# BTC Test Methodology

Gaurav Garg

This paper describes the testing methodology for the Broadcast Translation Circuit. It covers logic simulation, timing simulation, and a single push-button test for the device. The document is intended to be a supplement to documents WUCS-89-52 and WUCS-90-19, which outline the design of the BTC in full details. The intent of a push-button test is to have a single test that causes the device to process every type of packets that it could receive. This would include various combination of RC and OP fields, a test that examines every table entry thoroughly, as well as a test of the... Read complete abstract on page 2.

# BTC Test Methodology

Gaurav Garg

Complete Abstract:

This paper describes the testing methodology for the Broadcast Translation Circuit. It covers logic simulation, timing simulation, and a single push-button test for the device. The document is intended to be a supplement to documents WUCS-89-52 and WUCS-90-19, which outline the design of the BTC in full details. The intent of a push-button test is to have a single test that causes the device to process every type of packets that it could receive. This would include various combination of RC and OP fields, a test that examines every table entry thoroughly, as well as a test of the parity error detection and the signaling mechanism.

# BTC TEST METHODOLOGY

Gaurav Garg

WUCS-90-20

May 1990

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

## Abstract

This paper describes the testing methodology for the Broadcast Translation Circuit. it covers logic simulation, timing simulation, and a single push-button test for the device. The document is intended to be a supplement to documents WUCS-89-52 and WUCS-90-19, which outline the design of the BTC in full detail.

The intent of a push-button test is to have a single test that causes the device to process every type of packet that it could receive. This would include various combinations of *RC* and *OP* fields, a test that examines very table entry thoroughly, as well as a test of the parity error detection and signaling mechanism.

# Table of Contents

# List of Figures

# BTC Test Methodology

## 1. Introduction

This paper describes the testing methodology for the BTC. This includes logic simulation using ESIM, timing simulation using CAZM, as well as device testing. A checklist to be used before the chip description is sent to the fabricator is also provided here. This document is intended to be a supplement to documents WUCS-89-52 and WUCS-90-19, which outline BTC design in detail. Figure 1 provides a block diagram of the circuit as seen in WUCS-89-52.
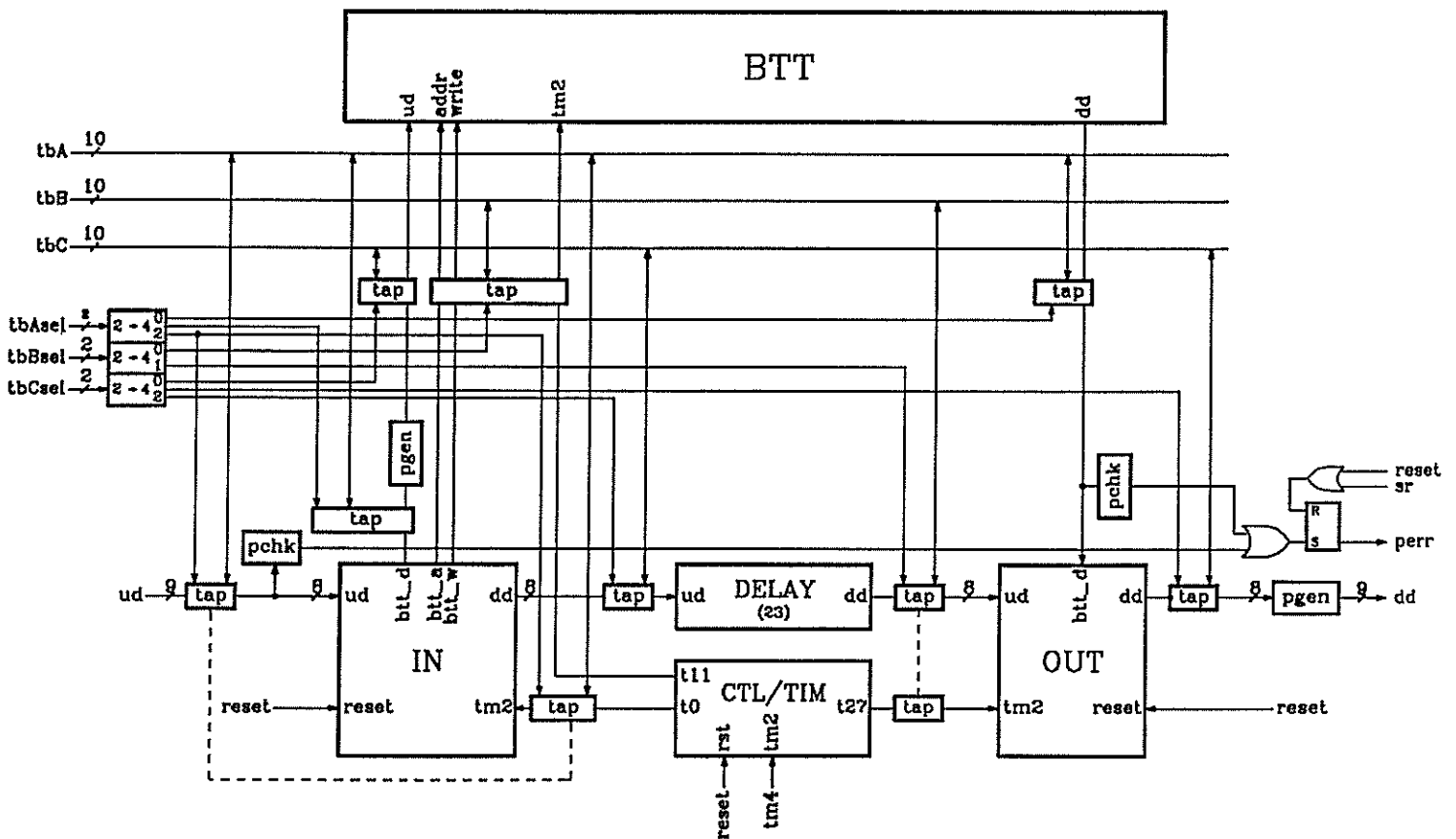


