Design of VLSI Packet Switch Element

James Sternbenz

This paper describes the design of the Packet Switch Element Chip, one of the components of a high speed broadcast packet switching network. The chip is a 2x2 switch element, from which a switch fabric of arbitrary size can be constructed. It is currently being implemented in 3um, CMOS technology.

Follow this and additional works at: https://openscholarship.wustl.edu/cse_research

Recommended Citation
DESIGN OF A VLSI PACKET SWITCH ELEMENT

James Sterbenz

WUCS-88-05

April 1988

Department of Computer Science
Washington University
Campus Box 1045
One Brookings Drive
Saint Louis, MO 63130-4899

Abstract

This paper describes the design of the Packet Switch Element Chip, one of the components of a high speed broadcast packet switching network. The chip is a 2x2 switch element, from which a switch fabric of arbitrary size can be constructed. It is currently being implemented in 3μm CMOS technology.

On leave of absence from IBM Corporation at Washington University in St. Louis. This work supported by Bell Communications Research, Italtel SIT, NEC and National Science Foundation grant DCI-8600947.
DESIGN OF A VLSI PACKET SWITCH ELEMENT

James Sterbenz  
jps@wuccrc.UUCP

1. Introduction

This paper describes the design of the Packet Switch Element Chip (PSE) for use within a broadcast packet switching network. There are two forms of this report. The high level design is described in [St88a]. The detailed design is described in [St88b], which is identical to [St88a], with the addition of appendix C describing the design and chip layout for all cells in the PSE. The broadcast packet switching network is described in [Tu85] and [Tu86].

1.1. Packet Switch

The overall structure of the prototype packet switch is shown in Figure 1. The switch module terminates 7 fiber optic links, and consists of five major components: the connection processor (CP), packet processors (PPs), copy network (CN), broadcast translation circuits (BTCs), and routing network (RN). The CN, BTCs, and RN are collectively known as the Switch Fabric (SF), and are

Figure 1: Switch Module
constructed from banyan-interconnected PSEs. The PPs serve as buffers between the transmission links and also perform the address translation required by routing. The RN routes the packets to the appropriate output link based on information encoded in the routing field (RF) of the packet. Since a banyan interconnection is self-routing, the PSEs need only examine a single address bit to select the proper output port. The CN replicates broadcast packets to multiple output links based on information in the RF, indicating the fanout of the packet. Point-to-point packets are arbitrarily routed through the CN. The distribution network (DN), not present in the prototype, will randomly distribute packets before the RN to avoid congestion occurring from persistent poor traffic patterns (resulting, for example, from communities of interest).

The format of the packets processed by the PSE is shown in Figure 2. The packet is organized as a sequence of 4-bit words. Each packet contains exactly 80 words, the first six of which constitute the packet header. Only the first four words, the routing field (RF), are used by the PSE. The meanings of the fields are:

- **Routing Control (RC).** This word determines how the packet is processed by the switch elements. 0000 signifies an empty packet slot, 0001 signifies a normal point-to-point data packet, 0011 signifies a normal broadcast packet, 0111 signifies a test packet.

- **Fanout (FAN).** If RC = 0011, the second word of the packet is taken to be the fanout, that is the number of switch fabric output ports that require copies of the packet.
• **Address (ADR).** If $RC = 0001$, the second word of the packet is taken to be the address of the packet, that is, the switch fabric output port to which the packet is to be delivered. The second word of the header is generically designated $FA$ (fanout/address).

• **Broadcast Channel Number (BCN).** If $RC = 0011$, the third and fourth words of the packet are taken to be the broadcast channel number. All packets within a particular multi-point connection have the same broadcast channel number.

• **Logical Channel Number (LCN).** If $RC = 0001$, the third and fourth words of the packet are taken to be the outgoing logical channel number. On the external fiber optic links, logical channel numbers are used to identify which connection a packet belongs to. The third and fourth words of the header are generically designated $BH$ and $BL$ respectively.

• **Control Field (CTL).** This word indicates various types of control packets, and is not used by the PSE.

