

Phase I: Define Problem

The single most important reason for simulation is to provide an insight into a problem area so that a decision can be made. Before a simulated model of a problem can be developed, it is essential that the problem area being addressed be well understood.

Phase II: Creating Model

The next phase involves the building of a correct model to accurately simulate the problem area. In this project, Model Specification and Design is achieved in Simulink by interconnecting various blocks in the Simulink Block Library, to represent model that simulates the problem area.

Phase III: Running Simulation

Once the previous phase is complete, the simulation may be executed. Since large volumes of data are generated by the simplest of simulations, Simulink allows the user to define and run simulations in a user-friendly environment. Configurable parameters of the executable simulation can be easily altered during successive simulation runs.

Phase IV: Analyse Results

The next phase of the simulation cycle involves analysing the data generated during successive simulation runs. Simulink provide analysis tool to allow users display these statistical results. Users can compare the result in a scope or XY graph form.

Phase V: Make Decisions

Once the other phases of the modelling and simulation cycle are complete, conclusions must be made based on the results obtained from the simulation. The process of making a correct decision is the primary objective of simulation.

While the process of validating a model has not been described as a simulation phase, it is an important aspect in the modelling and simulation cycle. The task of validating the model using Simulink takes place once the process model successfully compiles and executes. Model validation can involve the analysis of an executable simulation or the analysis of data generated from an executable simulation. Figure 3.1 depicts this with arrows leaving these phases. Before it can be determined whether or not the model needs to be rebuilt, the simulation may need to be re-executed. Figure 3.1 represents this with an arrow entering the "Execute Simulation" phase. Once it is determined that the model does not accurately simulate the problem area being addressed, the model is reconstructed. Eventually, the model will simulate the problem area and the process of making decisions can now take place.

The simulations models developed during preparation of this project have been validated through extensive testing. Each model performs a well-defined function (e.g. Generation of different type of cell data and queuing process before ATM fabric switch) which, when combined with other models, enables a simulation data cell pass through an ATM fabric switch.

3.2.2 Simulate ATM Tandem Banyan Switch Model

A 2 Inport by 2 Outport ATM Tandem Banyan Switch has been selected to be simulated. By using Simulink Software, simulation of a Tandem Banyan Switch can be done dynamically. To model the Tandem Banyan switching technique, understanding of Tandem Banyan switch architecture is very important. Below show the Figure 3.2 Architecture of an ATM Tandem Banyan Switch Fabric.

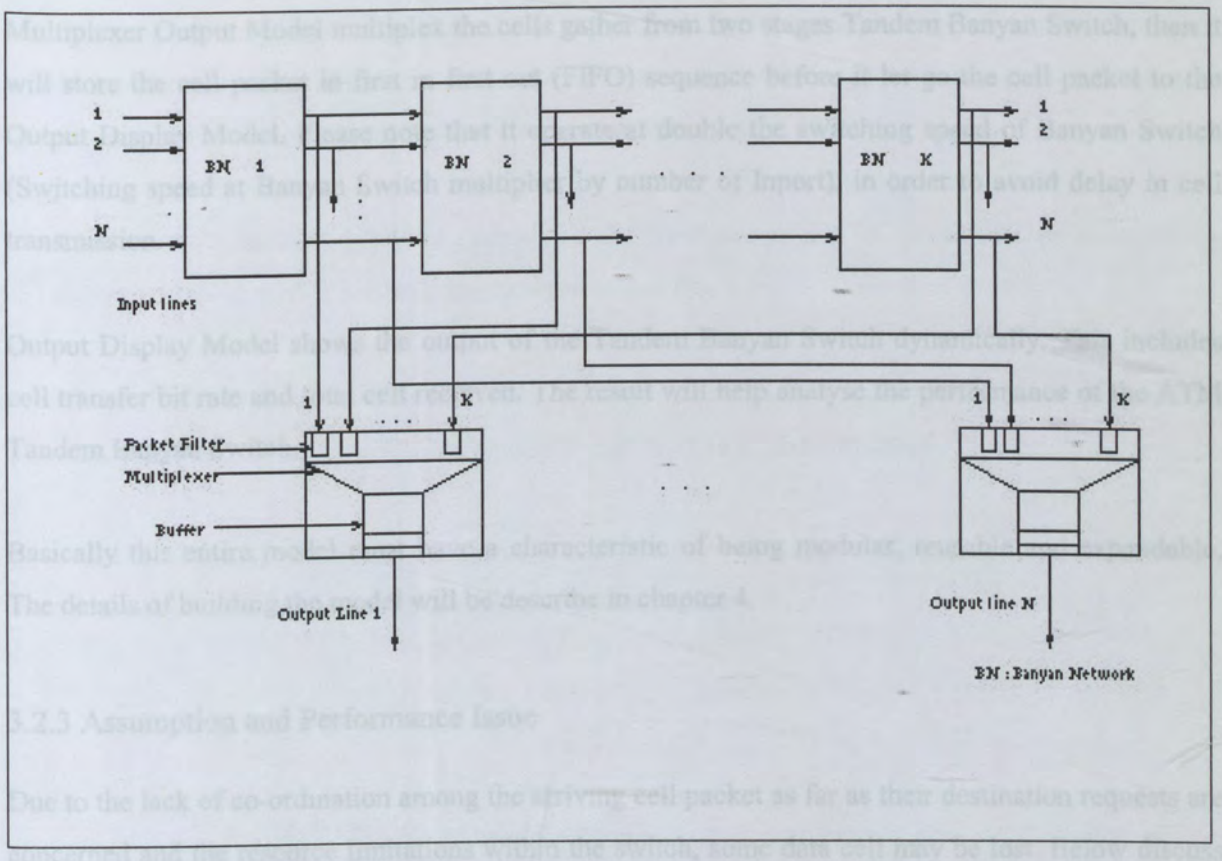


Figure 3.2 Architecture of an ATM Tandem Banyan Switch

Based on the above architecture, simulation model can be divided into four main models, namely Traffic Generator Model, ATM Banyan Switch Model, Multiplexer Output Model, and the Output Display Model.

The function of the Traffic Generator Model is to generate cell packets that have different bit rate, destination bit and total byte being generated. This will represent different characteristic of cell packet that pass through a cell switch fabric.

Banyan Switch Model will be built based on a 2 Inport x 2 Outport Banyan Switch architecture. This architecture allows model to be repeatedly built in series to represent Two-state of Tandem Banyan network. The Tandem Banyan Network will help gather all cells having internal blocking problem at the 1st stage of Tandem Banyan switch Model.

Multiplexer Output Model multiplex the cells gather from two stages Tandem Banyan Switch, then it will store the cell packet in first in first out (FIFO) sequence before it let go the cell packet to the Output Display Model. Please note that it operate at double the switching speed of Banyan Switch (Switching speed at Banyan Switch multiplier by number of Inport), in order to avoid delay in cell transmission.

Output Display Model shows the output of the Tandem Banyan Switch dynamically. This includes cell transfer bit rate and total cell received. The result will help analyse the performance of the ATM Tandem Banyan Switch.

Basically this entire model must have a characteristic of being modular, reusable and expandable. The details of building the model will be describe in chapter 4.

3.2.3 Assumption and Performance Issue

Due to the lack of co-ordination among the arriving cell packet as far as their destination requests are concerned and the resource limitations within the switch, some data cell may be lost. Below discuss some of the assumption and performance issue that should be considered while simulating the switch technique.

- Internal Blocking: Internal blocking occurs when two cells, destined for different output ports, contend over the same internal link. A switch that does not suffer from internal blocking is called nonblocking. Congestion at link can be one of the reasons, which cause blocking condition happen.

- Output Blocking: Output blocking happens when more than one cell requests the same output port at the same time. A blocking switch suffers from both internal blocking and output blocking. Nonblocking switches can be further classified into output blocking and output-nonblocking architectures. An output-nonblocking switch is able to clear all incoming cells to their requested output ports, instantaneously and simultaneously.

- Speed Up: The speed up of switching system can be implemented in time domain or in space domain. In the former, the switch is said to have a speed up factor of S if the internal switching

fabric operates S times faster than the external lines. In other words, the cell transmission time within the switching fabric is equal to 1/S of that outside fabric. In the latter implementation, S disjoint paths are provided simultaneously to any output port.

- Throughput: The throughput is defined as the average number of cells that are successfully delivered by the switch per cell-duration per output line.

- Delay: The delay is defined as the average time (in cell duration) a cell spends in the switch from time it arrives, till the time it is successfully delivered to its requested line.

- Cell loss probability: Cell loss probability is defined as the fraction of cell loss within the switch. This might occur as a result of blocking or buffer fullness.

3.3 Computer Simulation using Simulink

3.3.1 What is Simulink

Simulink is a software package for modelling, simulating, and analysing dynamical systems. It supports linear and non-linear systems, modelled in continuous time, discrete time, or a combination of the two. Systems can also be multiple rates, which means it can have different parts that are sampled or updated at different rates.

Simulink provides a graphical user interface (GUI) for building models as block diagrams by using click-and-drag mouse operations for modelling. This is the main advantage of Simulink over other simulation packages that require the formulation of differential equations and difference equation in a language or program. Simulink includes a comprehensive block library of sinks, sources, linear, and non-linear components and connectors that enable customised block to be created.

Model are hierarchical built by using both top-down and bottom-up approaches. The system can be viewed at a higher level, then double-click on blocks to go down through the levels to see increasing level of model details. This approach, provides insight into how a model is organised and how the blocks are interacted.

Simulation can be started after defining a model, either from Simulink menus or by entering commands in MATLAB's command window. Using display blocks provided, the simulation result can be seen while the simulation is running. In addition, the parameters can be changed to see the effects immediately, for "what if" exploration. The simulation results can be put in the MATLAB workspace for post processing and visualisation.

Simulink also includes model analysis tool likes linearisation and trimming tools, which can be accessed from the MATLAB command line, plus the many tools in MATLAB and its application toolboxes. And because MATLAB and Simulink are integrated, models can be simulate, analyse and revised in either environment at any point [SIMU 99].

3.3.2 Why use Simulink?

- By using Simulink, models can be built from scratch easily, or take an existing model and add to it.
- Simulation in Simulink are interactive, parameters can be changed "on the fly" and immediately can see what happens.
- All of the analysis tools in MATLAB can be accessed instantly through Simulink.
- With Simulink, one can move beyond idealised linear models to explore more realistic non-linear models such as factoring in fraction, air resistance and other things that describe real world phenomena.
- New blocks can be created and using masks to customise their appearance and use in Simulink.
- Subsystems where execution depends on triggering signal aids in simplifying the model design.
- Simulink debuggers are provided to debug Simulink models in an effective way.

3.3.3 Application in Simulink Toolboxes

One of the key features of Simulink is that it is built on top of MATLAB. As a result, Simulink users have direct access to the wide range of MATLAB-based tools for generating, analysing, and optimising systems implemented in Simulink. Here are some examples of professional toolboxes that are available from The MathWorks, Inc.

The Simulink Toolbox.
The Fuzzy Logic Toolbox.
The Image Processing Toolbox.
The Neural Network Toolbox.
The Signal Processing Toolbox.
The Control System Toolbox.

Below, shows an example of the contents of the Simulink block library. It contains of Continuous, Discrete, Math, Functions & Tables, Math, Nonlinear, Signals & Systems, Sinks and Sources block library.

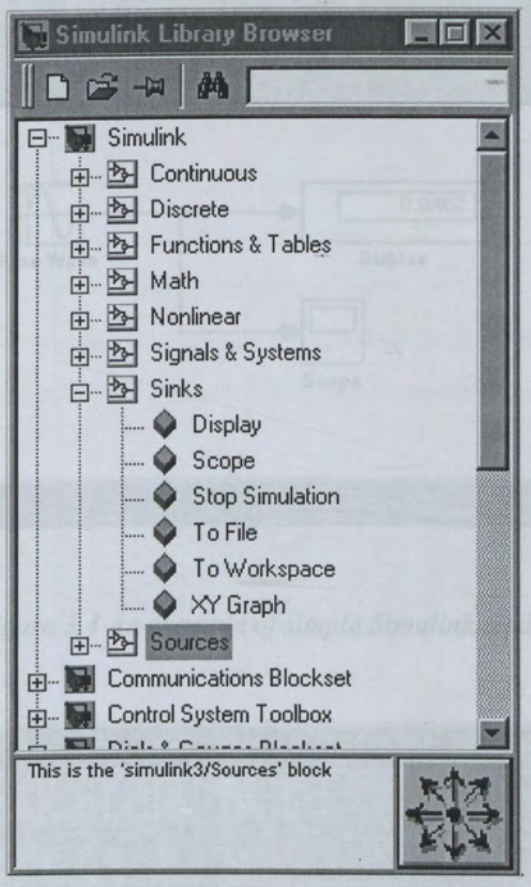


Figure 3.3 Simulink Block Library

Simulink is graphical and interactive. Models can be built easily by dragging the block inside the Simulink toolbox library to the Simulink window. Different block from different toolbox can be linked in the simulink window to create relationship. For example, first, drag a Sine Wave block and a Scope block to the Simulink window, second, connects a line in between each other, then run the simulation, the output Scope will show a sine wave output in wave form which will move against the time interactively.

Below show an example of how a simple model (Figure 3.4) represents an input, sine wave generator link to an output, that is a display and a scope (Figure 3.5).

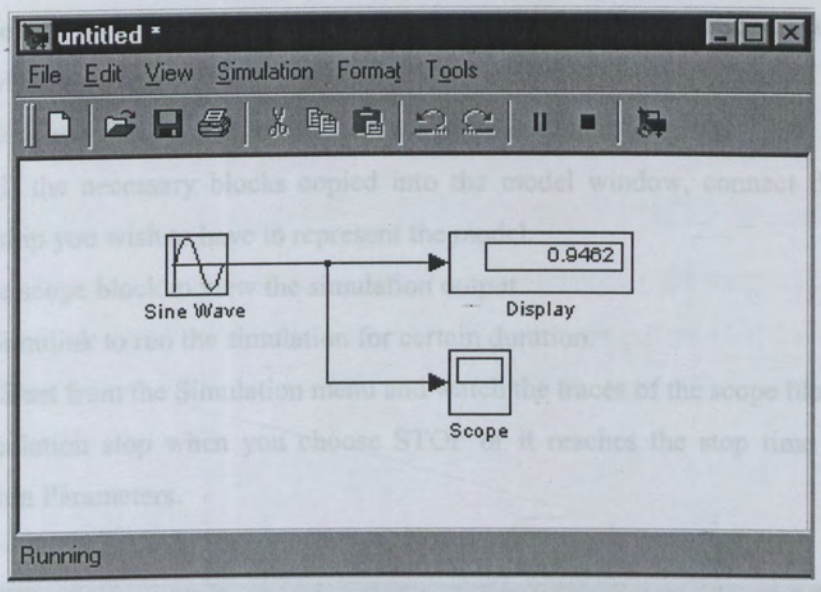


Figure 3.4 An example of simple Simulink model

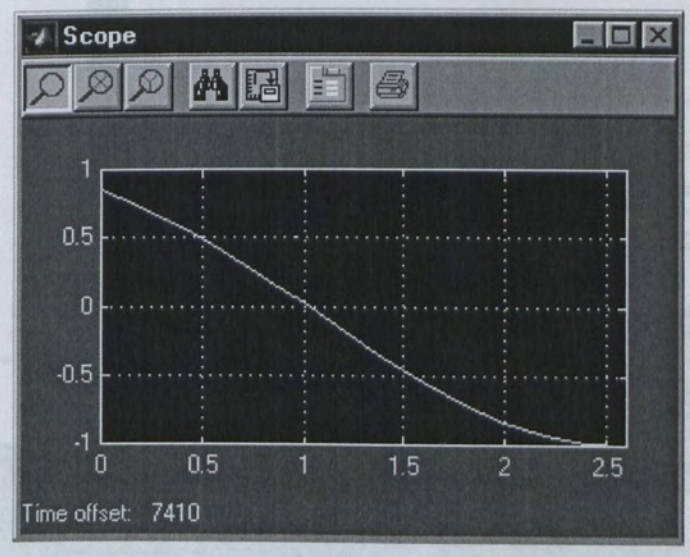


Figure 3.5 An example of Scope

The steps for creating a model and running a simulation are shown below:-

1. Type Simulink in the MATLAB command window.
2. Click the new model button to "On" the Library Browser's toolbar. Simulink will open a new model window.
3. To create a new model, copy block from the Simulink block libraries.
4. When all the necessary blocks copied into the model window, connect the blocks in the relationship you wish to have to represent the model.
5. Open the scope block to view the simulation output
6. Set up Simulink to run the simulation for certain duration.
7. Choose Start from the Simulation menu and watch the traces of the scope block's input.
8. The simulation stop when you choose STOP or it reaches the stop time specified in the Simulation Parameters.

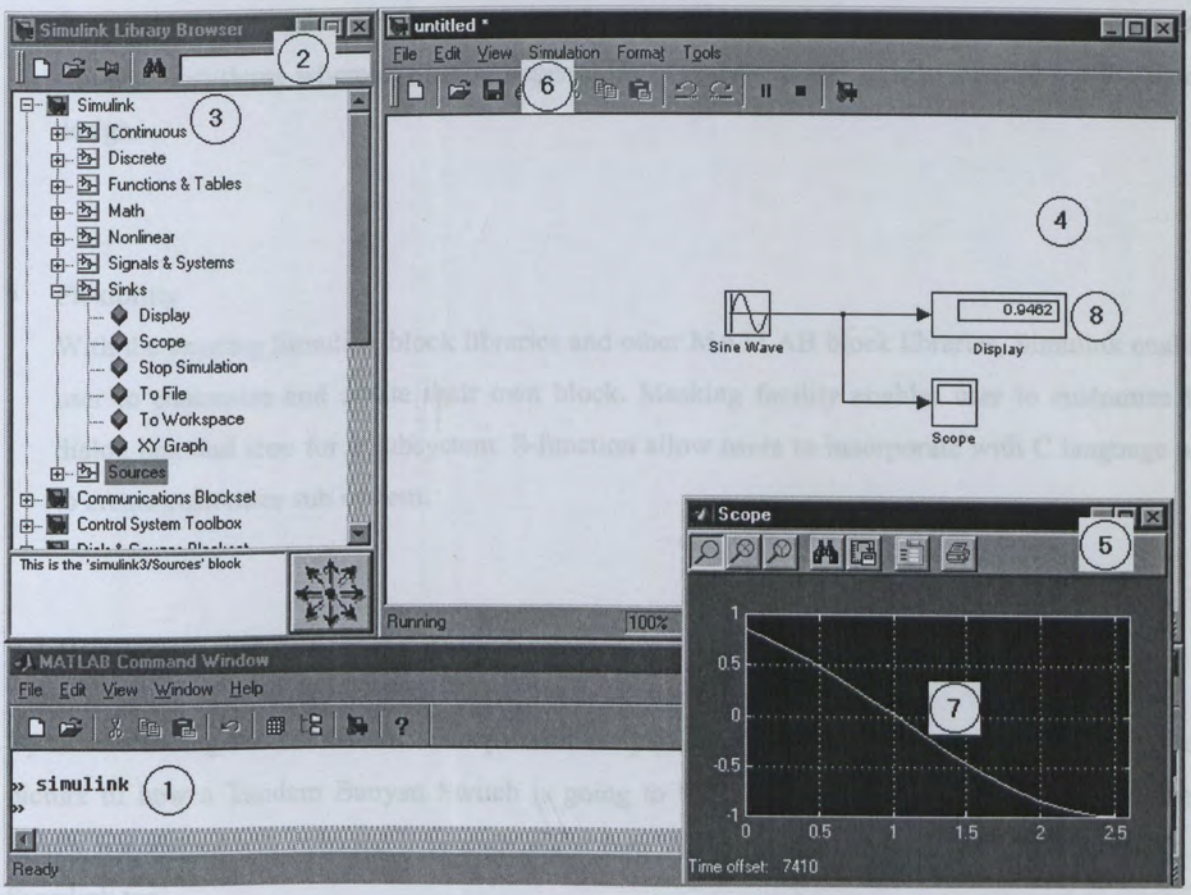


Figure 3.6 Layout of Simulink software on the computer screen

3.4 Advantage of using Simulink

- User Friendly Environment

  Simulink provides a user-friendly modelling environment through the graphical user interface (GUI) for building models as block diagram. By using click-and drag mouse operation, model can be drawn and refined efficiently.