Figure 1. BTC Block Diagram

## 2. Pre-fabrication Checklist

This list is a general list and would apply to most circuits designed using the set of CAD tools currently used by the Advanced Networks Group.

1. Logic Simulation - A complete set of ESIM tests.
2. Timing Simulation - CAZM for components and interconnections.
3. Power Bus Examination - set of ESIM tests with no labels except at pins.
4. Design Rule Check - Standard drc check for the chip.
5. Well Contacts - standard test invoked by "cif see WELLERR" in Magic.
6. Incorrect Well Contacts - /usr5/usr2/da/cad/bin/freds-lab-substrate.
7. Cif Generation - has to be performed to send file to MOSIS.

## 3. BTC Logic Simulation

The intent here is to test the logical function of the circuit without regard to timing. This is done by attempting to toggle every transistor on the device back and forth and by examining the integrity of every interconnection. The data paths are tested with various combinations of walking and running 1's and 0's along with all binary combinations for 8 bits. All the simulations were performed on a test deck constituting the entire BTC including the pads. The set of tests that were performed fall into the categories that are listed below, and they are examined in detail later in this section.

- Integrity of parity error detection mechanism.
- All combinations of *RC* and *OP* fields except those affecting BTT.
- All possible combinations of test access points and buses.
- BTT Testing - write to and read from every location twice.

### 3.1. ESIM Limitation Effects

ESIM has a limitation on the number of input vectors it can handle in one run. This limit is approximately 3000. This implies that with a four vector per clock tick configuration and 86 clock ticks per packet, one may simulate upto 8 packet cycles in a single ESIM run. It is not possible to string various input files together, and thus one must do the entire startup procedure including a hardware reset at the beginning of each run.

### 3.2. Parity Error Detection Mechanism

The circuit has two parity error detectors, one at the input and the other at the output of the BTT. The error on the input is forced by providing data with an incorrect parity to the circuit, while the error out of the table is forced by reading from entries in the BTT before writing to them. The parity error signal is reset via a soft reset signal a few clock cycles after being asserted high due to the error.

### 3.3. RC and OP Combinations

The test consists of trying all relevant combinations of RC and OP fields. We omit tests with multipoint packets and read/write BTT functions from this because these operations are addressed during tests of the BTT. This aspect of testing is organised into three tests to satisfy the constraint of 8 packets to a test deck. The first involves setting

the RC field to idle and trying all 8 combinations of the OP field including read and write BTT, which are rendered invalid because the RC field is idle. The second sets the RC field to point-to-point and tries all combinations of the OP field except the read and write BTT functions. Similarily the third sets the RC field to specific path and performs all OP field combinations except the read and write BTT functions.