• **Source (SRC).** This word identifies which switch fabric input port the packet came from, and is not used by the PSE.

• **Information (I).** This field normally contains user information. In the case of control packets, additional control information may be contained.
2. Chip Specification

2.1. Operation

A single PSE circuit is used within the routing, copy, and distribution networks that comprise the switch fabric. PSE routing decisions are based on the operation mode (OM) and packet type (RC in the packet header). The operation modes are:

- **Route** — Route all packets based on the appropriate bit of the ADR field of the packet header.
- **Copy** — Replicate packets based on the FAN field of the packet header.
- **Distribute** — Distribute packets arbitrarily and uniformly between outputs.

The packet types are:

- **Point-to-point** — Use ADR to route packets only for route operation mode.
- **Broadcast** — Use FAN to replicate packets only for copy operation mode.
- **Test** — Use ADR to route packets in all operation modes.

When arbitrary routing decisions can be made, the following policies are followed:

- Ties among input ports for the same output port are broken such that the last input port favored loses, to avoid individual starvation.

- Packets that can be routed to either output are uniformly distributed between output ports based on an output toggle. This includes all packets in the distribution network, and point-to-point and unreplicated broadcast packets in the copy network.

- Packets requiring both output ports in the copy network are favored over packets needing only one output.

- Packets requiring a specific output port are favored over packets requiring either output.

All packets are routed in the RN based on the ADR. The PSEs are interconnected to form a banyan network, which is self-routing. The routing algorithm, is simply to route to output port 0 or 1, depending on whether the bit corresponding to the switch element stage number (SN) is a 0 or 1.

Broadcast packets are replicated in the CN such that the number of packets leaving the CN is equal to original FAN of the packet. Each PSE makes a decision to copy if the FAN is greater than half the number of reachable outputs, specifically if $FAN > 2^{SN}$. The FAN of each replicated packet is adjusted to reflect the number of copies that must be subsequently made in the CN. This copy algorithm delays packet replication as long as possible to reduce congestion in the CN. Point-to-point packets, and broadcast packets not requiring replication in a particular PSE, are routed based on the distribution algorithm described below.

Packets in the distribution network are randomly distributed among the outputs of each PSE. This is accomplished by toggling a favored output bit for each input port, and thus distributing packets evenly across the outputs of the switch element.

Each PSE operates under control of a packet cycle, indicating the timing of packet arrival. A packet cycle is initiated by packet-time going high, indicating that data is present on the upstream
data lines at the input to the PSE. The Packets are either buffered, or if possible cut-through directly to the appropriate output port. Each input is capable of fully buffering a single 80 word (4 bits + parity) packet. If a packet is cut through, it appears on the downstream data lines at the appropriate output of the PSE eight clock cycles later. Successive words of the packet are received/transmitted every clock cycle.

Before the beginning of each packet cycle, a corresponding grant cycle begins, indicated by grant-time going high. During the grant cycle, grants are propagated backward through the fabric, indicating that successive switch element stages will be able to accept packets (either due to an empty buffer slot, or knowledge that a packet will be able to leave an occupied slot during the upcoming packet cycle). Grants are passed from downstream grant to upstream grant of neighboring PSES.

2.2. PSE Interface

The external interface signals of the PSE are shown in Figure 3 and are described briefly below. Signals incorporating multiple lines are represented collectively by square brackets, e.g. x[2:0] refers to x2, x1, x0, x[ab][10] refers to xa1, xa0, xb0, xb1, and x[1:0][10]' refers to x1, x0, x1', x0.'
• *Upstream data leads* (ud[a:b] [p3210]). Incoming data from upstream switch elements; 4-bits data plus odd parity for each input side of switch (a,b). Parity is inverted and maintained as even internal to the switch element.

• *Downstream data leads* (dd[10] [3210]). Outgoing data to downstream switch elements; 4-bits data plus odd parity for each output port of switch (0,1).

• *Upstream grants* (ug[a:b]). Grant signals to upstream neighbors, granting permission to transmit a packet during the next packet cycle.