- Dynamic modelling and Simulation

  Simulation in Simulink is interactive, parameter can be changed right away and the effect of the changes can be captured immediately even during the execution time of simulation. Blocks can be coloured based on their functionality to differentiate from others and to increase the readability of complex models.

- Modular design that provide reusability and expandability

  Subsystem blocks can be created using top to bottom approach to reduce the complexity of a model. Subsystems where execution depends on triggering signal aids in simplifying the model design.

- Flexibility

  With the existing Simulink block libraries and other MATLAB block libraries, Simulink enables user to customize and create their own block. Masking facility enables user to customize the dialog box and icon for a subsystem. S-function allow users to incorporate with C language and to create customize sub system.

3.5 Summary

By understanding the simulation concepts and the process of modelling a system, it helps create a picture of how a Tandem Banyan Switch is going to be simulated. Simulink Block that has been briefly introduced will help the user understand how ATM Tandem Banyan Switch is modelled in Simulink tool.

Chapter 4: Dynamic Modelling and Simulation

In Chapter 3, simulation and modelling concept have been introduced. Simulink tool has been introduced in general and it has been selected as a simulation tool to simulate ATM Tandem Banyan Switch. There are 3 steps in developing a simulation process, these include model design, model execution, and execution analysis. Chapter 4 will discuss the model design of the two-stage ATM Tandem Banyan Switch in details, which will then be implemented into Simulink Tool.

4.1 Modelling A Two-Stage ATM Tandem Banyan Switch

Models will be built using top-down approaches. The model of Two-State ATM Tandem Banyan Switch will be sub-divided into a smaller model according to their functionality and simplicity. The process of sub modelling can be down to a few layers until the model represents its unique functionality and simplicity. It helps to achieve the goal of building model that is reusable and expandable.

4.1.1 Two-State ATM Tandem Banyan Switch Model

A Two-State ATM Tandem Banyan Switch Model is shown below. It consists of 1st stage and 2nd stage Banyan Switch, and two Multiplexers for Outport 0 and Outport 1.

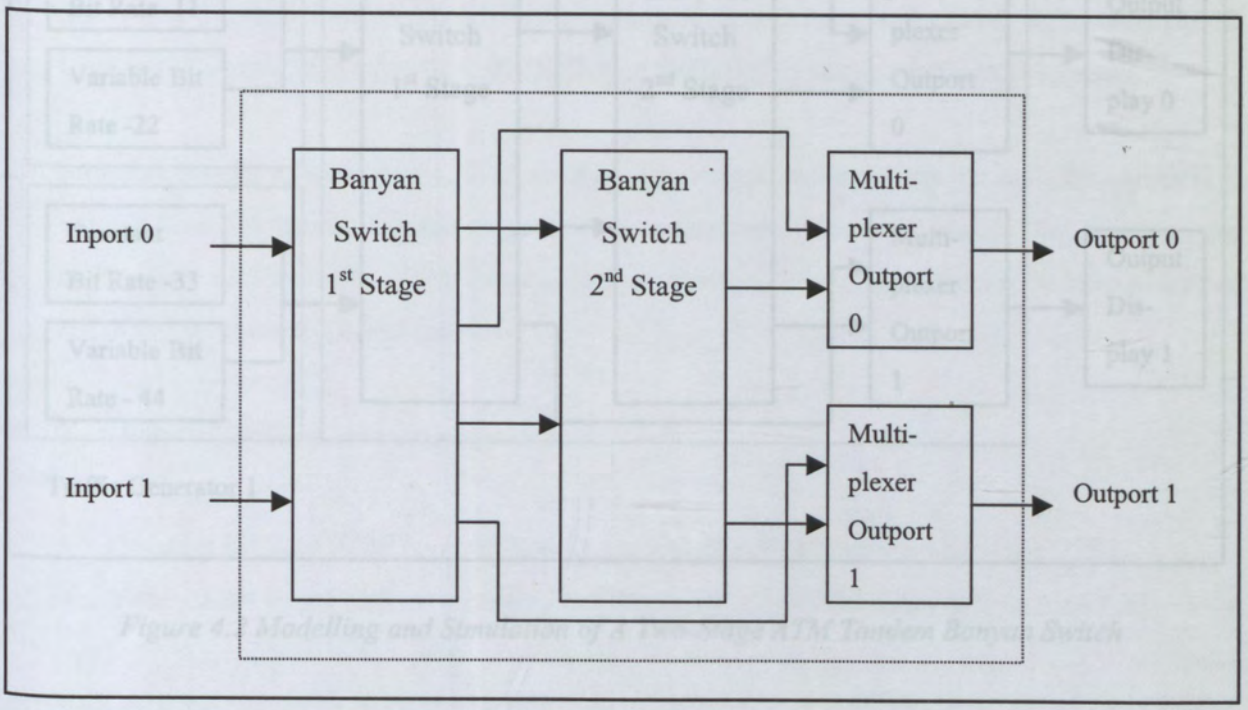


Figure 4.1: Two-Stage ATM Tandem Banyan Switch Model

4.1.2 Modelling and simulation of A Two-State ATM Tandem Banyan Switch Model

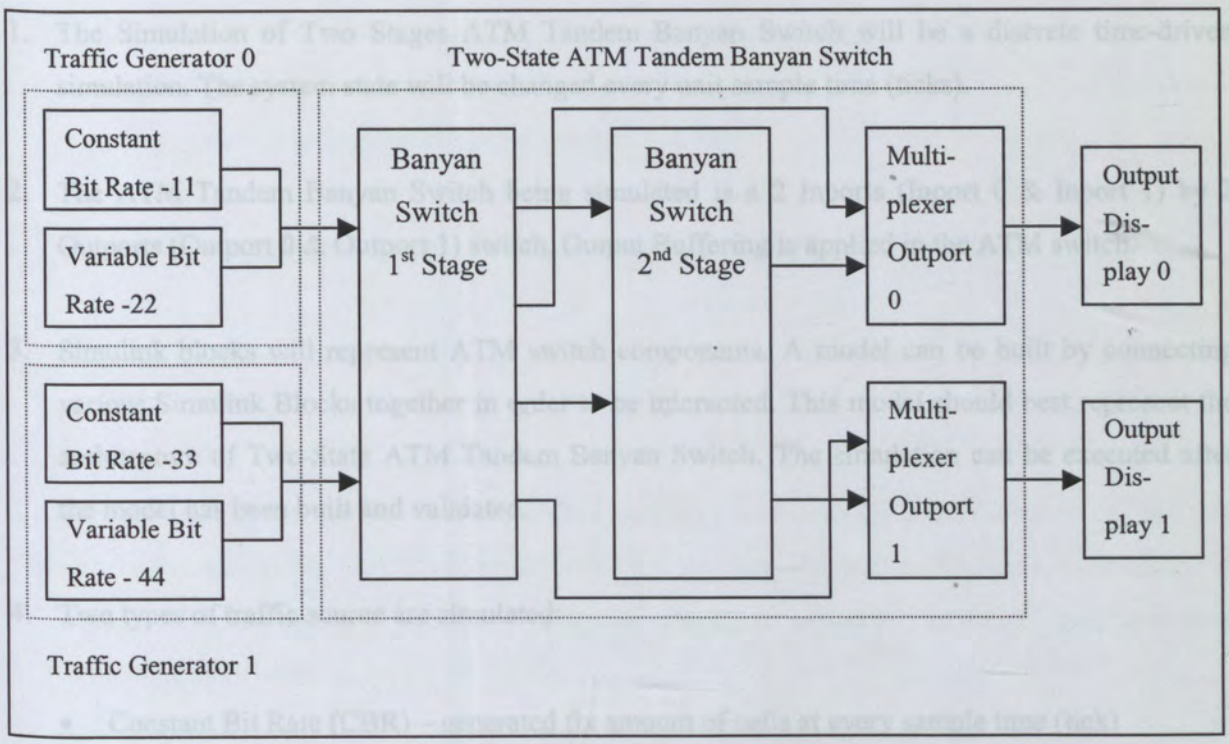


Figure 4.2 Modelling and Simulation of A Two-Stage ATM Tandem Banyan Switch

- Modelling and simulation of Two Stages ATM Tandem Banyan Switch require aid from extra two Traffic Generator Models and two Output Display Models. Two Stages ATM Tandem Banyan Switch Model itself is divided into four sub models, that is two Banyan Switch Models and two Multiplexer Output Models.

- Traffic Generator Model generates two types of traffic, that are Constant Bit Rate (CBR) cell and Variable Bit Rate (VBR) cell to the Inport of the Two Stages ATM Tandem Banyan Switch Model.

- Output Display Model shows the contents of the cell at the Outport of Two Stages ATM Tandem Banyan Switch. Contents of the cell are source, destination, traffic type and cell rate. Besides this, total cell received and total tick received are shown, in order to help analysing the performance of the Two Stages ATM Tandem Banyan Switch.

4.2 Characteristics of Two Stages ATM Tandem Banyan Switch Model

1. The Simulation of Two Stages ATM Tandem Banyan Switch will be a discrete time-driven simulation. The system state will be changed every unit sample time (ticks).

2. The ATM Tandem Banyan Switch being simulated is a 2 Inports (Inport 0 & Inport 1) by 2 Outports (Outport 0 & Outport 1) switch. Output Buffering is applied in the ATM switch.

3. Simulink blocks will represent ATM switch components. A model can be built by connecting various Simulink Blocks together in order to be interacted. This model should best represent the architecture of Two-State ATM Tandem Banyan Switch. The simulation can be executed after the model has been built and validated.

4. Two types of traffic source are simulated:

   - Constant Bit Rate (CBR) – generated fix amount of cells at every sample time (tick)
   - Variable Bit Rate (VBR) – generated not fix amount of cells at every sample time (tick)

5. Cell generated consists of 5 parameters, this parameters will help evaluate the performance of ATM Tandem Banyan Switch. Below show the 5 parameters:

- Number of Cells (In pulse form)
  - amplitude equal to number of cells generated, i.e. 50 cells
  - Change from 0 to 50 then back to 0 again for one cycle of sample time (tick.), i.e.1 second.
  - Pulse format is used because it help trigger the output buffer operation in Multiplexer Output Model later.

- Source – To differentiate varies source to the switch's inport.
  - 11 – Source CBR to Inport 0
  - 22 – Source VBR to Inport 0
  - 33 – Source CBR to Inport 1

- 44 – Source VBR to Inport 1

- Destination – To let switching algorithm know which outports the cell want to go
  - 0 – Switch to Outport 0 (Upper Port)
  - 1 – Switch to Outport 1 (Lower Port)
- Traffic Type – Represent Constant Bit Rate (CBR) and Variable Bit Rate (VBR)

- Number of Cells Generated – A static number (cell per tick) for display only while simulation is running.

6. The cell fields will be multiplexed before sending, and demultiplexed as soon as they are received at the Output Port. This is caused by the inability of Simulink to pass a combination of signals. Simulink will only pass numeric signal, which means that all the data (Signal) will have to be in numeric form.

The following section will describe in detail the function played by each model to simulate the Two Stages ATM Tandem Banyan Switch. Switching Algorithm and Mechanism will be included in the discussion too.

4.3 Implementation Issue

MATLAB Simulink tool is used in the dynamic modelling a Two State ATM Tandem Banyan Switch. Simulink tool is best used in modelling functional blocks. Therefore, functional block will best represent Two State ATM Tandem Banyan Switch. To build a reusable and expandable model, the functional block must be able to break into more simple functional block, which at the end, is able to represent the simplest function of the block.

4.3.1 How models are represented in Simulink tool.

Basically, models represent a system that consists of three basic functional blocks, they are input block, processing block and output block. Input block generates input signal to the system, processing block is based on its own logical characteristic to convert input signal to output signal, and the output block display the output for user understanding.

4.3.2 How models are built in Simulink

Model can be built from a few Simulinks block that interconnects each other. Users can choose Simulink block that best represent the function of a model, normally a model consist of source block that provides signal into the model, the processing block that process the input signal and sink block that display output from the model.

Model can consist of sub models that interconnect with each other or Simulink block. Normally, sub model is build to represent the model in a simple way. Sub model that has the same functionality can be reused to reduce the complexity of a model.

4.3.3 How models are connected among each other

Model or Simulink blocks itself consists of Inport or Outport. A line is connected between one Simulink block's Inport and other Simulink block's Outports, so that data can flow in between the two entities. Data flows must have the same data property at both ends, so that there is no software syntax error.

There are two data types used in the simulation. This includes pulse data type and numeric data type. Pulse data type generated by a pulse generator mainly represent a cell packet while numeric data type is used to represent header or data information.

4.3.4 How simulation works

Before simulation works, all Simulink block's data property must be set correctly, this includes data that flow from Source block, passing through processing block, and lastly end at Sink block. Besides that, the sample rate for each Simulink block and simulation start / end time need to be configured correctly.

There are three sample rates used in the simulation of ATM Tandem Banyan Switch. They are listed as below:

1) Sample rate at 10 ms per tick, this 10 ms sample rate used at Traffic Generator to general pulse data type for cell transmission.
2) Sample rate at 5 ms per ticks, these 5 ms sample rate used at the Multiplexer Output Model, to multiplex the cell gathered from 1st Stage ATM Tandem Banyan Switch and 2nd Stage ATM Tandem Banyan Switch.
3) Sample rate at 2.5 ms per tick, this 2.5 ms sample rate used at Multiplexer Output Model, to convert the output from item (2) into pulse data type again.

4.3.5 Important Simulink blocks that make this simulation project works

Below are shown some of the important Simulink blocks that make this simulation project work.

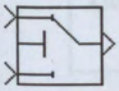
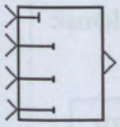
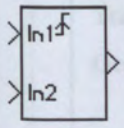
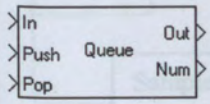
Simulink Block	Block Name	Description
	Pulse Generator	Pulse generator used to generate pulse data type according to the property set by user, i.e: sample rate & etc.
	Mux and Demux	Mux used to multiplex a few numeric data inputs into pulse data output. Demux used to demultiplex pulse data input back to a few numeric data outputs.
	Switch	Switch control by the middle control signal, it either received signal from upper port or lower port.
	Multiport Switch	Multiport Switch control by uppermost control signal, which will instruct the switch to receive signal from one of the available port.
	Sub System	Sub System represents a sub model. A sub model can consist of another sub-sub model. Some Sub Systems requires trigger signal in order to be executed.
	Queue	Queue requires Push and Pop signal in order to control the data flow in FIFO basic. Data can be stored inside the Queue as long as it does not exceed the storage limit.

Table 4.1 Mostly used Simulink Block

4.4 Traffic Generator Model

The functional block of a Traffic Generator Model is shown below. Each function is represented by a sub model. There are 5 major sub model that form the Traffic Generator Model, they are Traffic Generator, Sample Generator, Buffer Queue, Link Size, Control Algorithm and Control Mechanism. One Traffic Generator Model is dedicated for one inport.

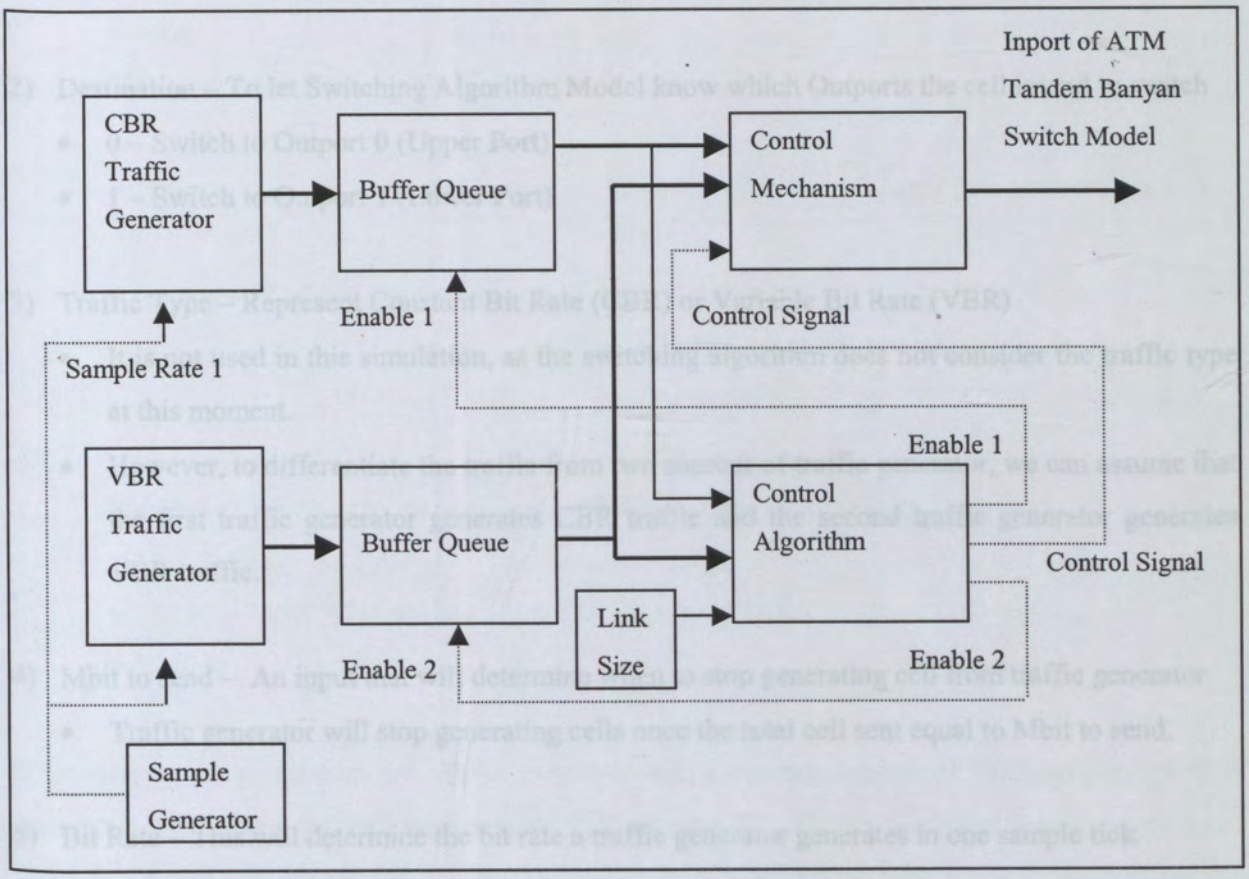


Figure 4.3 Traffic Generator Model

4.4.1 Traffic Generator Model

Traffic Generator Model provides two types of traffic, that are Constant Bit Rate (CBR) cell and Variable Bit Rate (VBR) cell to the Inport of Two Stages ATM Tandem Banyan Switch Model.