### 3.4. Tap Combinations

Test bus A multiplexes between 3 taps, test bus B between 2 taps, and test bus C between 3 taps. This test does not examine a path or function critical to the performance of the circuit. Also, it is not advisable to integrate this test with all the others at any time because the selections of taps should not be changed during a single run. Thus, one takes a input deck with 6 packets. This includes an idle packet, a write BTT packet, a read BTT packet, a point-to-point packet, a multipoint packet, and a specific path packet. This input deck is used in three tests each connecting a different set of taps to the test buses. Recall that BTC2 is set up so that it is impossible to write to internal buses from the test pins. This eliminates the set of tests that was used in BTC1 to examine such functionality.

### 3.5. BTT Testing

The 6-T SRAM can be simulated in ESIM only by attaching ground labels to the inputs to the pull-up transistors and disconnecting them from the pull-down transistors. This implies that the logic simulation is performed on memory cells that are substitutes for the real cells. Thus the logic simulation is used to test the interconnections in the BTT, not the memory itself. The memory is simulated during timing simulation. The integrity of the memory is examined during device testing, and the output patterns from the logic simulation may be used to develop such a test.

Each of the 64 entries in the BTT was tested on a separate input deck. Table 1 summarizes the input deck, which consisted of 8 packets. This test writes to the BTT three times, and reads from it four times, each under different conditions. It is important that one is able to read from a table entry twice in succession. It is also convenient to have the ability to write to the BTT with a multipoint packet because this implies that one may write the same entry to tables in a number of BTC's at the same time.

### 4. BTC Timing Simulation

This is not a functional test. It is supposed to test drive capabilities, frequency of operation, and other issues that are not covered by logic simulation. However one does not need to test every possibility as with logic simulation. For example, a satisfactory test would consist of having satisfactory rise and fall times on all major data buses. Unfortunately *CAZM* can only process about 20,000 transistors at a time. This implies that it is usually not possible to simulate an entire IC, rather each test simulates part of the circuit.

The major components of the BTC are the global CTL/TIM the INC, the OUTC, the delay line, and the BTT. All of the CAZM simulations were performed at an operational speed of $42MHz$. The clocks had rise and fall times of $1ns$, and the data lines had rise and fall times of $3.5ns$. The process parameters from the first fabrication run for the BTC were used as the process parameters for the simulation. These process parameters

Table 1. Logical Test for a BTT Entry

| Sequence Number | Type of Packet | Routing Control | Operation Field |
|---|---|---|---|
| 1 | Idle, Vanilla | Idle | Vanilla |
| 2 | Write BTT | pt-to-pt | Write BTT |
| 3 | Multipoint | multipoint | Vanilla |
| 4 | Write BTT with Multipoint | multipoint | Write BTT |
| 5 | Read BTT | pt-to-pt | Read BTT |
| 6 | Read BTT with Multipoint | multipoint | Read BTT |
| 7 | Write BTT with Specific path | Specific Path | Write BTT |
| 8 | Read BTT with Specific path | Specific Path | Read BTT |

are provided in Figure 2.

The first fabrication run of the BTC indicated that the point-to-point path for the circuit functioned correctly at upto $28MHz$. Thus this was not studied very carefully for the second run. The focus was on the the BTT, its memory cells, and its interaction with neighboring components. This led to a set of tests that are described in this section.