• *Downstream grants* (dg[10]). Grant signals from downstream neighbors, indicating permission has been given to transmit a packet on the corresponding dd port during the next packet cycle.

• *Packet-time* (pt). Goes high when first word of a packet is present on the ud leads.

• *Grant-time* (gt). Goes high when valid grants are present at the dg inputs.

• *Operation mode* (om[10]). Switch element operation mode: 01=RP, 10=DN, 11=CN.

• *Stage number* (sn[10]). Indicates the switch network stage of which the PSE is a part. The furthest downstream column of switch elements is stage 0.

• *Rotate rotating field* (rrf). Causes the PSE to rotate the second through fourth words of test packets.

• *Error flag* (err). Report errors, including parity errors, packet header validity errors and framing errors (with respect to pt).

• *Hard reset* (rh). Initialize internal state machines and reset err.

• *Soft reset* (rs). Reset error flag err.

• *Clock* (φ1, φ2). Two-phase, symmetric, non-overlapping clock, duty cycle ≈ 25 % per phase.

• *Power and Ground* (Vdd, Gnd).
3. High Level Design

The PSE is a buffered 2-input, 2-output switch with broadcast capability. The main data path is symmetric with respect to each side of the switch element (a,b) and each output (0,1). Control and arbitration logic is shared across both halves of the switch. A block diagram is shown in Figure 4.

The major functional blocks of the switch element are:

- **Input Shift Register** — The two input shift registers (ISR, one per switch input) are 6 stage 5-bit static shift registers with an output tap after the first stage. Packets are shifted into each input shift register from the corresponding upstream data lines. The first stage of shift delay is for synchronization with pt. Parity is checked after the first shift stage. The remaining 5 stages allow sufficient time for control and routing decisions to be made by packet header and output enable decision logic.

- **Packet Header Logic** — Each packet header logic block (one per switch input) receives the packet from the output of the first stage of the corresponding input shift register. The first 4 words, comprising the packet header, are loaded into four 4-bit registers. The logic decodes the packet type, and determines the output port(s) requested based on packet type, switch operation mode, and FAN or ADR fields. Invalid header information sets error flags, and forces the packet type to *point-to-point* (so that bad packets are not broadcast through the fabric).
• **Packet Buffer and Cut-Through Path** — The packet buffer shift registers (bSR, one per switch input) are 80 stage 5-bit static shift registers. Based on the availability of requested outputs, a packet is either shifted into a packet buffer, or cut through directly to an output port. A 5-bit 2-to-1 multiplexer selects either buffer or cut-through path to the output routing logic.

• **Output Routing Logic** — Each output routing block (one per output) consists of a PLA to modify the FAN of broadcast packets ([$\pm2$] or [$\pm2$] +1), a single stage 5-bit shift register to delay packet words for test packet header manipulation, and two 5-bit 4-to-1 multiplexors to select the appropriate path to the outputs (undelayed, delayed, delayed [$\pm2$], or delayed [$\pm2$] +1). The output of each multiplexer passes through a 5-bit 2-input NAND gate, which is enabled by the output enable decision logic to route the packet to the appropriate output(s). The outputs from one of the NAND gates are merged with the corresponding outputs from the other output routing logic block (only one of which has been enabled) using a 5-bit 2-input (negative logic) NOR gate. This data is latched by a 5-bit register, which drives the PSE downstream data outputs.

• **Enable and Grant Decision Logic** — A single PLA-based output decision logic block determines how the output logic routing will enable packets to PSE output ports, based on the requests generated by the packet header logic, and by the downstream grant lines (indicating whether packets may flow through each output). Upstream grant lines are also set, based on whether each input port will be able to accept new packets.

• **Switch Element Control** — Switch element control blocks (one per switch input) are PLA-based finite state machines which operate essentially in lock-step, to control operation of each switch half.
4. Design of major functional blocks

This section provides a more detailed description of each of the functional blocks described previously.

4.1. Input Shift Register

The input shift register logic is shown in Figure 5.