Traffic Generator Model require some pre-set value in order to generate the required bit rate cell for simulation, the required pre-set value are listed as below:

1) Source - to differentiate various sources to the switch's Inport.
   - 11 – From Traffic Generator Model 0 to Inport 0 (CBR)
   - 22 – From Traffic Generator Model 0 to Inport 0 (VBR)
   - 33 – From Traffic Generator Model 1 to Inport 1(CBR)
   - 44 – From Traffic Generator Model 1 to Inport 1 (VBR)

2) Destination – To let Switching Algorithm Model know which Outports the cell intend to switch
   - 0 – Switch to Outport 0 (Upper Port)
   - 1 – Switch to Outport 1 (Lower Port)

3) Traffic Type – Represent Constant Bit Rate (CBR) or Variable Bit Rate (VBR)
   - It is not used in this simulation, as the switching algorithm does not consider the traffic type at this moment.
   - However, to differentiate the traffic from two sources of traffic generator, we can assume that the first traffic generator generates CBR traffic and the second traffic generator generates VBR traffic.

4) Mbit to send – An input that will determine when to stop generating cell from traffic generator
   - Traffic generator will stop generating cells once the total cell sent equal to Mbit to send.

5) Bit Rate – This will determine the bit rate a traffic generator generates in one sample tick.

Traffic Generator based on the preset input value, generate cells that have the following header and data format:

Number of Cells (In pulse form) | Source | Destination | Traffic Type | Number of Cells Generated.

- Number of Cells (In pulse form)
  - Pulse amplitude represents number of cells generated in one sample time (tick.), i.e. 50 cells in one tick.
  - Pulse format help 'push in' and 'pop out' operation in buffer queue.

- Number of Cells Generated
  - A static number for display only, while simulation is running.

- Below is shown an example on how the number of cell is generated (Assume sample rate is 1 second = 1 tick).
  - Given preset value as below:

    Bit Rate = 19.2 Kbit per tick

    Mega Bit to send = 0.384 Mbit or 384 Kbit

  - As known, one cell equal to 5 byte header and 48 byte data or 384 bit data (48 byte * 8 Bit)

    Thus, number of cell generated per tick = Bit Rate / bit

    = 19200 bit per tick / 384 bit

    = 50 cells per tick

    Number of tick in order to complete sending all Mega bit data cells

    = Mbit to send / Bit Rate

    = 384 Kbit / 19.2 Kbit per tick

    = 20 ticks

By looking at the above example, traffic generator with a constant bit rate of 19.2Kbit per tick, will generate a pulse data type with amplitude of 50 (represent 50 cells per tick) and it will require total of 20 pulses data type (represent 20 ticks) to completely sending all 384Kbit cell into buffer queue.

4.4.2 Sample Generator Model

Sample rate need to be set before Sample Generator starts to generate pulse to the traffic generator. Sample rate will affect the cycle time and will affect the frequency of cells generated by the Traffic Generator. The faster the sample rate, the faster the Traffic Generator, generates cells into the buffer queue. There is three sample rate used in this simulation, it is stated in Section 4.3.4.

4.4.3 Buffer Queue Model

As there is only one inport available for cell generated by two different Traffic Generators, cells have to be queued at their respective Buffer Queue before it is their turn to pass through. Buffer Queue have a preset buffer size which will store cells that are pushed into the buffer. If the number of cell stored in the buffer exceeds the buffer storage limit, it will drop the cell. In order to pop out cell in First In First Out basic, buffer queue have to receive enable siganl from the Control Algorithm Model.

At every sample tick, only pulse data cells from one traffic generator are allowed to pass through the Buffer Queue model. Total number of cells that is pushed into the Buffer Queue must be equal to total number of cells that pops out from the Buffer Queue. This is to make sure all the cells are sent out from the Traffic Generator.

4.4.4 Link Size Model

Link size represent the broadband of a connection medium, for example a fiber optic cable that connect, in between a computer and a ATM Switch in a LAN network. Link size has higher transmission capability to transmit cells at higher bit rate, thus it allows different traffic type of data cell to pass through the link.

4.4.5 Control Algorithm Model

Control Algorithm Model generates control signal to Buffer Queue Model and Control Mechanism Model. The control signal based on the different traffic bit rate ratio, to allocate certain sample time (ticks) for certain traffic cell to pass through the link. This means only one type of traffic cells will be allowed to pass through the link at a time and different traffic cells have to take turn to pass through the link until there is no more cell generated from the traffic source.

For example, given the link size is 96 Kbit per tick, and the link is connected by two traffic generators, for example CBR (19.2 Kbit per tick) and VBR (9.6 Kbit per tick). The rest of the link size (67.2 Kbit per tick) will be let idle. The ratio of bit rate for different traffic type will be CBR (2), VBR (1) and other (7). This ratio will be output to the Control Mechanism Model, the CBR cell will

be allocate 2 ticks time to pass throught the link, VBR cell will be allocated 1 tick time to pass through the link and the other 7 ticks time will let the link be idle.

4.4.5 Control Mechanism Model

Control Mechanism Model acts as a switch that interconects between link and the traffic generator. It is based on the control signal given by the Control Algorithm Model, allocates tick time for different traffic type to pass through it. For example, based on the previous example. At first two ticks, Control Mechanism Switch will switch to CBR traffic generator, in the third tick, Control Mechanism Switch will be switched to VBR Traffic Generator and the rest of the seven ticks will be let idle. This cycle will be repeated untill there is no cell generated from the traffic generator.

4.5 Output Display Model

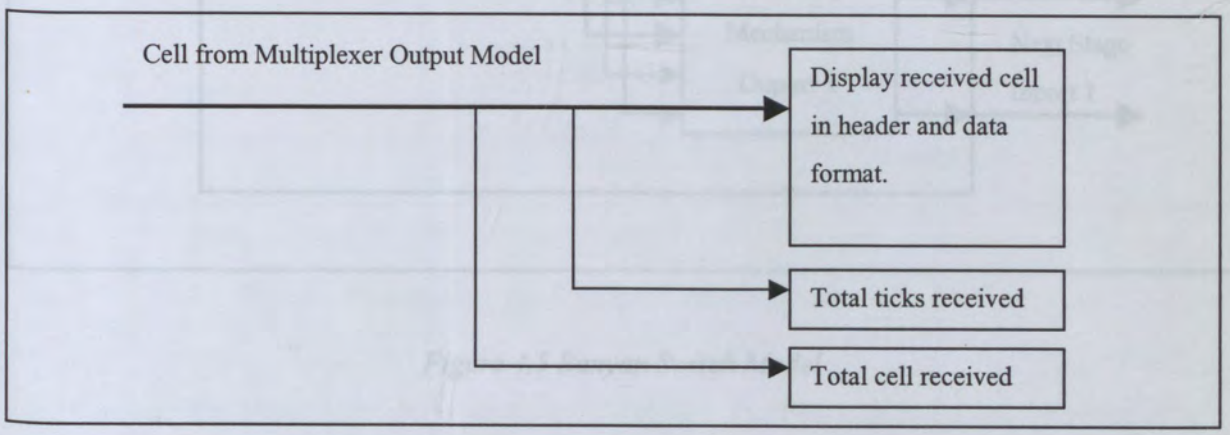


Figure 4.4 Output Display Model

Output Display Model displays the cell output from Multiplexer Output Model. It show the cell in header and data format. This enables cells to be different by looking at the source, type of bit rate and number of cell received.

Beside this, Output Display Model also show total ticks received and total cell received. This two parameter will help determine whether the ATM Switch experience any cell loss or not and whether the output is active or idle. The output also can be linked to other Simulink analysis tool for further output analyse.

4.6 Banyan Switch Model

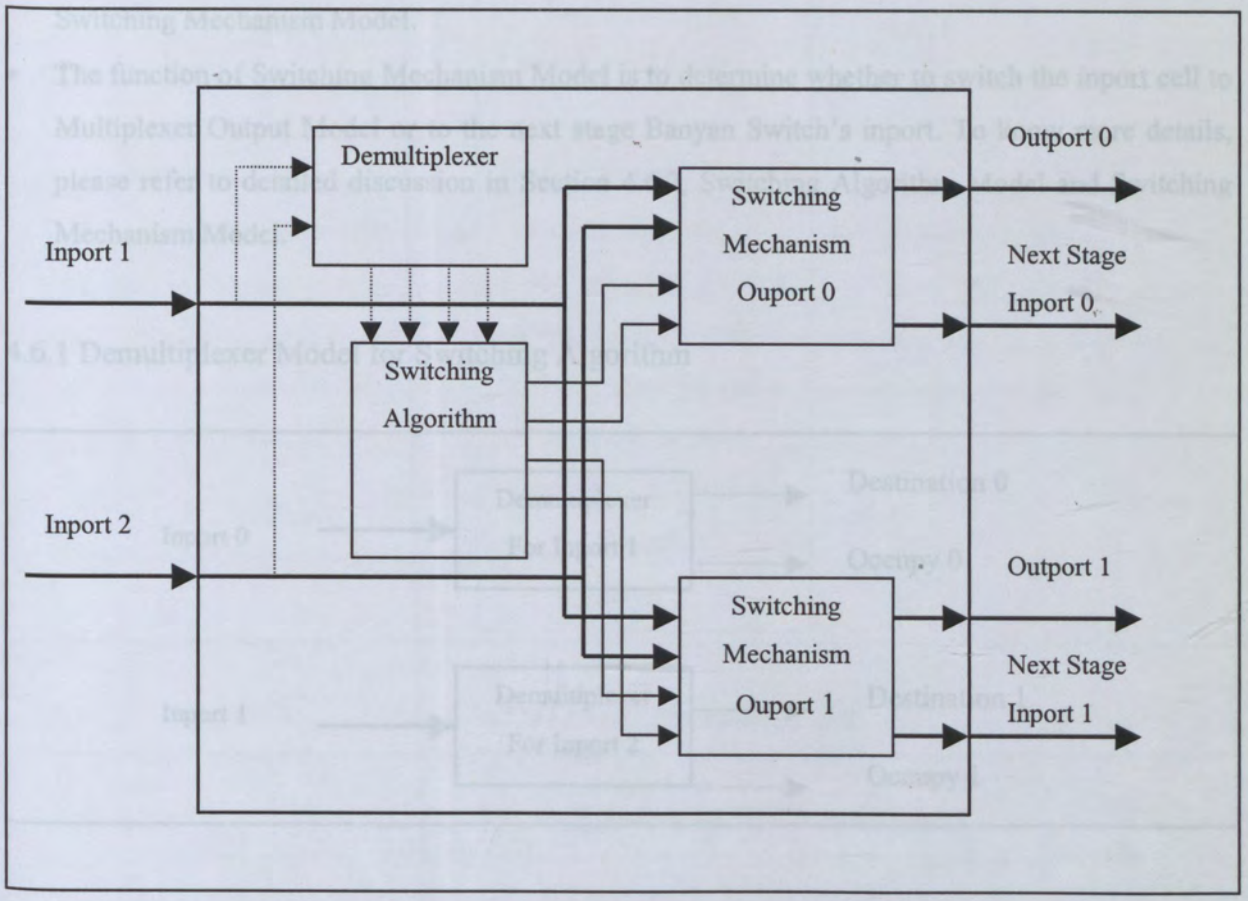


Figure 4.5 Banyan Switch Model

- The 2 Inports x 2 Outports Banyan Switch Model is shown at Figure 4.5. By connecting two Banyan Switch Models in series, together with two Multiplexer Output Models, a Two-State ATM Tandem Banyan Switch Model is formed.
- Banyan Switch Model consists of four sub models, that is Demultiplexer Model, Switching Al
- gorithm Model and two Switching Mechanism Models for two outports.
- The function of Demultiplexer Model is to extract the destination (*dest*) bit and the number of cell generated (*occupy*) bit from the cell in order for Switching Algorithm Model to run the logic expression.

- The function of Switching Algorithm Model is to examine the inport cells' destination bit and to detect any blocking condition before passing the switching signal either (*Switch* or *Mark*) to the Switching Mechanism Model.
- The function of Switching Mechanism Model is to determine whether to switch the inport cell to Multiplexer Output Model or to the next stage Banyan Switch's inport. To know more details, please refer to detailed discussion in Section 4.6.2, Switching Algorithm Model and Switching Mechanism Model.

4.6.1 Demultiplexer Model for Switching Algorithm

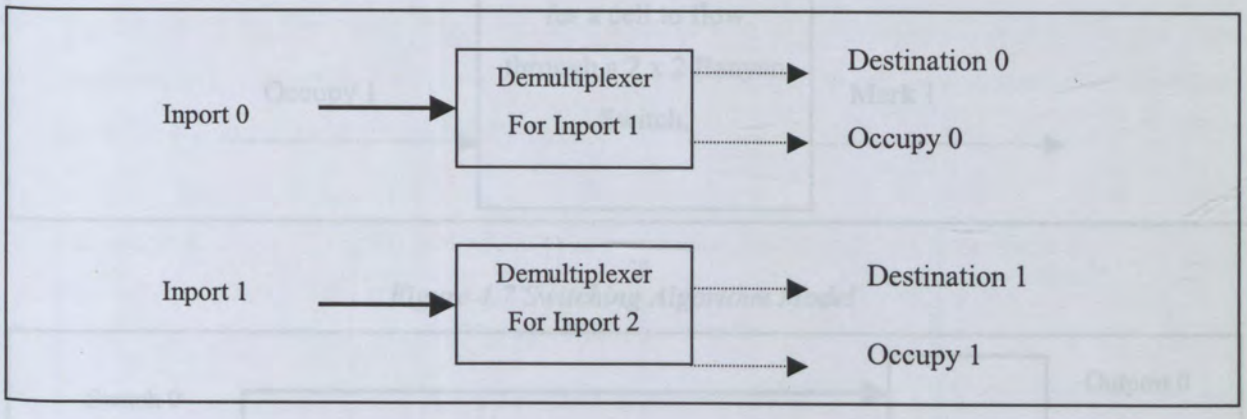


Figure 4.6 Demultiplexer Model for Switching Algorithm Model

- Demultiplexer Model extracts the destination data (*dest*) and the number of cell generated data (*occupy*) from the cell in order for Switching Algorithm Model to run logic expression.
- Decimal to Binary Converter inside the Demultiplexer Model converts the destination data from decimal value to binary value.
- Destination (*dest*) data will be either 0 or 1. 0 means the cell intends to switch to Outport O (upper port), 1 means the cell intend to switch to Outport 1 (lower port).
- Data present (*occupy*) represent whether there is any cell generated at a particular time. If there is cell generated, *occupy* is 1. If there is no cell generated, *occupy* is 0.
- This *occupy* bit will let switching algorithm ignore *dest* bit if there is no cell generated (*occupy* is 0).