---

```
/* N-MOS Description */
model cmosn nmos level=2.00000 ld=0.250000u tox=395.000e-10
+nsub=5.706000e+15 vto=0.744570 kp=5.987000e-05 gamma=0.497800
+phi=0.600000 uo=684.8 uexp=0.185773 ucrit=40591.3
+delta=1.44455 vmax=73636.6 xj=0.150000u lambda=3.557268e-02
+nfs=4.667514e+12 neff=1.0 nss=1.000000e+12 tpg=1.00000
+rsh=28.94 cgso=3.2783e-10 cgdo=3.2783e-10 cgbo=7.168240e-10 cj=1.2010e-4
+mj=0.6756 cjsw=5.545e-10 mjsw=0.2636 pb=0.8


/* P-MOS Description */
model cmosp pmos level=2.00000 ld=0.250000u tox=395.000e-10
+nsub=6.078000e+15 vto=-0.723651 kp=2.404000e-05 gamma=0.513800
+phi=0.600000 uo=274.99 uexp=0.300442 ucrit=35276.9
+delta=1.00000e-06 vmax=94066.1 xj=0.050000u lambda=5.530815e-02
+nfs=2.697453e+12 neff=1.001 nss=1.000000e+12 tpg=-1.00000
+rsh=109.4 cgso=3.2783e-10 cgdo=3.2783e-10 cgbo=5.553545e-10 cj=2.357e-4
+mj=0.5589 cjsw=2.785e-10 mjsw=0.3161 pb=0.8
```

Figure 2. BTC Process Parameters

---

## 4.1. INC to BTT Interface

The input deck for this circuit includes the CTL/TIM, the INC, and part of the BTT, along with appropriate taps and connections to the pad frame. All input signals are provided at the pads. The BTT is complete except that it has no memory elements. The focus of the study is on the *mil* and *lat* signals in the BTT, whether they overlap with the succeeding $\phi2$ or not, and on the address and data being latched by the BTT correctly. Figure 3 shows this input deck.

Figure 4 shows the output. Notice that it shows a few select signals from those that were watched. Both *mil* and *lat* signals have rise and fall times that are approximately

---

```
/* Input Deck for INC to BTT, includes pads, runs at 42 MHz */

wave phi1_x bit {000001111000} pw=2.0n off=0.0 on=5.0 rt=1.0n ft=1.0n
wave phi2_x bit {110000000000} pw=2.0n off=0.0 on=5.0 rt=1.0n ft=1.0n
wave phi3_x bit {000001111000} pw=2.0n off=0.0 on=5.0 rt=1.0n ft=1.0n

wave tbas1_x bit {0000000000000000000000000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave tbas0_x bit {0000000000000000000000000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave tbbs1_x bit {0000000000000000000000000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave tbbs0_x bit {0000000000000000000000000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave tbcs1_x bit {0000000000000000000000000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave tbcs0_x bit {0000000000000000000000000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n

wave hrst_x bit {100000000000000000000000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave srst_x bit {001000000000000000000000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave tm4_x  bit {000010000000000000000000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n

wave ud8_x bit {0000000010000000010101000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave ud7_x bit {0000000000000000010100000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave ud6_x bit {0000000000000000101010000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave ud5_x bit {0000000010000000010100000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave ud4_x bit {0000000000000000101010000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave ud3_x bit {0000000000000000010100000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave ud2_x bit {0000000010000000101010000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave ud1_x bit {0000000000000000010100000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave ud0_x bit {0000000000000000101010000000000000} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n

volt vdd vss 5.0
options reltol=0.001

plot {phi1 phi2 ph1 ph2 ph1' ph2' ph3 rdwrb pt cou6 cou5 cou4 cou3 cou2 cou1
      cou0 newcontrol3_0/mil newcontrol3_0/lat mil lat ld in8 in7 cdrv2_28/bit
      cdrv2_24/bit a5 a4 a3 a2 a1 a0 inadd5 inadd4 inadd3 inadd2 inadd1 inadd0}

transient 768.0n 1.0n powerup
```