The interface signals to the input shift register are:

- **Inputs**
  - *Upstream Data* (ud[p3210]) — The 5-bit word input from the ud inputs to the PSE.
  - *Clock* (Phi1, Phi1', Phi2, Phi2') — Shift register clocking.

- **Outputs**
  - *Data* (t6d[p3210]) — The 5-bit output from the shift register; input data delayed by 6 clock cycles.
  - *Header Data* (hd[p3210]) — The 5-bit output from the first stage of the shift register.
  - *Parity Check* (par_odd, par_even) — Outputs indicating the parity of data on the hd lines.

The input shift register SHRIN consists of three major components: a single stage 5-bit shift register, a parity check circuit, and a five stage 5-bit shift register.

The single stage shift register SHR5K1 shifts 5-bit packet data in from the upstream data (ud) inputs of the PSE, allowing for synchronization with the beginning of a packet cycle. The output of this stage is passed to the header logic block.
The parity check circuit PARCHK is logically an EXCLUSIVE-OR tree to determine the parity of the incoming data. Outputs (parvn, parodd) are passed to the control logic to take appropriate action in case of a parity error. Parity is not restored if in error, but an error will cause the error output (err) of the PSE to go high.

The remaining five stages of shift SHR5X5 allow the header logic circuit and output enable logic sufficient time to make control decisions that affect routing of the packet, before the packet proceeds too far into the PSE data path. These decisions include whether the packet is to be buffered or cut through, if any header modification is to occur, and to which output port the packet is destined.

4.2. Packet Header Logic

The packet header logic is shown in Figure 6.

The interface signals to the header logic are:

- **Inputs**
  - *Header Data* \((\text{hd}[3:10])\) – The 4-bit word from the output of the first stage of the input shift register, which is selectively loaded into the appropriate header registers.
- **Header Register Load (1drc, 1dfa, 1dbh, 1dbl)** – The load control lines to load the header RC, FA, BH, and BL into the corresponding header registers.

- **Operation Mode (om[10] [10]')** – The switch network operation mode within the switch fabric.

- **Stage Number (sn[10] [10]')** – The stage number of which this PSE is a part.

- **Header Register Select (hs[10] [10]')** – Selection for header register to be examined on hr outputs.

- **Routing Control Parity Error (rcpar)** – Parity of RC was invalid.

- **Outputs**

  - **Output Port Request (r[110]')** – Switch element output port(s) requested: 000=(no port), 100=(either port), 101=(port 0), 110=(port 1), 111=(both ports).


  - **Header Register Data (hr[3210])** – Contents of the header register selected by hs.

  - **Broadcast Channel Low-order Bit (bl[0])** – Low order bit of the BL.

The header logic block HEADER consists of three major components: the header registers, the header request logic, and the header register multiplexor.

The header registers HDRREGS are four 4-bit registers (parity is not needed), which store the contents of the four header words RC, FA, BH, and BL. The four registers share a common input data bus, which comes from the input shift register SHRIN. The load inputs are successively clocked by the control state machine CONTROL, as the header is shifted through SHRIN.

The header request logic HDRREQ examines the relevant contents of the header registers and determines what output ports the packet requires. The circuit HDRRC examines the RC and determines the packet type (nil, point, bcast, test), as well as indicating if the RC is invalid (rcbad), and determining the high order request bit (rH), i.e. if any request is to be made. The circuit HDRFA examines the FA. The specific output port requested (rqeq0)1 is determined by examining the SNth bit of the ADR in the case of a point-to-point packet. The relation of the FAN to the SN (eq2sn, 1t2sn) is determined for broadcast packets (FAN = 2SN, FAN < 2SN). The HDRCOM circuit decodes the OM of the switch element. The HDRPORT circuit uses the outputs of HDRRC, HDRFA, and HDRCOM to determine the specific port request bits r[10].

The header multiplexor HDRMUX is a 4-bit 4-to-1 multiplexer with inputs from HDRREGS to select the specific register whose contents will appear on the output hr for diagnostic and test use.