4.6.2 Switching Algorithm Model and Switching Mechanism Model

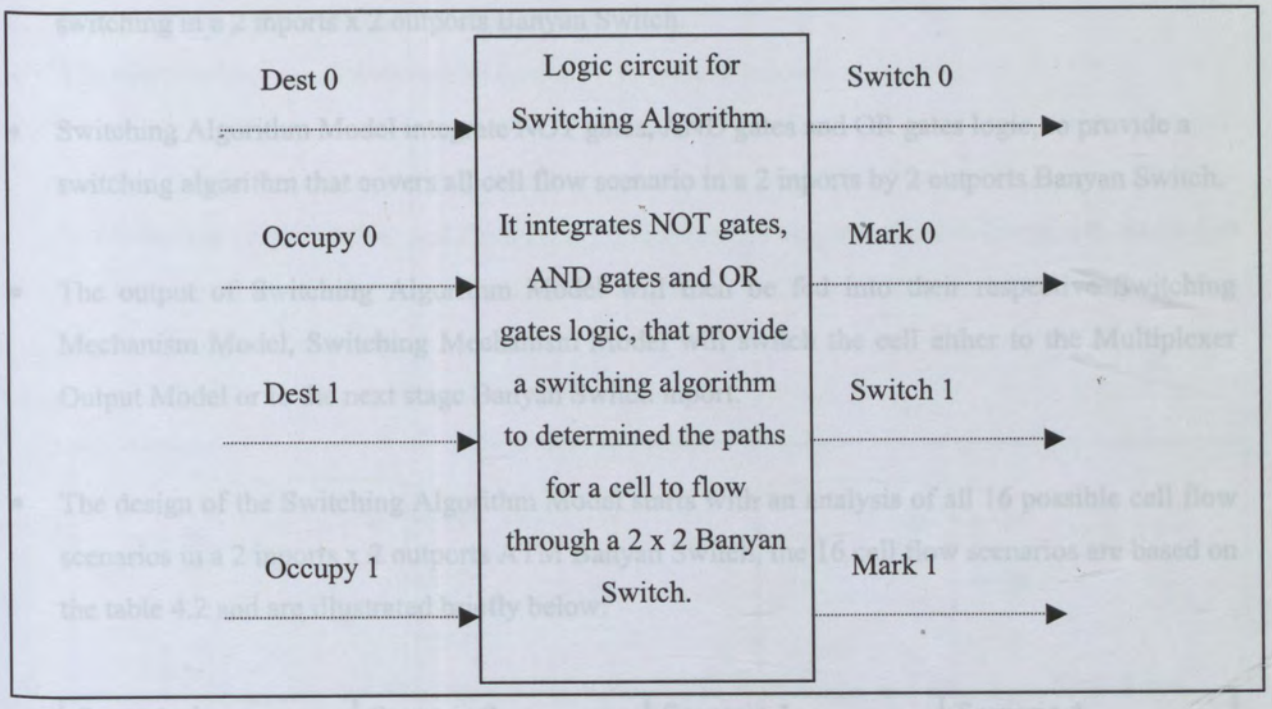


Figure 4.7 Switching Algorithm Model

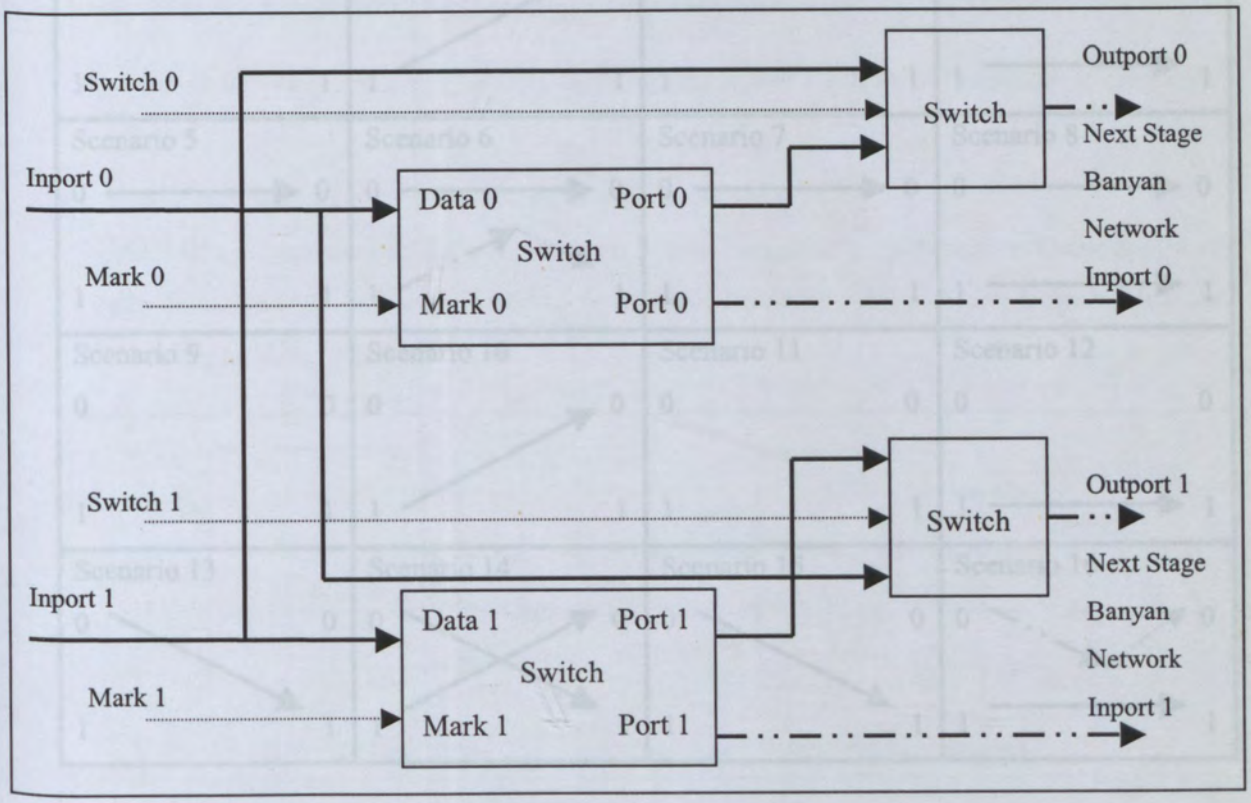


Figure 4.8 Switching Mechanism Model

- Switching Algorithm Model and Switching Mechanism Model work together to perform switching in a 2 inports x 2 outports Banyan Switch.
- Switching Algorithm Model integrate NOT gates, AND gates and OR gates logic, to provide a switching algorithm that covers all cell flow scenario in a 2 inports by 2 outports Banyan Switch.
- The output of Switching Algorithm Model will then be fed into their respective Switching Mechanism Model, Switching Mechanism Model will switch the cell either to the Multiplexer Output Model or to the next stage Banyan Switch inport.
- The design of the Switching Algorithm Model starts with an analysis of all 16 possible cell flow scenarios in a 2 inports x 2 outports ATM Banyan Switch, the 16 cell flow scenarios are based on the table 4.2 and are illustrated briefly below:

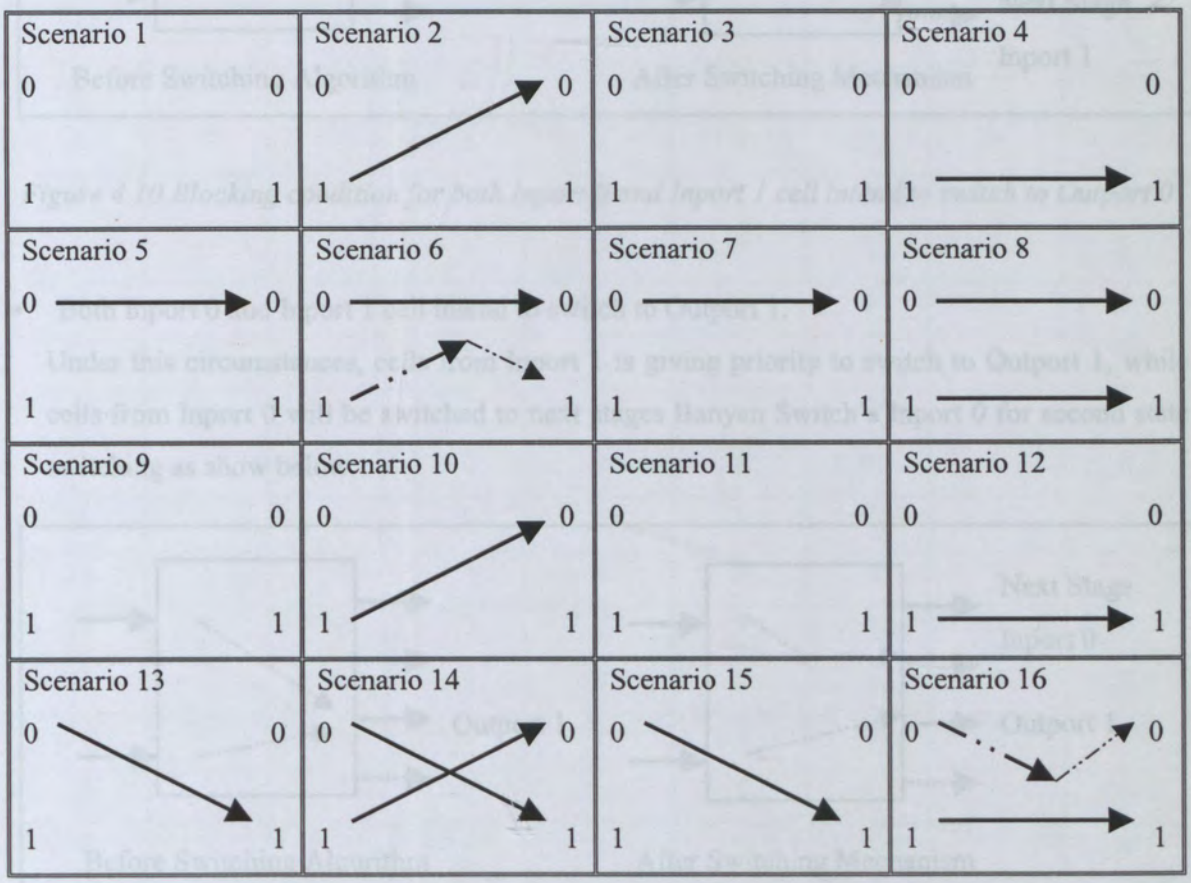


Figure 4.9 All 16 possible cell flow scenarios in a 2 Inports x 2 Outports ATM Banyan Switch

- Based on the above illustration, there are only two possibilities for blocking condition to be happen, it happens at Scenario 6 and Scenario 16.
- The two blocking conditions can be further described as below:

  - Both Inport 0 and Inport 1 cell intend to switch to Outport 0.

    Under this circumstance, cell from Inport 0 is given priority to switch to Outport 0, while cell from Inport 1 will be switched to next stage Banyan Switch's Inport 1 for second state switching as show below.

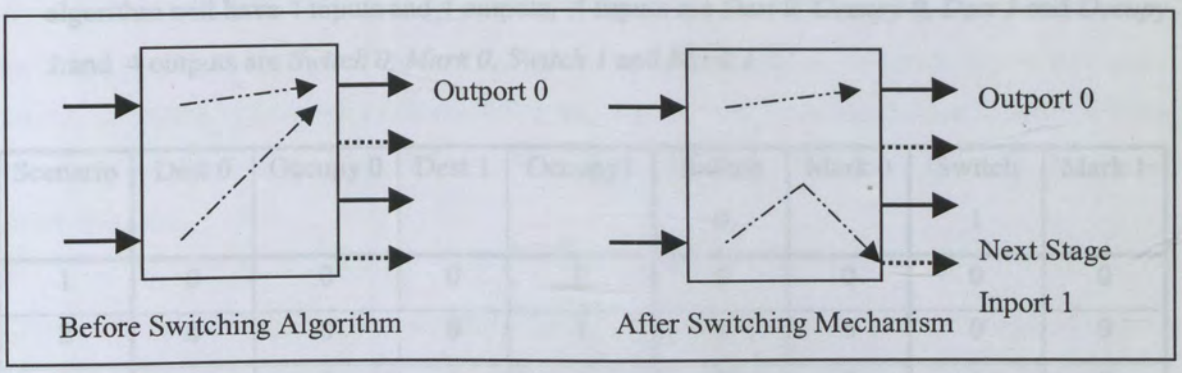


Figure 4.10 Blocking condition for both Inport 0 and Inport 1 cell intend to switch to Outport 0

  - Both Inport 0 and Inport 1 cell intend to switch to Outport 1.

    Under this circumstances, cells from Inport 1 is giving priority to switch to Outport 1, while cells from Inport 0 will be switched to next stages Banyan Switch's Inport 0 for second state switching as show below.

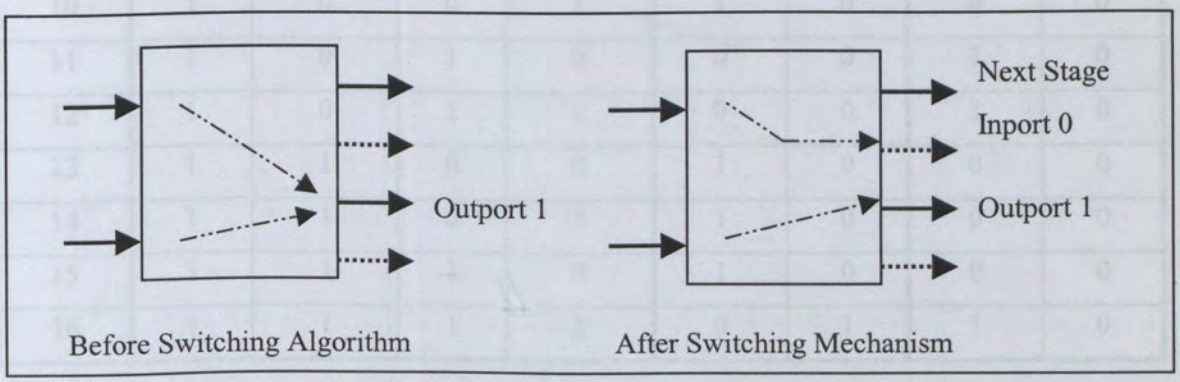


Figure 4.11 Blocking condition for both Inport 0 and Inport 1 cell intend to switch to Outport 1

- Switching Algorithm Model consist of a logic integrated circuit that provides switching algorithm for all 16 cell flow scenarios in a 2 inports x 2 outports ATM Banyan Switch.

- The design of the switching algorithm must first look into 4 parameters that create 16 cell flow scenarios in a 2 inports x 2 outports ATM Banyan Switch. Later every consequence of these 16 scenarios has to be listed out. A logic expression table can well explain the switching algorithm that covers all these 16 cell flow scenarios.

- The logic expression table for Switching Algorithm Model is shown below. The switching algorithm will have 4 inputs and 4 outputs, 4 inputs are *Dest 0, Occupy 0, Dest 1* and *Occupy 1*,and 4 outputs are *Switch 0, Mark 0, Switch 1* and *Mark 1*

Scenario	Dest 0	Occupy 0	Dest 1	Occupy1	Switch 0	Mark 0	Switch 1	Mark 1
1	0	0	0	0	0	0	0	0
2	0	0	0	1	1	0	0	0
3	0	0	1	0	0	0	1	0
4	0	0	1	1	0	0	1	0
5	0	1	0	0	0	0	1	0
6	0	1	0	1	0	0	1	1
7	0	1	1	0	0	0	1	0
8	0	1	1	1	0	0	1	0
9	1	0	0	0	1	0	0	0
10	1	0	0	1	1	0	0	0
11	1	0	1	0	0	0	1	0
12	1	0	1	1	0	0	1	0
13	1	1	0	0	1	0	0	0
14	1	1	0	1	1	0	0	0
15	1	1	1	0	1	0	0	0
16	1	1	1	1	0	1	1	0

Table 4.2 Logic Expression for Switching Algorithm Model that covers all 16 cell flow scenarios

- Referring to the above logic expression table, the blocking condition that was mentioned before, happens at scenario 6 and 16.

- At scenario 6, both Inport 0 and Inport 1 cell (*Occupy* 0 =1 and *Occupy* 1 = 1) intend to switch to Outport 0 (*Dest* 0 = 0 and *Dest* 1 = 0). Under this circumstance, cell from Inport 0 is given priority to switch to Outport 0 (*Switch* 0 = 0 and *Mark* 0 = 0), while cell from Inport 1 will be switched to the next stage Banyan Switch's Inport 1 (*Switch* 1 = 1 and *Mark* 1 = 1), for second states switching.

- At scenario 16, both Inport 0 and Inport 1 cell (*Occupy* 0 =1 and *Occupy* 1 = 1) intend to switch to Outport 1 (*Dest* 0 =1 and *Dest* 1 = 1). Under this circumstance, cell from Inport 1 is given priority to switch to Outport 1 (*Switch* 1 = 1 and *Mark* 1 = 0), while cell from Inport 0 will be switched to the next stage Banyan Switch's Inport 0 (*Switch* 0 = 0 and *Mark* 0 = 1), for second states switching.

- Based on the above logic expression table, the 4 outputs can be derived in Boolean logic format. Given A =*Dest,* B = *Occupy,* C = *Dest,* and D = *Occupy*. The Boolean logic format for 4 outputs will be as below:

$$\text{Switch 0} = \bar{A}\bar{B}\bar{C}D + A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D + AB\bar{C}\bar{D} + AB\bar{C}D + ABC\bar{D}$$
$$= \bar{A}\bar{B}\bar{C}D + A\bar{B}\bar{C} + AB(\bar{C}\bar{D} + \bar{C}D + C\bar{D})$$

$$\text{Mark 0} = ABCD$$

$$\text{Switch 1} = \bar{A}\bar{B}C + \bar{A}B\bar{C} + ABC + A\bar{B}C + ABCD$$
$$= \bar{A}\bar{B}C + \bar{A}B + A\bar{B}C + ABCD$$

$$\text{Mark 1} = \bar{A}B\bar{C}D$$

- The 4 outputs (bit 0 or 1) will be fed into the respective Switching Mechanism Models to control the cell flow switching from respective Inport to either Multiplexer Output Model or next stage Banyan Switch Inport.

- If the output for Switch 0 or Switch 1 is 0, then the respective Switching Mechanism Models intend to let the cell from Inport 0 pass through the model. Else if the output for Switch 0 or Switch1 is 1, then the respective Switching Mechanism Models intend to let the cell from Inport 1 pass through the model.

- If the output for Mark 0 or Mark 1 is 0, then the cell flow from the respective Inport will not be direct to next stage Banyan Switch inport. Else if the output for Mark 0 or Mark 1 is 1, then the cell flow from the respective Inport will be directed to next stage Banyan Switch respective inport.

4.6.3 Multiplexer Output Model

Below, is shown the signal and cell flow in the Multiplexer Output Model

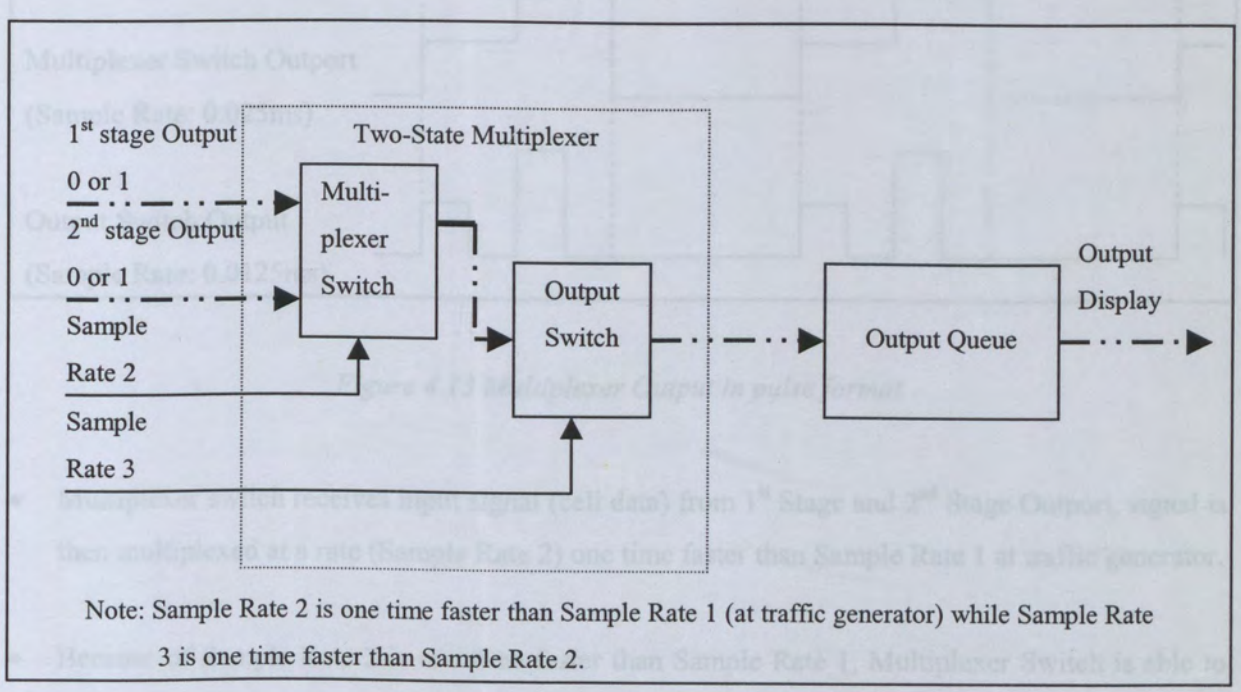


Figure 4.12 Multiplexer Output Model

- Sample Rate 2 is one times faster than Sample Rate 1 at traffic generator, for example, if Sample Rate 1 is 0.050ms, then the Sample Rate 2 will be 0.025ms, It is used to capture cell data from 1st stage ATM Banyan Switch and 2nd stage ATM Banyan Switch.