Figure 3. Input Deck - INC to BTT
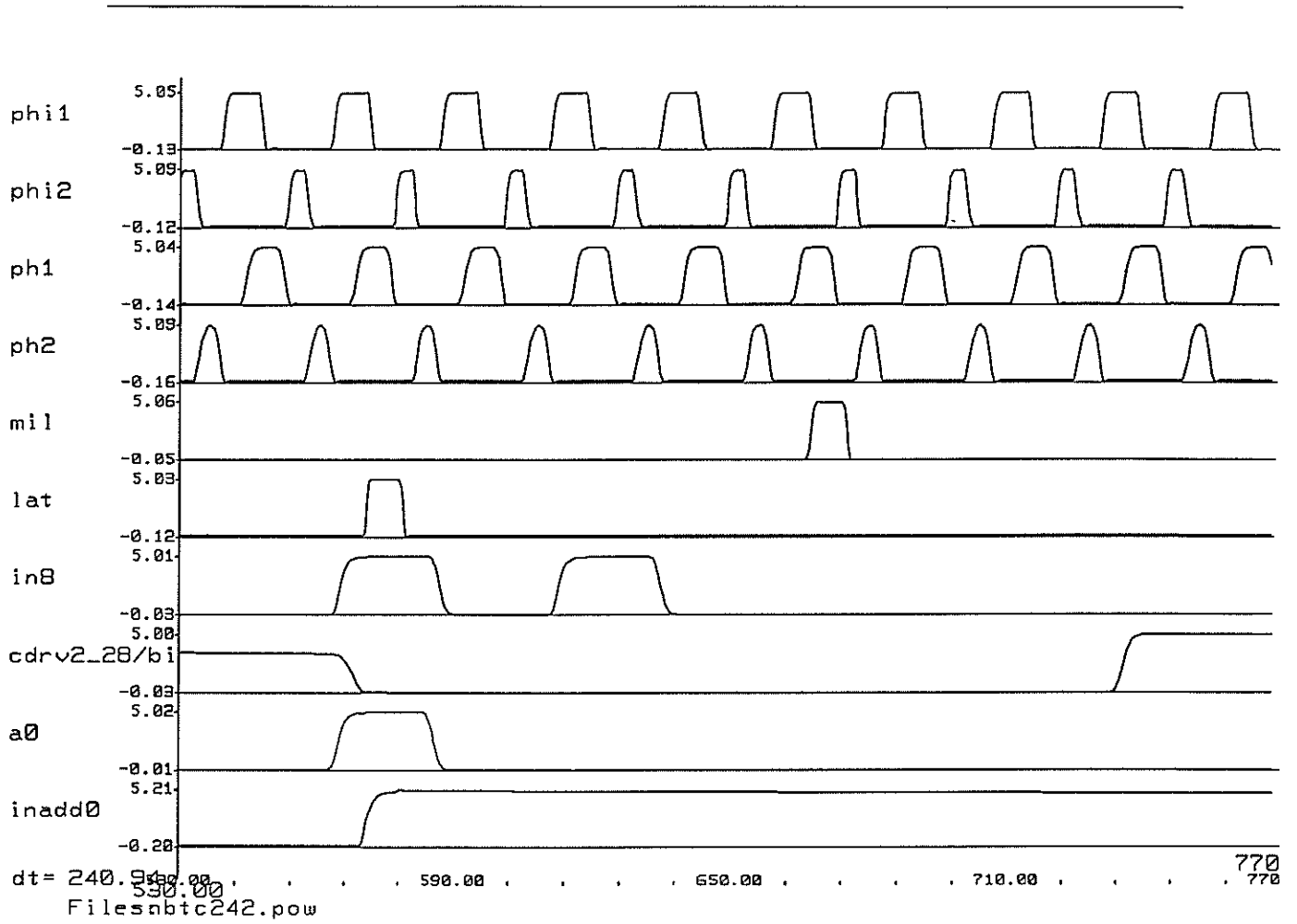
---

Figure 4. INC to BTT Interface

2*ns*. The data at the output of the column driver going into memory (cdrv2_28/bi..), and the latched address signal (inadd0) have rise and fall times of approximately 3*ns* which is acceptable.