### 4.3. Packet Buffer and Cut-Through Path

The packet buffer and cut-through path logic is shown in Figure 7.

The interface signals to the packet buffer and cut-through path are:

- **Inputs**

  - **Data In (t8d[p3210])** – The 5-bit input from SHRIN.

  - **Packet Data Path Select (spath)** – Buffered/cut-through output path select.

  - **Clock (Phi1, Phi2)** – Shift register clocking.

  - **Shift Register Control (start, stop, full, noout)** – Shift register clocking control.
Figure 7: Packet Buffer and Cut-Through Path

- Outputs
  - Data Out (d[p3210]) – The 5-bit word output delayed one clock cycle.
  - Data Undelayed Out (dud[p3210]) – The 5-bit word output bypassing delay.

The packet buffer and cut-through path MAINPATH consists of three major components: a packet buffer shift register, a multiplexer, and a single stage 5-bit shift register.

The packet buffer SHR5X80 is based on an 80 stage 5-bit shift register. The shift register additionally contains very large line drivers (CLKBUFF) to drive the shift register clock lines, and control logic (CLKKILL) to selectively disable shifting, to allow the buffer to hold a packet.

The multiplexer is a 5-bit 2-to-1 multiplexer which selects either the packet buffer or cut-through path to go to an output port.

The single stage shift register serves to delay the packet data by one clock cycle, and is the normal path through which packet data is output (d). The bypass path (dud) allows a word to effectively pass another word when the shift register is disabled. This is used to rotate the header of a test packet when rrf is asserted to the RSE.

4.4. Output Routing Logic

The output routing logic is shown in Figure 8.

The interface signals to the output routing logic are:

- Inputs
  - Data Input (d[p3210]) – Packet data proceeding to one or both output ports, which has passed through the shift stage at the output of MAINPATH.
- **Data Undelayed Input** (dud\(\{p3210\}\)) – Packet data proceeding to one or both output ports, which has bypassed the the shift stage at the output of MAINPATH.
- **Output Multiplexor Select** (os\([10][10]'\)) – Selection of path through the output multiplexor.
- **Output Enable** (en\([10]\)) – Enables the packet data to the appropriate output port.
- **Output Data from Other Half** (b0\([p3210]\)) – Packet data from the output routing logic of the other half of the PSE.
- **Load Downstream Data Register** (1ddd) – Load to the downstream data register of the packet data that will shift through to the PSE dd outputs.

*Outputs*

- **Downstream Data** (dd\(\{p3210\}\)) – The 5-bit output to the dd outputs of the PSE.
- **Output Data to Other Half** (a1\([p3210]\)) – Packet data to the output routing logic of the other half of the PSE.

The output routing logic block OUTPUT consists of three major components: the header modification circuit, the output multiplexor, and the output data path director.

The header modification circuit OUTHDR performs modification of FAN for replicated broadcast packets. It consists of a PLA which increments and (integer) divides FAN by two (maintaining parity good or bad), and an EXCLUSIVE-OR gate to maintain parity for a simple divide by two. Input to this circuit comes from the (normal) delayed data input (d) from MAINPATH.

The output multiplexor OUTMUX selects the appropriate data path to proceed to switch port outputs, based on the output select (os) from the control logic. The paths to be selected are:

- delayed data from MAINPATH, the normal path for packet data
- data divided by two for FAN of replicated packets
- data incremented and divided by two for FAN of replicated packets
• undelayed data from MAINPATH, used only during test packet header rotation, controlled by the rrf PSE inputs

Two 5-bit 4-to-1 multiplexors are used, connected such that during FAN modification of replicated packets, one output will receive the packet with FAN divided by two, and the other output the packet with FAN incremented and divided by two. The BCN of the header controls this. In particular, if BCN is odd, output port 0 will receive the packet with FAN simply divided by two, if BCN is even it will be port 1. This allows the determination of which copy of a packet will reach a particular output port, which is required by the BTC to perform the correct channel translation on each copy.