- Sample Rate 3 is one times faster than Sample Rate 2. If Sample Rate 2 is 0.025ms, then the Sample Rate 3 will be 0.0125ms, It is used to create output in pulse form again (Likes the pulse form generated from Traffic Generator). Below shown the output signal in pulse format. Amplitude will represent number of cells transmitted per tick.

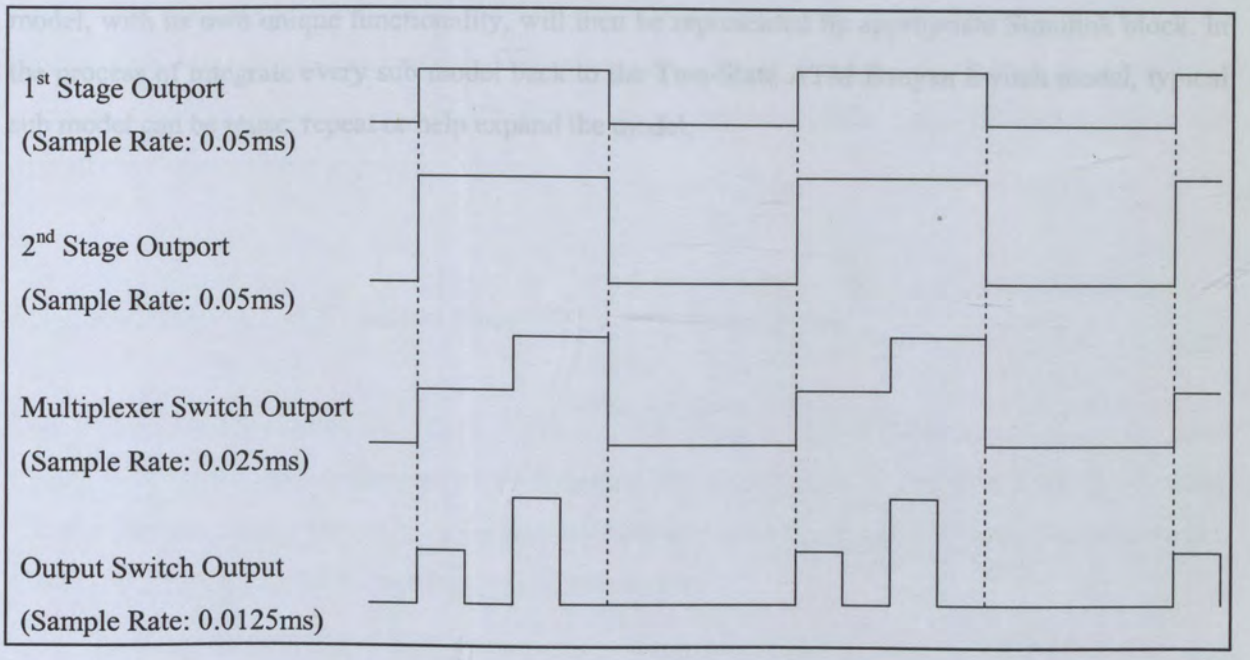


Figure 4.13 Multiplexer Output in pulse format

- Multiplexer switch receives input signal (cell data) from 1st Stage and 2nd Stage Outport, signal is then multiplexed at a rate (Sample Rate 2) one time faster than Sample Rate 1 at traffic generator.

- Because of Sample Rate 2 is one time faster than Sample Rate 1, Multiplexer Switch is able to multiplex both signal from1st Stage Outport and 2nd Stage Outport without any cell loss.

- Output Switch later acts as traffic generator again to generate cell data in pulse format from Multiplexer Switch to Outport Switch. The Sample Rate 3 used at Output Switch is one time faster than Sample Rate 2 at Multiplexer Switch.

4.7 Summary

The modelling of Two-State ATM Banyan Switch uses top-down approach that enable model designer to sub-divide the switch into smaller model according to its unique functionality. Each sub model, with its own unique functionality, will then be represented by appropriate Simulink block. In the process of integrate every sub model back to the Two-State ATM Banyan Switch model, typical sub model can be reuse, repeat or help expand the model.

Chapter 5 Discussions and Conclusions

This chapter discusses the simulation of the two stages ATM Tandem Banyan Switch. Functionality and the performance of the switch have been the evaluated after the simulation. Explanation is given on how the architecture design of the switch has influence on the simulation result. During the simulation, different cell flow scenarios are tested in the simulation and various cell flow patterns are collected. Discussion is mainly concentrated on two main cell flow patterns that is cell flows with blocking and without blocking condition. Discussion is also on how the design of the ATM Tandem Banyan Switch's architecture will help to reduce the cell loss rate when there is cell flow with blocking condition. Here, some assumptions are made in the simulation. Advantages and limitations of using Simulink in modelling and simulation are explained. Finally, areas of enhancements and overall conclusion for the project are made.

5.1 Two Stage ATM Tandem Banyan Switch Simulation

The 2 Inports x 2 Outports, Two-State ATM Tandem Banyan Switch being simulated, is the most simple multi-stages Tandem Banyan network design. The switch actually enable cell that losses at the 1st stage Banyan Switch Model have the second chance to pass through the 2nd stage Banyan Switch Model, therefore eliminate the cell loss rate at the switch.

The design of this Banyan Switch architecture is so unique that it enable Banyan Switch Model to be repeatable and expandable. For example, if a 4 Inport by 4 Outport, ATM Banyan Switch needs to be simulated, the 2 Inports by 2 Outport, Banyan Switch Model can be duplicate and expand to become a 4 Inports by 4 Outports, Banyan Switch Model. Figure 2.5 in Chapter 2 best illustrated the 4 Inports by 4 Outports Banyan Switch Model.

The success criteria of this simulation are based on the functionality and performance of the switch. The functionality of the switch will examine whether the switch can function as per design, result shown that all cells are routes to its Outport correctly according to the destination bit carried in the cell. Another success criterion is to measure the performance of the switch, various tests have been carrying out to examine the switch performance. Simulation results shown that the switch passes all 16 cell flow scenarios that available in the 2 Inports x 2 Outports Banyan Switch. In the next section,

test and simulation result will be explained in more details, in order to understand the functionality and performance of the switch.

Refer to Figure 4.9. A 2 Inports by 2 Outports ATM Banyan Switch consists of 16 cell flow scenarios. Out of these 16 cell flow scenarios, 4 typical cell flow patterns can be determined and it is shown as below:

Cell Flow Pattern	**Scenario #**
No cell flow between Inport and Outport of the switch	1,3,9 & 11
Cell flow from one inport to one outport	2,4,5,7,10,12,13 & 15
Cell flow from two inports to two different outports	8 & 14
Cell flow from two inports to the same outports	6 &16

Table 5.1 Cell Flow Pattern for a 2 Inports x 2 Outports ATM Banyan Switch

5.1.1 No cell flow between Inport and Outport of the switch

- This is applied for cell flow scenario 1, 3, 9 and 11 (Refer Figure 4.9)
- There is no cell generated from both the Traffic Generators, hence, no cell flow between Inport and Outport of the switch.
- No Switching performed by the ATM Tandem Banyan Switch.
- Total cell received at both the Outport 0 and Outport 1 is zero.

5.1.2 Cell flow from one inport to one outport

- This is applied for cell flow scenario 2, 4, 5, 7, 10, 12, 13 and 15 (Refer Figure 4.9)
- There is cells generated from one of the Traffic Generator and its flow to one of the Switch's Outport.
- The 1st stage Banyan Switch performs switching to the cell according to its destination. (No blocking condition) at Switching Speed of 0.05ms (Sample Rate 1).
- The 2nd stage Banyan Switch does not need to perform switching, as there is no cell available.

- As sample rate at Multiplexer Switch (0.025ms) is greater than sample rate at Banyan Switch (0.05ms), Multiplexer Switch is able to switch all cell from both the Banyan Switch Model to its Output Switch. Total cells received at the Outport equals to total cell generated from respective Traffic Generator, there is no cell loss in between the flow.

5.1.3 Cell flow from two inports to two different outports

- This is applied for cell flow scenario 8 and 14 (Refer Figure 4.9).
- There are cells generated from both the Traffic Generators and it flows to different Outport, hence, no blocking condition happens.
- The 1st stage Banyan Switch performs switching to the cell according to its destination. (No blocking condition happen)
- The 2nd stage Banyan Switch does not need to perform switching, as no cell available.
- As sample rate at Multiplex Switch (0.025ms) is greater than sample rate at Banyan Switch (0.05ms), Multiplex Switch is able to switch all cell received from Banyan Switch Model to its Output Switch. Total cells received at the Outport equals to total cell generated from both Traffic Generators, there is no cell loss in between the flow.

5.1.4 Cell flow from two Inports to the same outports

- This applied for cell flow scenario 6 and 16 (Refer Figure 4.9).
- There is cell generated from both the Traffic Generators and its flow to the same Outport, hence, blocking condition occurs.
- The 1st stage Banyan Switch performs switching to one of the Inport cell according to its destination and redirect another Inport's cell to the next stage Banyan Switch.
- The 2nd stage Banyan Switch performs switching to the redirected cell according to its destination.
- At a later stage, the Multiplex Output Model, multiplex the cell from both the 1st and 2nd Stage Banyan Switch to the respective Outport. This can be done because the sample rate 2 (0.025ms) at Multiplex Switch is double the sample rate (0.05ms) at the 1st and 2nd Stage Banyan Switch. Hence, no major delay occurs during the transmission.
- Total cells received at the Outport equal to total cells generated from both Traffic Generators, as there is no cell loss in between the flow.

5.2 Discussion on Simulation Result

By studying cell flow pattern at the previous section, cell flow can be further categorised into two types, that is cell flows without blocking condition and with blocking condition. In the next section, Examples will show this two-cell flow type in a 2 Inports by 2 Outports ATM Banyan Switch. To make cell flow analyses easier, preset value for Traffic Generator 0 and 1 has been standardised before the start of the simulation. The preset value in details is shown below.

Traffic Generator 0 (Assume it is CBR Source)

MBIT TO SEND = 0.384MBIT ; BIT RATE = 19200 BIT PER TICK

Data Cell Generated per tick = Bit Rate / 48 Byte Data Cell
= 19200 bit per tick/ (48 * 8) bit
= 50 Data Cell per tick

Total tick required to send all Mbit = Mbit to send / Bit Rate
= 0.384 Mbit / 19200bit per tick
= 20 ticks

Traffic Generator 1 (Assume it is VBR Source)

MBIT TO SEND = 0.384MBIT ; BIT RATE = 38400 BIT PER TICK

Data Cell Generated per tick = Bit Rate / 48 Byte Data Cell
= 38400 bit per tick/ (48 * 8) bit
= 100 Data Cell per tick

Total tick required to send all Mbit = Mbit to send / Bit Rate
= 0.384 Mbit / 38400bit per tick = 10 ticks

Tick defined as cycle time for a specific sample rate. For example:

Sample Rate 1 (0.05ms) at Banyan Switch means one tick will required 0.05ms.

Sample Rate 2 (0.025ms) at Multiplexer Switch means one tick will only required 0.025ms.

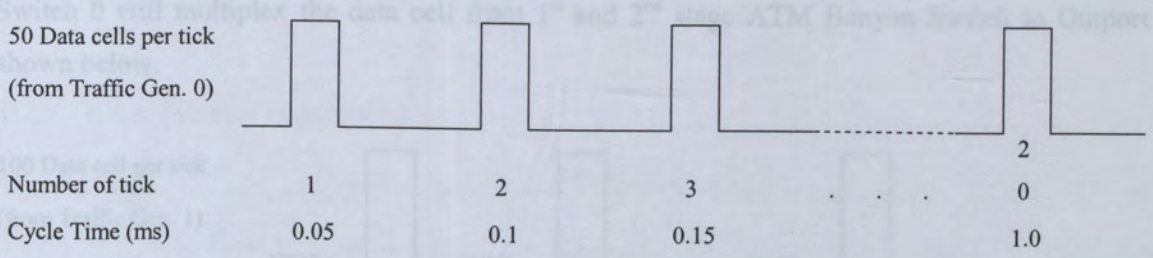
5.2.1 Example of Cell Flow without Blocking Condition

This example covers cell flow from one source to another without any blocking condition. This means that there is no cell from two sources that aimed for the same destination.

Under this example, the Outport that receives data cell from respective Traffic Generator, will show number of cell received and number of tick (total cycle time) as below:

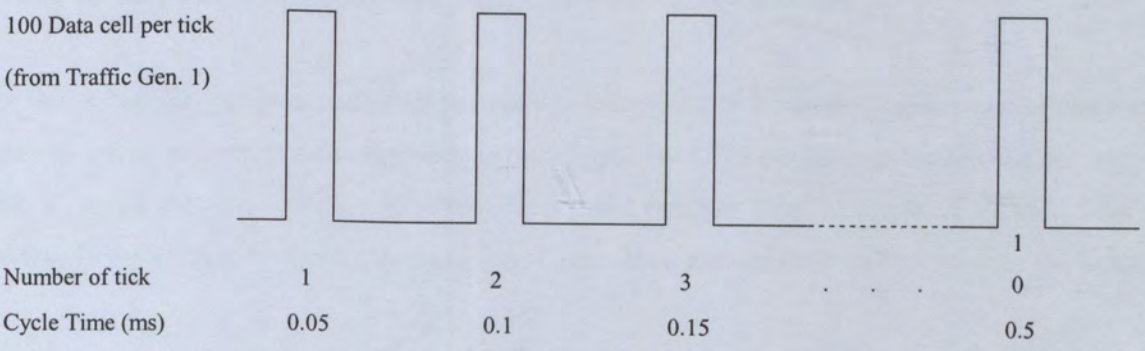
Traffic Generator 0

Traffic Generator 0 will generate 50 data cells per tick, and Banyan Switch requires total of 20 ticks to complete send out 0.384 Mbit data cells. The Outport which received data cell from Inport 0 show result in graph form as below:



Traffic Generator 1

Traffic Generator 0 will generate 100 data cells per tick, and it requires a total of 10 ticks to complete send out 0.384 Mbit data cells. The Outport, which received data, cells from Inport 1 shows the result in the graph form as below:

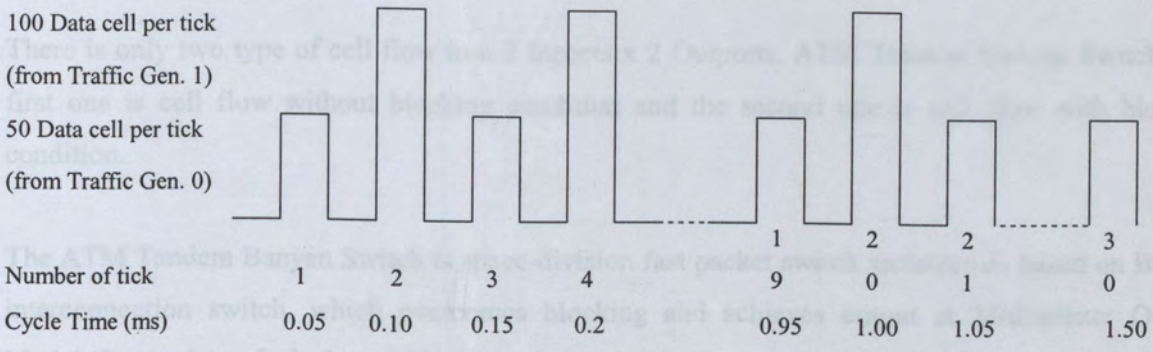


5.2.2 Example of Cell Flow with Blocking Condition

These example covers cell flows from two different Inports that aim for the same outport. When data cells from two Inports reach at the 1st stage Banyan Switch, only data cell from one of the inport pass through the switch. However, data cell from another Inport is given a second chance to pass through the 2nd stage Banyan Switch. The Multiplexer Switch will then multiplex the data cell from 1st and 2nd stage Banyan Switch, at Sample Rate 2 (one times faster than Sample Rate 1 at Banyan Switch) to relevant Outport.

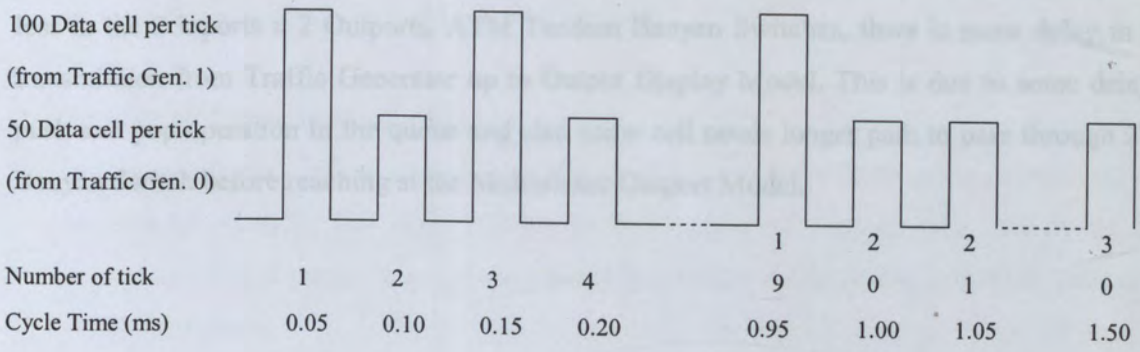
Data cell from Inport 0 and Inport 1 aims for Outport 0

Data cell from Inport 0 will be given privilege to pass through the 1st stage Banyan Switch. Data cell from Inport 1 will be redirected to pass through 2nd stage Banyan Switch. Later the Multiplexer Switch 0 will multiplex the data cell from 1st and 2nd stage ATM Banyan Switch to Outport 0 as shown below:



Data cell from Inport 0 and Inport 1 aims for Outport 1

Data cell from Inport 1 will be given privilege to pass through 1st stage Banyan Switch. Data cell from Inport 0 will be redirect to pass through 2nd stage Banyan Switch. Later the Multiplexer Switch 1 will multiplex the output from 1st and 2nd stage Banyan Switch to Outport 1 as shown below but with 100 data cell received first instead of 50 data cell.



5.2.3 Overall Result Discussion

There is only two type of cell flow in a 2 Inports x 2 Outports, ATM Tandem Banyan Switch. The first one is cell flow without blocking condition and the second one is cell flow with blocking condition.