The output path director logic OUTFIR selectively passes data out, based on the enable (en) inputs. Input from one multiplexor in OUTFIR is passed, if enabled, through to the downstream data register within this OUTPUT block. Input from the other multiplexor is output (a1) to the other half of the switch, destined for the corresponding output. Packet data from this switch half is combined (oRed) with packet data from the other switch half (b0) destined for this output. Only one of the OUTPUTs can have a path enabled to a given output port, so the enabling and combination of the data path behaves as a 2 x 2 crosspoint switch. The output is latched in a final register stage (REGS), to be shifted out to the downstream data leads (dd) of the PSE.

4.5. Enable and Grant Logic

The output enable and upstream grant generation logic is shown in Figure 9.

The interface signals to the enable and grant logic are:

• Inputs
  - Output Port Requests (r[ab][N10]) – The 3-bit request vectors from the header logic of each switch side (ra, rb). The interpretation of the vector is: 00=(no port), 10=(one port), 11=(both ports).
  - Downstream Grants (dg[10]) – Grant signals from downstream neighbors, indicating that permission has been given to transmit a packet on the corresponding dd port during the next packet cycle.
  - Grant-Time (gt) – High when valid grants are present on the dg inputs.
  - Latch Downstream Grants (1dg) – A control signal indicating that the dg inputs should be latched for the current grant cycle.
  - Post-cycle (pc[12]) – Timing control signals for the enable logic. The first half computation state is saved by asserting pc1, favored input/output toggles are saved by asserting pc2 in preparation for the next grant cycle.

• Outputs
  - Output Enables (e[ab][10]) – Switch element output port enable signals to each output side of the switch (ea, eb). The interpretation of the vector is: 00=(no port), 01=(port 0), 10=(port 1), 11=(both ports).
  - Upstream Grains (ug[ab]) – Grant signals to upstream neighbors, granting permission to transmit a packet to the corresponding ud port during the next packet cycle.
The output enable and grant generate logic block ENABLE consists of a PLA to generate the outputs, along with random logic to latch inputs and outputs, and to perform timing functions.

The PLA ENGEN uses the request vectors \( r[ab] \) and additional state vectors to generate the output enables \( e[ab] \).

One of the additional state vectors is the last-used toggle \( u[10] \). This is a two bit code indicating the most recently used switch input \( I=0:a, I=1:b \) and output \( 0=0, 0=1 \). These allow the routing policies of arbitrary input tie breaking, and uniform output distribution (under the proper circumstances) to be followed. During each grant cycle, new values are generated \( (newu[10]) \) based on the inputs and outputs used.

The other state vector is output port availability \( (p[10]) \). This specifies which output ports are available, and is used to allow packets to be routed only to outputs that can accept them. It is generated by PGEND based on downstream grants (latched by ldg), and the last enable vectors.

The upstream grants \( ug[ab] \) are generated by UGGEN based on which inputs have an empty packet buffer, or will be able to transmit a packet from a full buffer during the next packet cycle. This allows a new packet to shift into the packet buffer immediately behind a leaving packet.

The enable vectors are calculated twice during the grant cycle. The first calculation occurs during the propagation of grants through the switch fabric, and is applicable to packets waiting
in the packet buffers. This is based on the value of the port availability vector p. The second calculation occurs during the beginning of the next packet cycle, and applies to packets entering the switch element. This allows an entering packet to either proceed into the buffer, or cut through directly to an output port.

The two enable calculations are performed by ENGEN, with the second set of results ored with the first and stored by ENLATCHOR.

The full specification for enable and grant generation is presented in appendix A.

4.6. Switch Element Control

The switch element control logic is shown in Figure 10.

The interface signals to the control logic logic are:

- **Inputs**
  - Clock (Phi1, Phi2) – State machine clocking.
  - Control Inputs (*ctl-in*) – Control inputs.

- **Outputs**
  - Control Outputs (*ctl-out*) – Control outputs.

The switch element control consists of a finite state machine CONTROL, with one per switch side operating in lock-step (with the exception of some conditional control outputs).

The state machine logic is implemented as a PLA (STATEPLA). Input and output state is held in a set of latches (STATELAT). The control signals are described in the sections relevant to each signal, and will not be enumerated here.

STATELAT consists of a set of input latches (FFDX2B), output latches (GATE, GATENOCLK), and clock control circuitry.
Figure 11: Switch Element Timing

The input latches are loaded on $\phi_1$. The output latches are loaded during the interval including $\phi_1$ through the next $\phi_2$. This signal ($\phi_3$) is generated by the clock control circuitry, and provides 75ns to propagate through the state machine and load, rather than the 25ns available from a single clock phase. The synchronization of output control signals is achieved by selectively gating the latched signals to the outputs. GATE contains gates to enable the outputs based on the external connection of clock signals. By the appropriate placement of contacts, signals are made available on any desired clock phase. Signals that should be enabled during the entire clock cycle are provided by GATENOCLK.

The PSE is driven by a 2-phase, non-overlapping clock, with a major cycle of 100ns, and duty cycle of 25% per phase. A high level timing diagram is shown in Figure 11.
5. Implementation

This section discusses the implementation of the PSE as a 3\(\mu\)m CMOS chip.

5.1. Integrated circuit design

The PSE chip was designed using a set of CAD tools from the University of California at Berkeley, described in [ScMa86]. The circuit layout tool is used is MAGIC, which is a graphical layout editor providing a hierarchical cell structure. Technology based design rules are checked by magic, and are based on a manhattan Mead-Conway design style with \(\lambda=1.5\mu\)m [MeCo80]. This results in a minimum line width of 3\(\mu\)m. Chip fabrication technology is a bulk p-well CMOS process, with facilities provided by MOSIS [USCA]. Various references used during the design of the PSE are listed in the bibliography.

5.2. Circuit design verification

The PSE functional design was executed in a top-down hierarchical manner, but the chip was implemented bottom-up. As each cell was designed, it was simulated before incorporation into higher level cells. The various tools which were available for PSE simulation will be briefly described.

Circuit-level simulation was performed using SPICE [VIZh]. Due to the computational intensity of these simulations, only low level components were simulated in this manner, particularly the basic flip-flops, shift register cells, and clock line drivers.

Faster (and somewhat less accurate) circuit simulation has been performed using FACTS [MCNC], but this tool did not become available to this project until much of the cell design had been completed. FACTS was used to simulate multiple stage shift registers.

Timing analysis (other than results from SPICE and FACTS runs) was performed by CRYSTAL [Ou85], which provides worst-case delay analysis.

Logic-level simulation was executed using ESIM [ScMa86], which is efficient enough to allow simulation of all PSE logic functions, to the top level. Currently all major functional blocks (SHRN, HEADER, MAINPATH, OUTPUT, ENABLE, CONTROL), have been simulated. Simulation of the entire chip was deferred until the individual chips (described in the next section) have been tested.

VESIM, a front-end interface to ESIM, was written which allows test vector input and simulation results to be printed in a vertical format. This enables test vectors considerably longer than 80 to be easily specified, and results interpreted, since the width of a terminal or print line is not a limit.

5.3. Testability features

Since this is the first version of the PSE designed, and the first use of CMOS within the project, testability, particularly with respect to verification of simulation results, is extremely important. It was decided to break the single PSE chip into a three chip set to allow for maximum testability; this is described in the next section.

Testability of the entire PSE chip is enhanced by access to the data path at critical points, to allow errors to be isolated within the functional block level. A full set of registers is present to store the packet header (only the first two words plus one bit are actually needed for PSE function), along with multiplexors and outputs to test pins. This allows the headers to be examined for packets residing in each switch side, and may be useful for switch fabric diagnostics.
The control logic is able to detect and report framing errors, e.g., the occurrence of a packet-time signal at the wrong time, or the incorrect occurrence of a grant-time signal. Parity errors in data input to the upstream-data lines are also reported.