The ATM Tandem Banyan Switch is space-division fast packet switch architecture based on Banyan interconnection switch, which overcomes blocking and achieves output at Multiplexer Outport Model. It consists of placing two stages of Banyan Switch in series, such that output of 1st stage Banyan Switch is connected to both the Inport of 2nd stage Banyan Switch in the series and the corresponding Multiplexer Outport Model. The exception is that the 2nd stage Banyan Switch for which the outputs are only connected to the Multiplexer Outport Model.

In this arrangement, upon a conflict between two data cell at 1st stage Outport, one of the two data cells is routed properly, while the other is routed into the 2nd stage Banyan Switch. At the Outport of the 1st stage Banyan Switch, the packets that have reached their Multiplexer Outport Model are placed in the corresponding Outport Buffers. Later, these data cell will be multiplex to the Outport.

Due to only two Inports at the ATM Banyan Switch, thus 2nd stages Banyan Switch is adequate to eliminate the cell loss rate at 1st stage Banyan Switch. Further more, the Sample Rate 2 at Multiplexer Switch, which is double the speed of Sample Rate 1 at Banyan Switch, enable Multiplexer to multiplex the data cell from 1st and 2nd stage of Banyan Switch without major delay.

The queue's buffer size set in the Multiplexer Outport Model is sufficient to cater the waiting time for data cell queuing in the buffer before it is their turn to pop out from the queue. Despite no cell loss in the 2 Inports x 2 Outports, ATM Tandem Banyan Switches, there is some delay in the cell transmission from Traffic Generator up to Output Display Model. This is due to some delay at the push and pop operation in the queue and also some cell needs longer path to pass through 2nd Stage Banyan Switch before reaching at the Multiplexer Outport Model.

5.3 Assumptions

A number of assumptions have been made for the above simulation:

1. All the traffic sources generate cells at the same time when simulation begins.

   In this assumption, all the sources are synchronized to begin generates cell at the same time. In reality, this might not be true, cells can be generated randomly by the source or it can be generated intermittently or can be suddenly stopped.

2. All cells will have equal priority.

   In this assumption, cell generated from both traffic sources are considering constant bit rate and variable bit rate type. This again may not be true in an actual situation. For CBR traffic, the QoS is guaranteed and the cells are given priority but in this model, CBR and VBR cells are given equal priority.

3. No traffic management is needed

   The simulator does not incorporate any of the traffic management function because it treats all cells having equal priority. In reality, traffic management is needed to guarantee the Qos of each cell.

4. There will be no congestion during cell transmission.

   Assumption made in the simulation for every cell flow is smooth. In reality, the output buffer or link can be full and this will cause congestion during cell transmission. Buffer storage limit is set at a high limit, in order to allow more cell data store at the buffer.

5. The network resources are always available.

   Network resources are adaptable and realistic model will have to be considered not only resource variation but also rapid depletion of critical resource requirements.

6. No delay will occur in the Traffic Generator and Switch.

   Hardware components have inherent delays and a more realistic model will have to consider this limitation.

7. FIFO method are applied for buffering / queuing at Traffic Generator and Switch.

   This method of buffering is reasonably fair but does not take into consideration QoS for CBR traffic. Although the simulation considers two types of traffic CBR and VBR at the source, the QoS in terms of bandwidth allocation (Contract) is not fully complied with.

8. Request calls for establishing and releasing connection are not required.

   Connection admission control is the first defense in protecting from excessive load. When a user requests a new VCC, the user must specify the service required in both directions for that connection. This model adopted does not consider any connection control.

9. All the physical links are unidirectional.

   A computer (end system) can be a source or sink and the links are bi-directional. In the model adopted, the traffic flow is unidirectional.

5.4 Advantage of Dynamic Modelling an ATM Banyan Switch

- Simple architecture of ATM Banyan Switch

  The 2 Inport by 2 Outport ATM Banyan Switch is the simplest architecture to be modelled. The 16 cell flow scenarios that exist within the cell fabric are easier to be expressed by the Boolean logic which represented by a switching algorithm model using Simulink tool.

- User friendly modelling and simulation environment

  Simulink provides a user-friendly modelling environment through the graphical user interface (GUI) for building models as block diagram. By using click-and drag mouse operation, model can be drawn and refined efficiently. Blocks in a model can be coloured based on their functionality to differentiate from others to increase the readability of complex models. Subsystem blocks can be created to using top to bottom approach to reduce the complexity of a model. This help user easier understand the ATM Banyan Switch architecture

- Simulation sample time and necessary parameters can be set and changed through the dialog boxes provided. For example, the parameters for Random Number block include mean and variance of number, and the initial speed sample time of the block. User can interact with the model dynamically.

- Interactive simulation environment

  Simulink is a powerful tool for simulating dynamic system. The simulation environment is interactive in a sense where the parameters can be changed on the spot. The effect of parameter changes can be captured immediately even during the simulation's execution. In an ATM Banyan Switch, many parameters influence the performance of the switch such as the buffer size and traffic condition. With this Simulink's feature, ATM Banyan Switch model can be simulated in a more effective way

- Expandable and Reusable

  ATM Banyan Switch model build from different type of sub models. With the existing Simulink block libraries, Simulink enables users to customize and create their own block. Block created can be reused and expanded to form a more complex model, this help simplifies and increase the readability of a complex model.

5.5 Limitation of Dynamic Modelling an ATM Banyan Switch

- Inability to recognize non-numeric signal

  All the Simulink signal must be in numeric form such as integer or real. This has caused some inconveniences for the ATM network simulation. Due to the variety of ATM signal form, for example, the ATM cells traffic type: CBR, VBR is in character form. In order to send the cell traffic type signal, they have to be represented by numeric.

- Inability to send grouped data using one signal

  One signal can only send one data in Simulink. For an ATM cell, at least three types of data are to be sent in one signal – source, traffic type and destination. However, this problem can be resolved by multiplexing the signals before sending and demultiplexing after receiving, by using Simulink's MUX and DEMUX block.

5.6 Areas of Enhancements

- Virtual Connection

  The virtual connection simulated is Permanent Virtual Connections that are actually not common at current ATM virtual connection development. It is chosen because it helps to simplify the process of establishing and releasing of virtual connection and makes the model simulated, to be simple. In reality, the switch plays a role in establishing and releasing of virtual connection. For future works, Switched Virtual Connection can be simulated to reflect the current ATM network development more accurately.

- Switch Architecture

  The current simulated switch architecture is 2 Inports x 2 Outport Two Stages ATM Tandem Banyan Switch. The model developed includes the reusable and expandable criteria. It can be easily modified to model 4 Inports x 4 Outport Four Stages ATM Tandem Banyan Switch. It can also be modified to model an 8 Inports x 8 Outport Knockout Tandem Banyan Switch. Traffic management and congestion control can be incorporated into the switch so as to make the model more comprehensive.

- Traffic Generator

Currently, one Traffic Generator can just connect to one application that is not true for the current development. Traffic Generator should be able to cater for multiple application, which are dynamically set by the users.

- Traffic

  The source traffic can be modeled using some traffic modeling methods such as Poisson and Self-Similar. Traffic characteristics such as burstiness nature and arrival pattern can be taken into consideration. By taking traffic characteristic into consideration, the simulated results will be more realistic.

5.7 Overall Conclusion

This project is intended to model ATM Tandem Banyan Switch using Simulink tools. ATM switch, simulation concepts, and Simulink tools have been studied, so as to dynamically model a Two-State ATM Tandem Banyan Switch. Simulink is found to be a good simulation tool to simulate dynamic systems. The variable functionality of Simulink block has given users the flexibility to build their own model efficiently. Customized model can be built easily in a user-friendly environment through the graphical user interface. It is the reusable and expandable criteria that help users to build a complex model, with high readability, though due to the allowance of using top-down approach to group the complex model into a simple sub system.

By simulating ATM Tandem Banyan Switch, researchers are able to study the ATM Tandem Banyan Switch architecture and understand its functionality. ATM switch designers can design different ATM switch architecture and simulate under different traffic load, in order to obtain various simulation results. By going through this simulation, optimal performance and optimal switch architecture design can be expected.

Although fairly simple ATM switch architecture was simulated, the model developed and experience gained will enable more complex switch architecture to be developed. This will enable a designer to study other issues related to performance of a switch, for example traffic management, congestion control and routing algorithm.

The models explored are based on various assumptions listed in section 5.3. The model adapted is idealistic and does not closely represent the real ATM Switch environment, It has be emphasized in assumption 4 and 6 in section 5.3. In an actual network environment, congestion will be present during cell transmission and any realistic simulation should take this factor into account. Delays in Traffic Generation and ATM Switch are the factors that need to be considered in the hardware design of ATM Networks.

Future enhancement using Simulink in simulation design will relax some of the assumptions made. It will help to create a more realistic model of an ATM switch. An attempt will also be made to take the enhancement listed in section 5.6 into consideration to reflect the real ATM switch and its component behavior, more accurately.

BIBLIOGRAPHY

[BIRA 94] Giora Biran, "*Introduction to ATM Switching*". http://www.rad.com/networks/1994/gbrian/atm_swi.htm, 1994.

[CHEN 95] Chen, Thomas M. and Stephen S. Liu, "*ATM Switching System*", Arteh House, Incorporated, 1995.

[CHIO 99] John A. Chiong. "*Internetworking ATM for the Internet and Enterprise Networks*". McGraw-Hill. 1999.

[CHXI 94] Chen, Xing et al., "*Performance Comparison of Two Input Access Methods for a Multicast Switch*". IEEE Transactions on Communications Vol 42 Nol 5, May 1994.

[FAHM 95] Sonia Fahmy. "*A Survey of ATM Switching Techniques*". http://www.cis.ohio-state.edu/~jain/cis788-95/atm_switching/ , 1995.

[FISH 94] Fishwick, P.A. "*Computer Simulation: Growth Through Extension*", European Simulation Multiconference, Barcelona, Spain 1994.

[FLOO 95] Flood, J. E. "*Telecommunications Switching, Traffic and Network*", Prentice Hall, 1995.

[JAIN 95] Raj Jain. "*Congestion Control and Traffic Management in ATM Networks: Recent Advances and A Survey*", Computer Networks and ISDN Systems, Vol 28, Nol 3, Frebruary 1995.

[LUFA 95] Fang Lu, "*ATM Congestion Control* ", http://www.netlab.ohio-state.edu/~jain/cis788-95/atm_cong/index.htm, Department of CIS at OSU, 1995.

[NIST 94] Nada Golmie, et al. "*The NIST ATM/HFC Network Simulator: Operation and Programming Guide Version 4.0*". U.S. Department of Commerce. December 1994.

[ONVU 95] Onvural, Raif O., "*Asynchorous Transfer model Networks: Performance Issues*", Bosto: Artech, 1994.

[OPNE 99] Opnet - A Telecommunication Network Simulation Package. http://www.ee.ucl.ac.uk/dcs/commercial/opnet/opnet.html, Department of Electronic & Electrical Engineering, UCL, 1999.

[POOC 93] Udo W. Pooch and James A. Wall. "*Discrete Event Simulation: a practical approach*". CRC Press Inc. 1993.

[ROBE 93] Rober Fazzi, Thomas G., "*Performance Calculation of high speed switching fabrics and networks: ATM, broadband ISDN and MAN Technology*". IEEE Press, 1993.

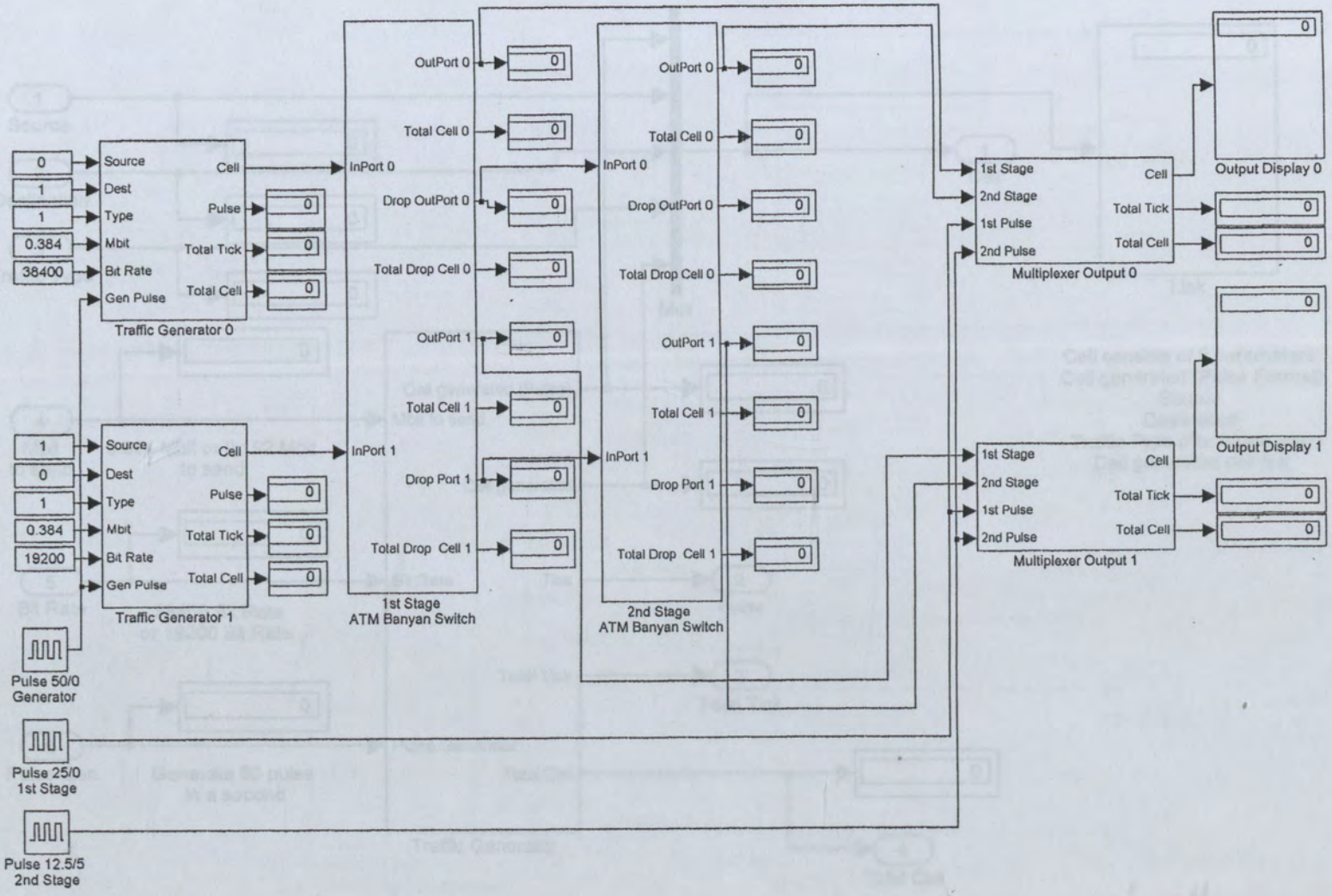
[STAL 97] William Stallings. *Data and Computer Communications*. Prentice Hall. New Jersey, 1997.

[STAL 98] William Stallings. *High Speed Networks: TCP/IP and ATM design principles*. Prentice Hall. New Jersey, 1997.

[SIMU 99] *Using Simulink, Version 3.* The MathWorks, Inc. 1999.

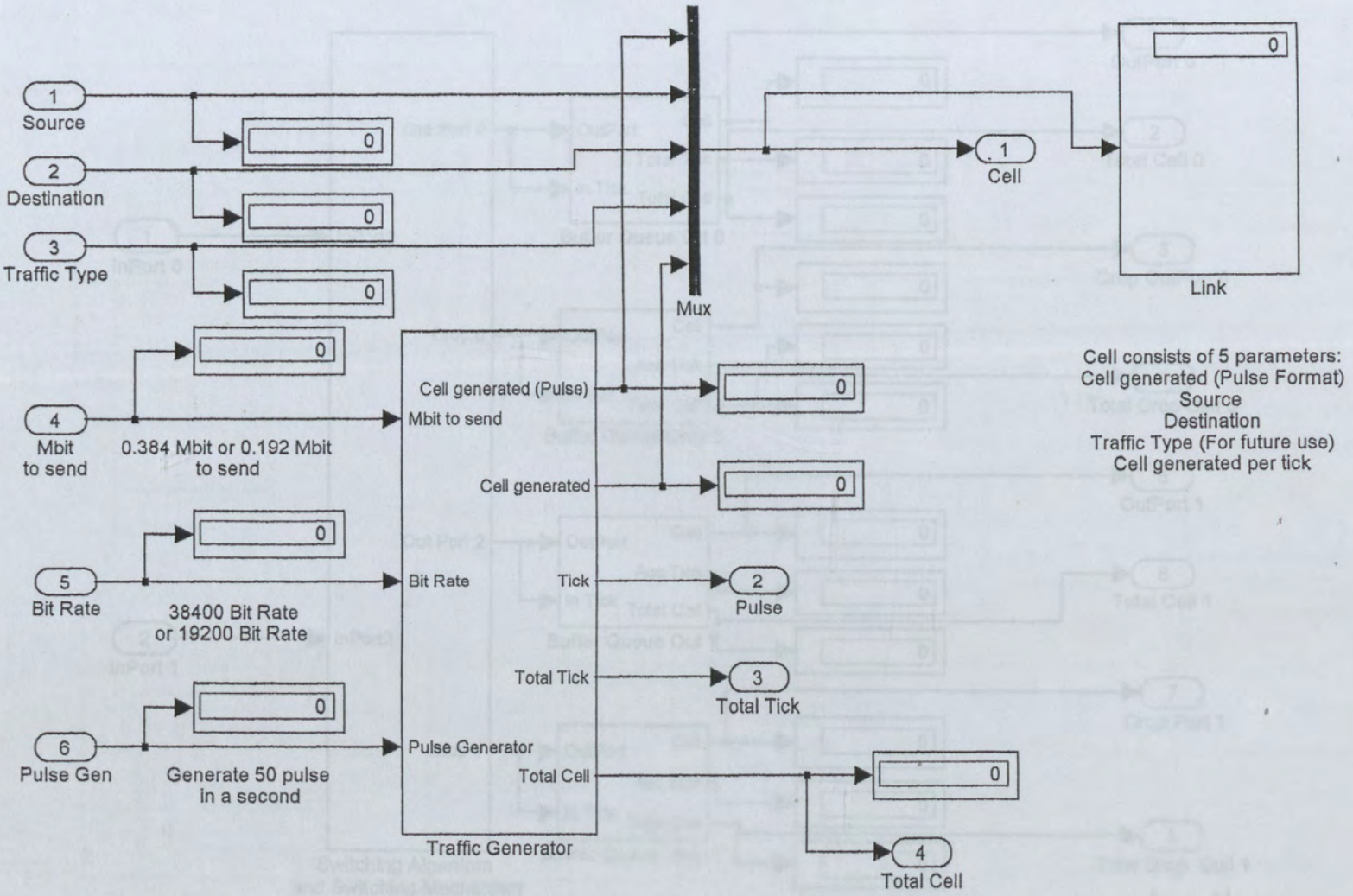
[TAYS 00] Tay Sok Kem, "*Dynamic Modelling of cell flow for an ATM Network*", Faculty of Computer Science and Information Technology, 2000.