6. Preliminary Chip Fabrication

Due to the complexity of the PSE logic, and the inexperience with the MOSIS CMOS fabrication process, it was decided to split the PSE into a preliminary chip-set for the first fabrication run. The result is a three chip set: PSE1, PSE2, PSE3. PSE1 consists of the header logic (HEADER), and the output logic (OUTPUT), and contains 1555 transistors. PSE2 consists of the input shift register (SHRIN) and the packet buffer and cut-through path (MAINPATH), and contains 8329 transistors. PSE3 contains the enable and grant logic (ENABLE), the switch element control (CONTROL), and the two control counters (CNTR7, CNTR3).

In every case, each logic block is isolated from the others (including power), providing for 8 “logical” chips: HEADER, OUTPUT, SHRIN, MAINPATH, ENABLE, CONTROL, CNTR7, and CNTR3.

The chip frames were designed to conform to MOSIS 64-pin standard frames. The I/O pads used were designed at MIT and obtained from MOSIS[USCh]. The complete PSE1 and PSE2 photomicrographs are shown in Figures 12 and 13, respectively. Due to design changes in the switch element control (which will be discussed in the next section), PSE3 will not be fabricated.
Figure 12: Packet Switch Element Chip 1
Figure 13: Packet Switch Element Chip 2
7. Further Work

7.1. Integrated circuit test

The first two chips of the PSE preliminary chip-set have been fabricated, and need to be tested. A Tektronics DAS 9100 digital analysis system will be used to test the chips. Test vectors that were generated for the ESIM simulations can be downloaded to the DAS for chip verification.

7.2. Packet Switch Element enhancements

The design of the next version of the PSE has begun, with several functional and speed related enhancements.

The new PSE will have 8-bit data paths, instead of 4-bit paths, and each side of the switch element will have two packet buffers. The buffer shift register cells will employ a smaller quasi-static design. The queueing discipline is FIFO with bypass, i.e. the first packet arrived will have its output port request satisfied if possible, otherwise the second packet may bypass and have its request satisfied.

The grant propagation will be simplified so that grants only depend on the availability of an empty packet buffer, and not on whether a full packet buffer will be able to transmit its packet during the next packet cycle. This reduces the buffer utilization, and is the motivation for including the second packet buffer in each switch side. This also will result in a simplified enable logic circuit, reducing the delay through the the associated PLA.

The control and enable logic will be generalized to handle an arbitrary number of packet buffers in each switch side, allowing expansion in the future, if desired. The control logic will be implemented as a fast counter with decoders, allowing considerably faster operation than possible with a PLA-based state machine. An automatic control and timing circuit generator tool has been developed for this purpose.
8. Acknowledgements

The PSE design began as a project for an advanced level integrated circuit design class, EE 563. The class was taught by Fred Rosenberger, whose guidance during the early design, and availability to answer numerous questions until the end, is greatly appreciated. The initial specification for the PSE, as well as the motivation for involvement in the project were provided by Jon Turner.

The other participants in the class project were David Gesswein and Ken Lorber. David designed and simulated the shift register and flip-flop cells, the packet buffer (including clock drivers), and counters. He also designed the control state machine and associated registers. Ken designed and simulated the original enable and grant generation circuit, as well as the output header modification circuit. He also wrote VESIM, the vertical front end interface to ESIM.

Bill Thomas and Roger Chamberlain provided insight and advice at various points in the design cycle, and George Robbert put up with numerous questions involving the details of plain \TeX, when \LaTeX wouldn't do what I needed.

Any errors or omissions, are solely the responsibility of the author.
References


[St88a] Sterbenz, James P. G., Design of a VLSI Packet Switch Element, Washington University Computer Science Department, WUCS-88-5, St. Louis, April 1988

[St88b] Sterbenz, James P. G., Design and Implementation of a VLSI Packet Switch Element, Washington University Computer and Communications Research Center, WUCRC-88-1, St. Louis, April 1988


[usca]  *MOSIS User's Manual*, University of Southern California Information Sciences Institute, Marina Del Rey, Calif., (no date)

[uscb]  *MIT-SCPSU-PADS.CIF*, University of Southern California Information Sciences Institute, Marina Del Rey, Calif., (no date)