[VIST 99] VISTA: *A configurable Visualization and Simulation Tool for ATM switches*. http://vip.cs.edu/systems/pubs/nrthesis/ch2/ch2_3.html, 1995.



c:\wrga project thesis\atm tandem banyan switch.mdl

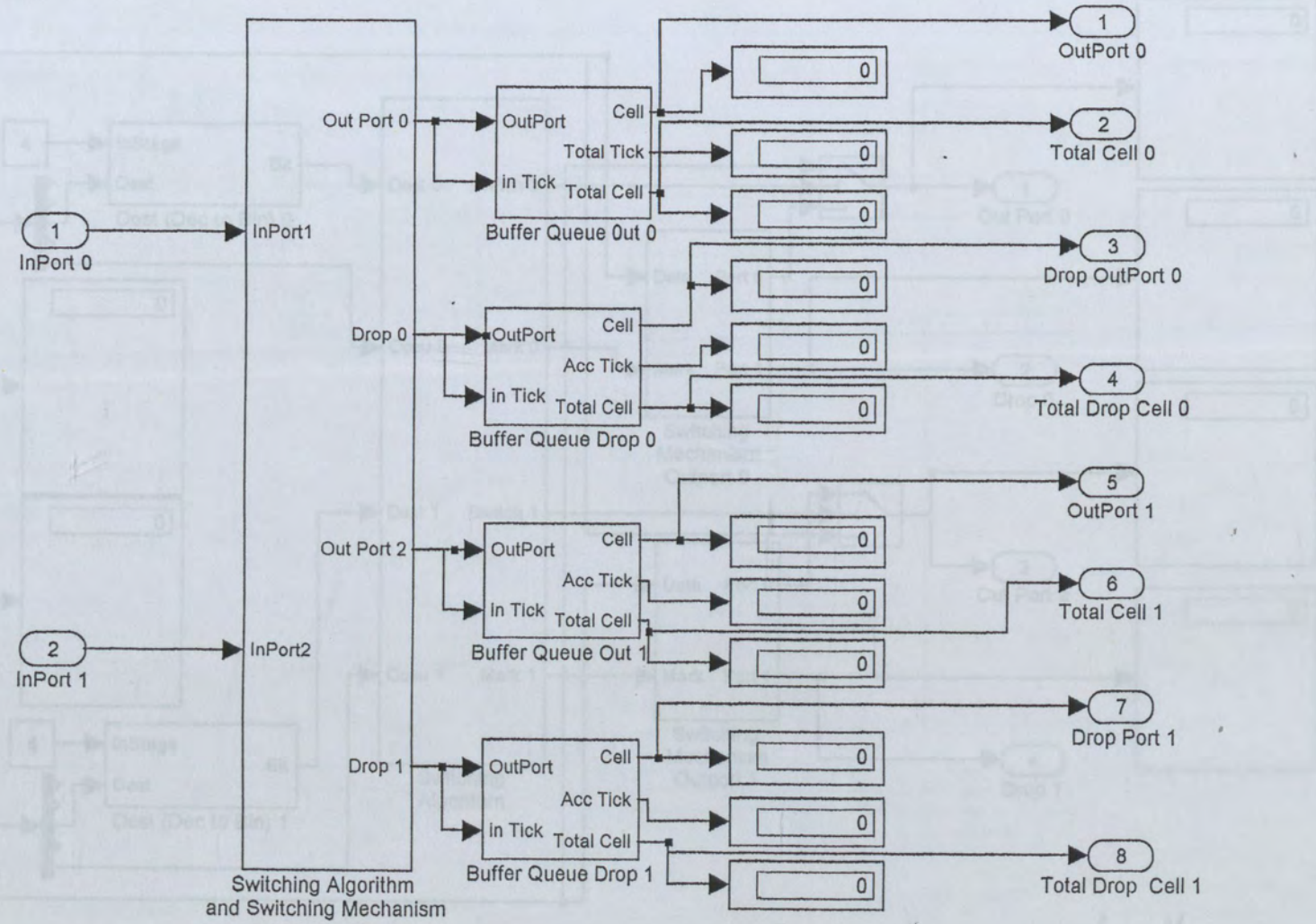
printed 12-May-2001 17:58

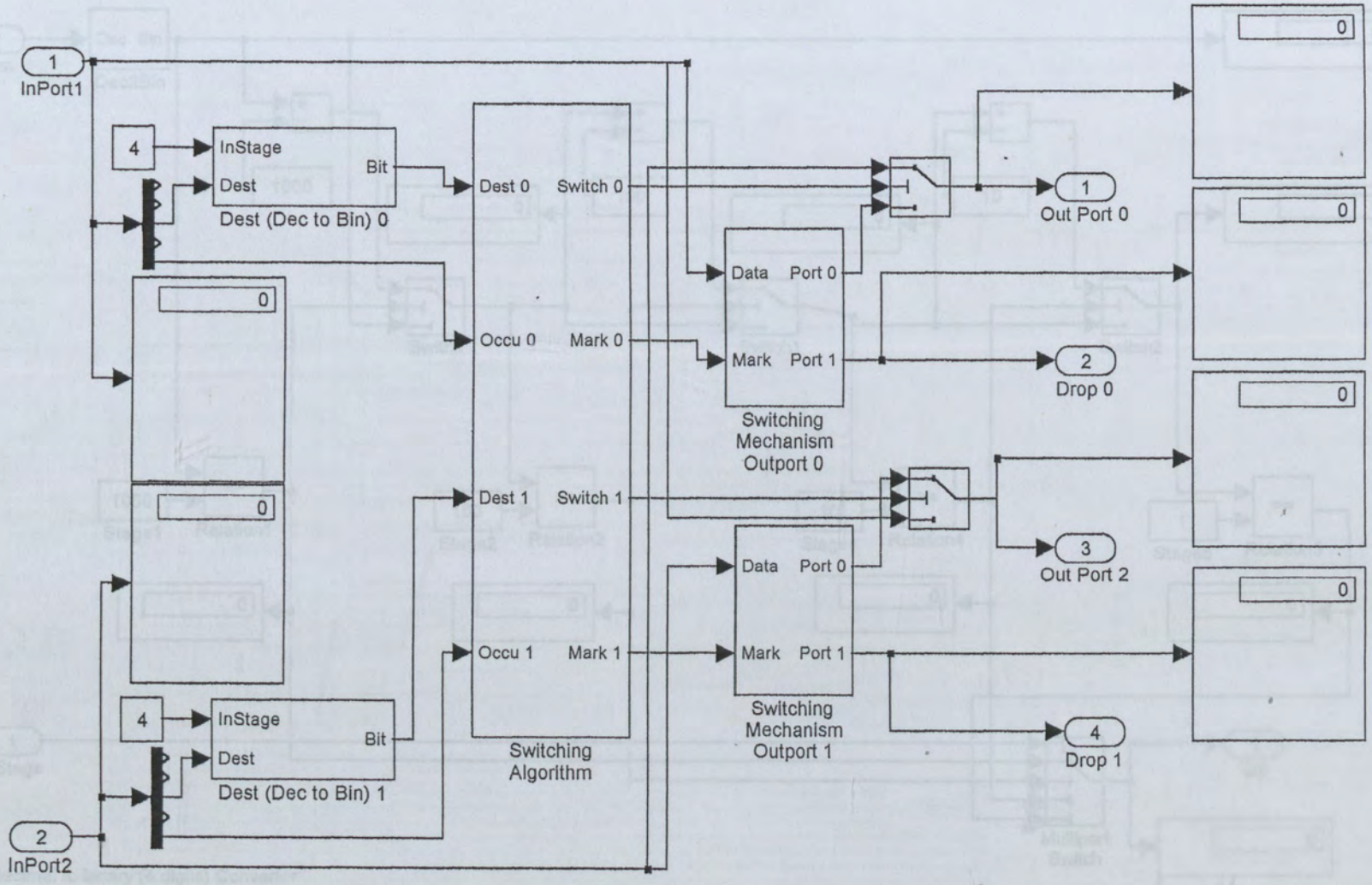


c:\wrga 6180 project thesis\atm tandem banyan switch.mdl

printed 12-May-2001 18:41

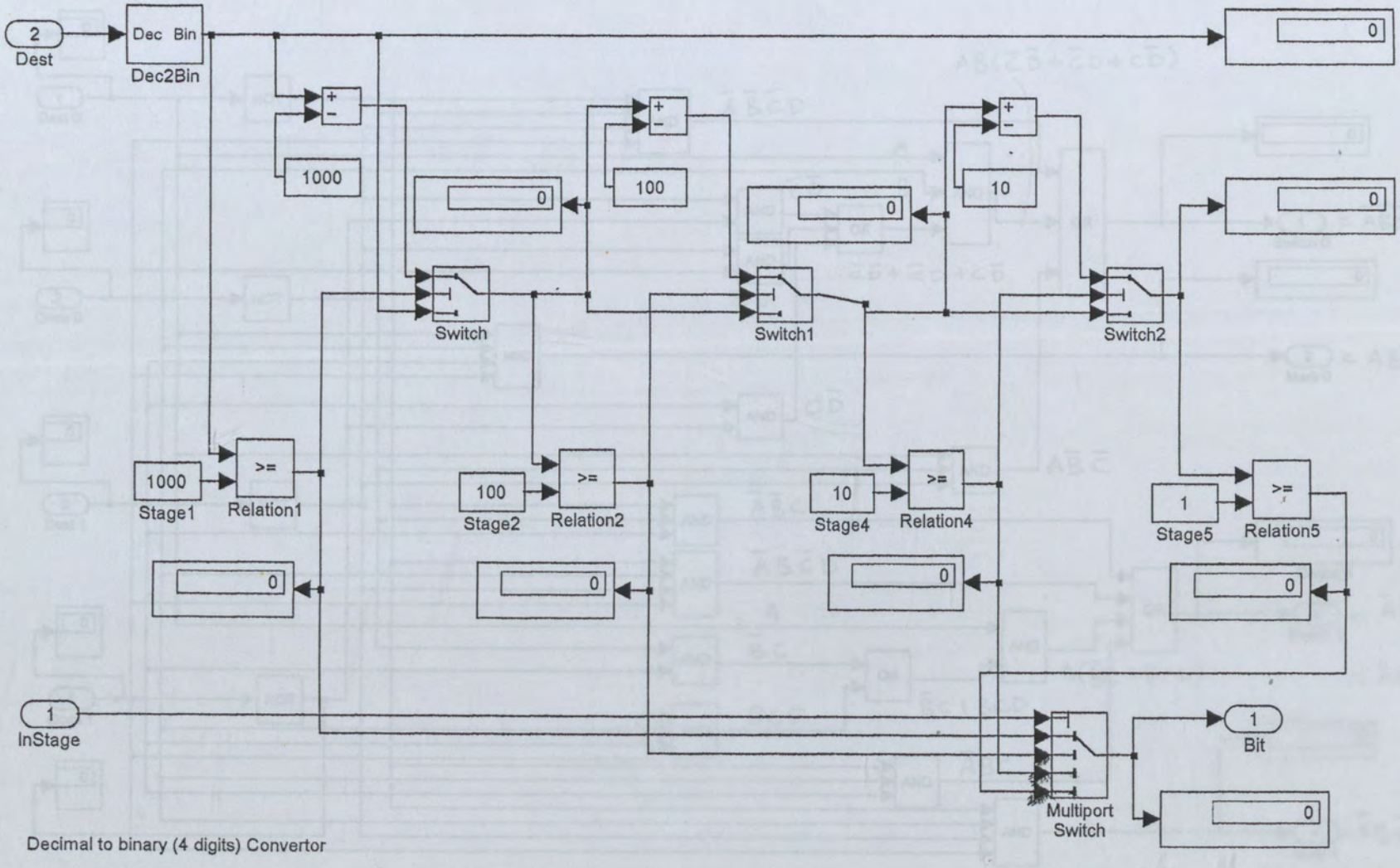
page 1/1

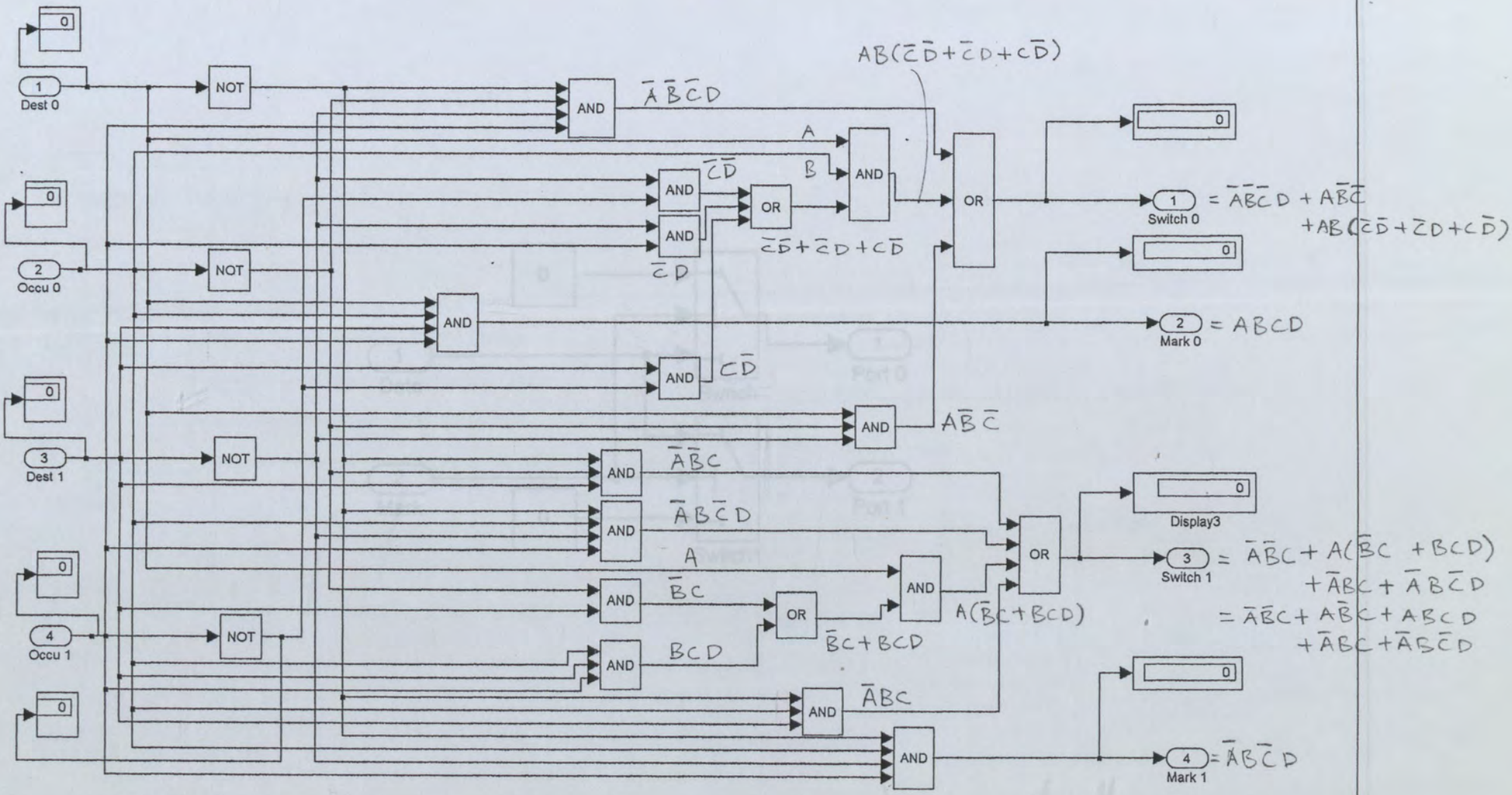

1
InPort 0
InPort1
2
InPort 1
InPort2
Switching Algorithm
and Switching Mechanism
Out Port 0
Drop 0
Out Port 2
Drop 1
OutPort
in Tick
Cell
Total Tick
Total Cell
Acc Tick
Buffer Queue 0ut 0
Buffer Queue Drop 0
Buffer Queue Out 1
Buffer Queue Drop 1
0
1
OutPort 0
2
Total Cell 0
3
Drop OutPort 0
4
Total Drop Cell 0
5
OutPort 1
6
Total Cell 1
7
Drop Port 1
8
Total Drop Cell 1



c:\wrga 6180 project thesis\atm tandem banyan switch.mdl

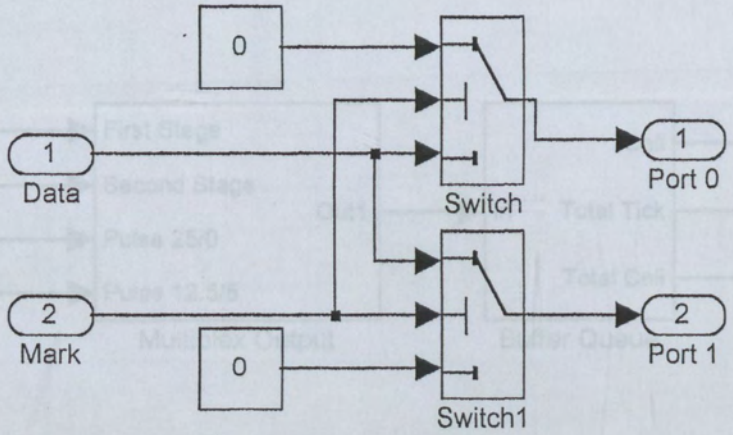
printed 12-May-2001 19:02

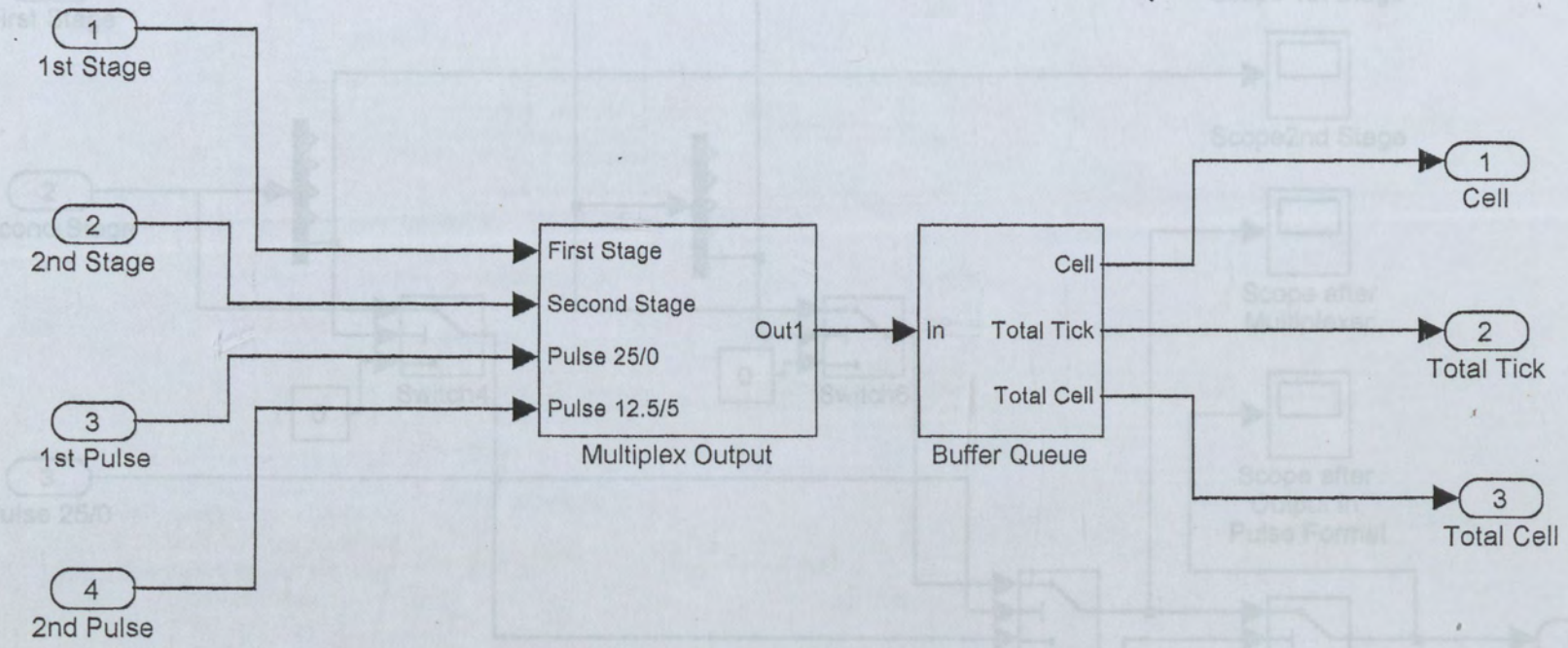

2
Dest
Dec Bin
Dec2Bin
0
1000
0
100
0
10
0
Switch
Switch1
Switch2
1000
Stage1
Relation1
100
Stage2
Relation2
10
Stage4
Relation4
1
Stage5
Relation5
0
0
0
0
1
InStage
1
Bit
Multiport
Switch
0
Decimal to binary (4 digits) Convertor

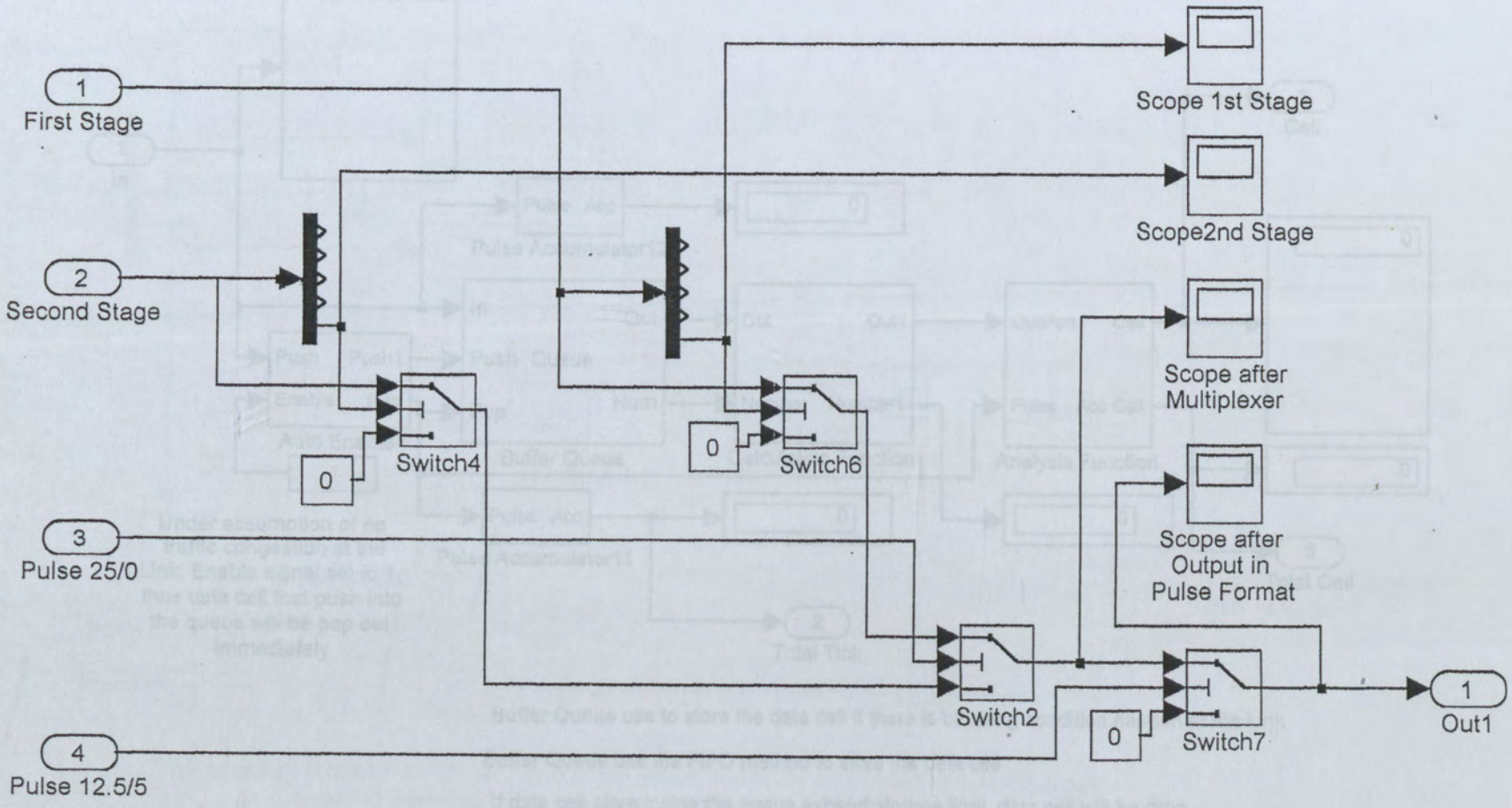


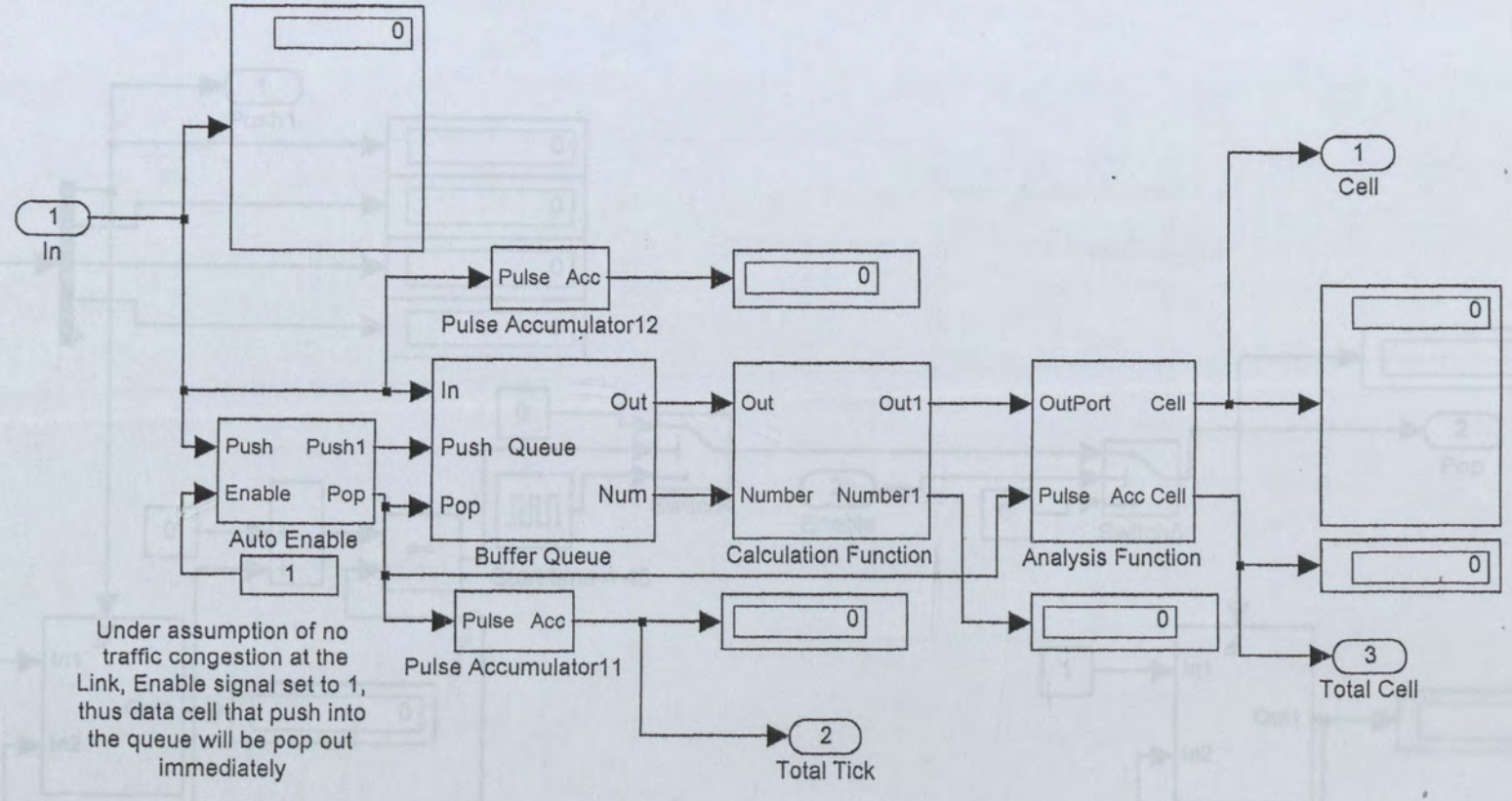
c:\wrga 6180 project thesis\atm tandem banyan switch.mdl

printed 12-May-2001 18:55


0
1
Data
Switch
1
Port 0
2
Mark
0
Switch1
2
Port 1


1
1st Stage
2
2nd Stage
3
1st Pulse
4
2nd Pulse
First Stage
Second Stage
Pulse 25/0
Pulse 12.5/5
Out1
Multiplex Output
In
Cell
Total Tick
Total Cell
Buffer Queue
1
Cell
2
Total Tick
3
Total Cell


1
First Stage
2
Second Stage
3
Pulse 25/0
4
Pulse 12.5/5
Scope 1st Stage
Scope2nd Stage
Scope after
Multiplexer
Scope after
Output in
Pulse Format
Switch4
Switch6
Switch2
Switch7
0
0
0
1
Out1

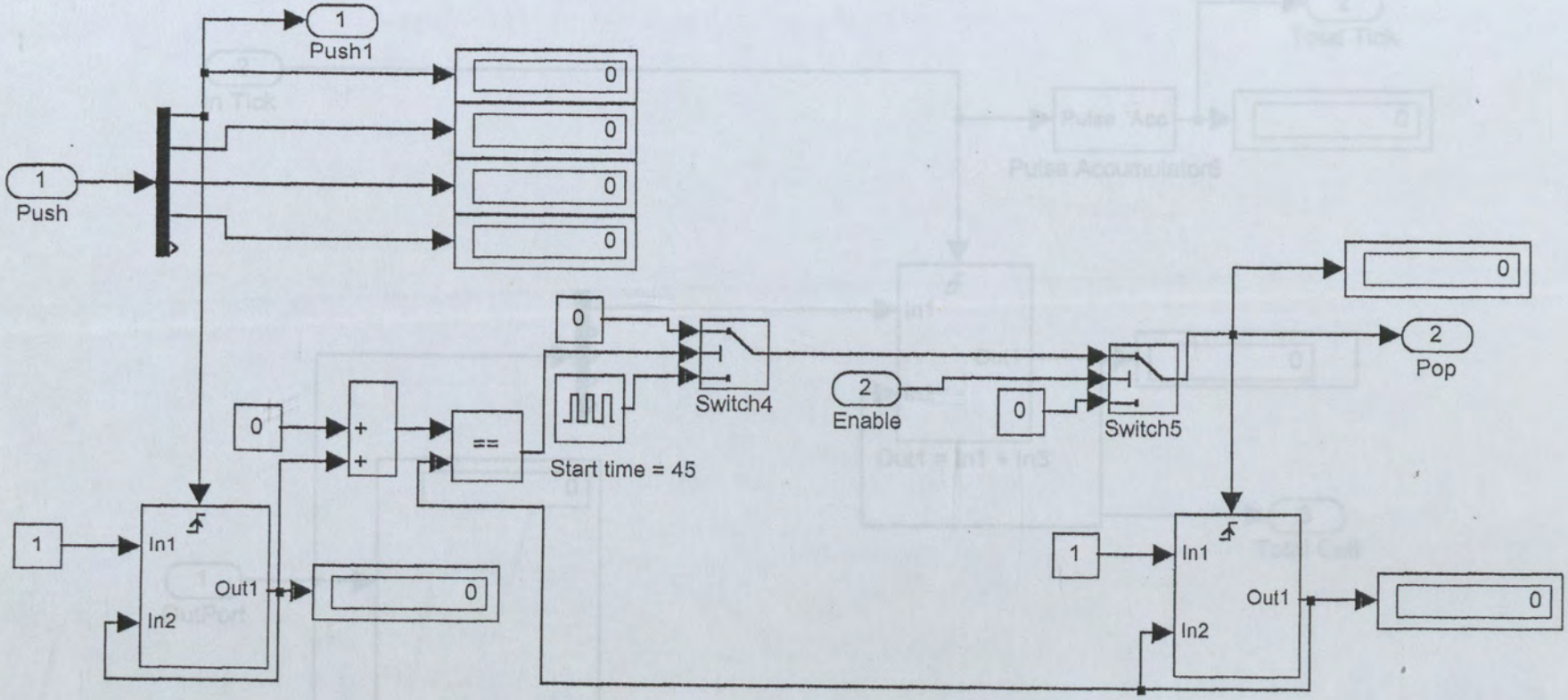


Buffer Queue use to store the data cell if there is blocking condition happen at the Link

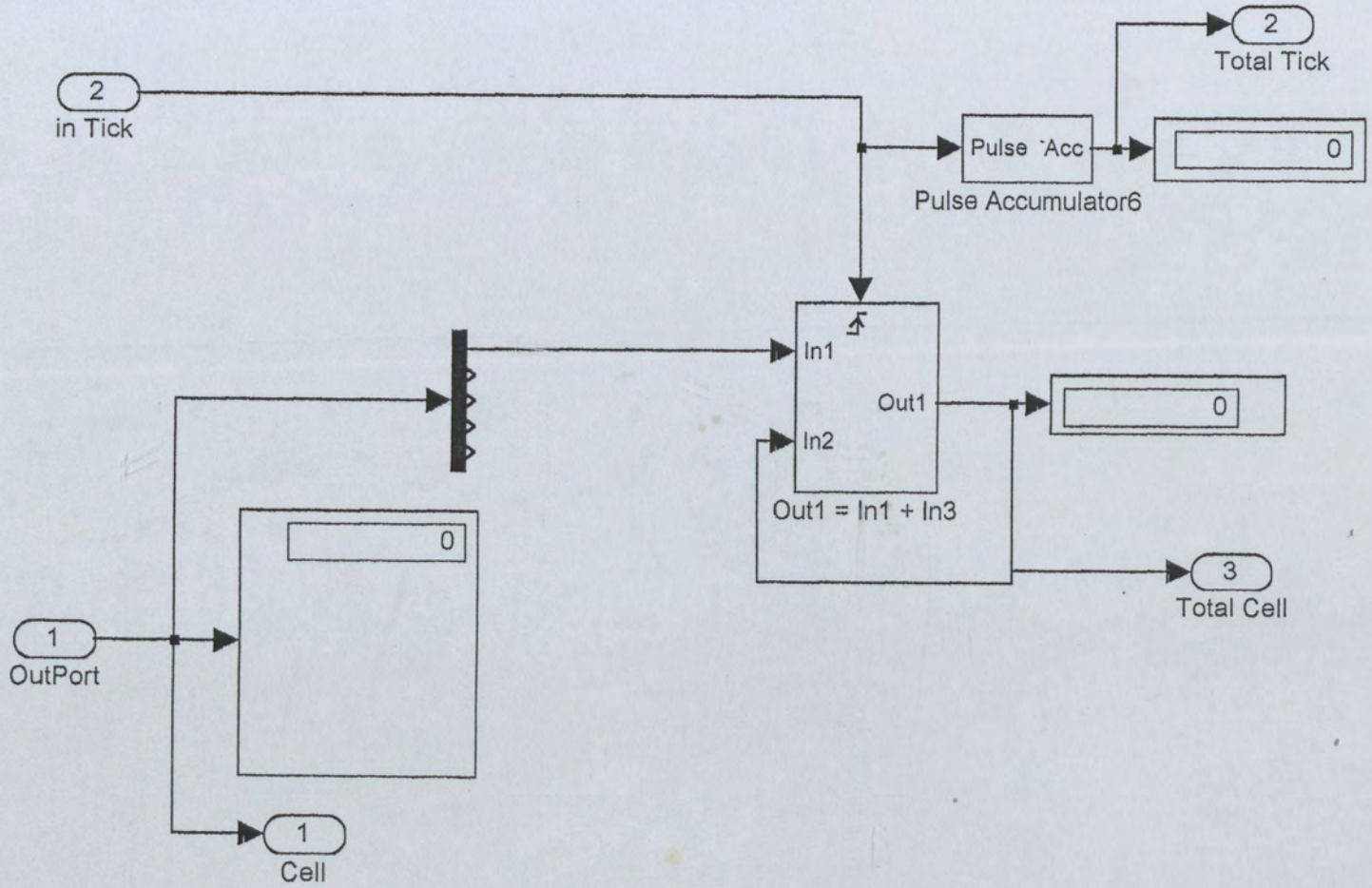
Buffer Queue use the FIFO method to store the data cell

If data cell store inside the queue exceed storage limit, data cell will be drop

Total number of cell that push in must be equal to total number of cell pop out from the queue



This algorithm is to provide enable signal to the queue, and to generate pop signal to the queue once the queue received data cell from the Link (Push signal trigger)



c:\wrga 6180 project thesis\atm tandem banyan switch.mdl

printed 12-May-2001 19:08