## 4.2. Complete BTT Test

This test was intended to test the BTT during the write cycle. The test was conducted with the BTT isolated from all other components, and with all its memory intact. The important nodes here were the storage nodes (memtemp...) in the memory elements on both sides of the address decoder which runs through the center of the BTT. Figure 5 shows that both of these come up to almost 5.0 volts by the time that the word line (wsig) on the memory is at 3.0 volts.
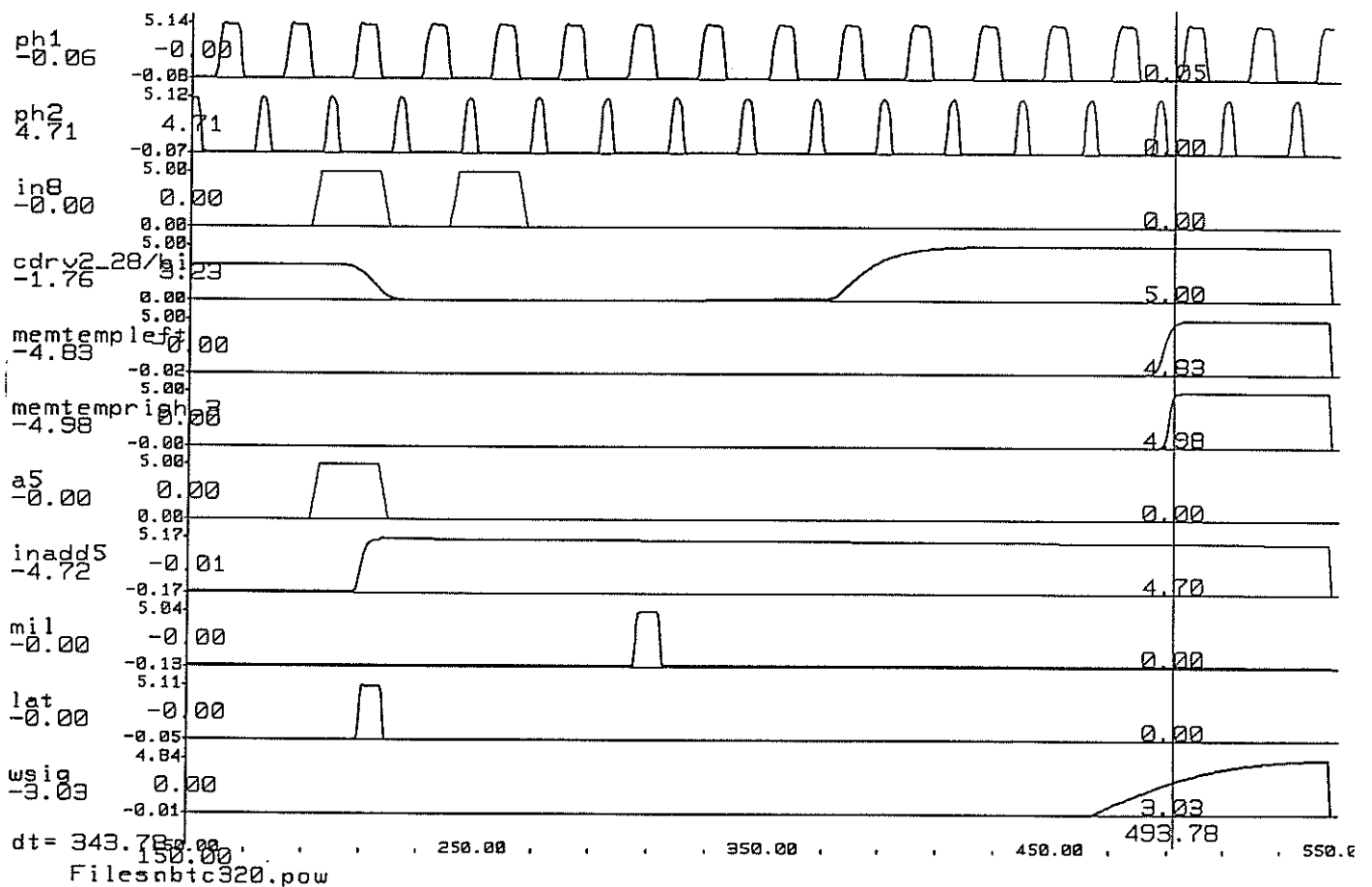


Figure 5. BTT Test

## 4.3. BTT to OUTC Interface

This test checks the BTT to OUTC interface. The test deck includes the BTT without memory elements, the OUTC, and appropriate taps and connections to the pad frame. The inputs include various control mechanisms including *pt* signals to the BTT as well as the OUTC, data into the OUTC with the *RC* field set to pending (see WUCS-89-52) so that it will accept data from the BTT, and data at the downstream side of the BTT set up so that it appears as if it is coming from *bit* lines from the memory. The signals to watch are the signals on the BTT to OUTC interface and on the outputs from the pads. Figure 6 shows the signals from the simulation, and all the relevant signals appear to be acceptable.
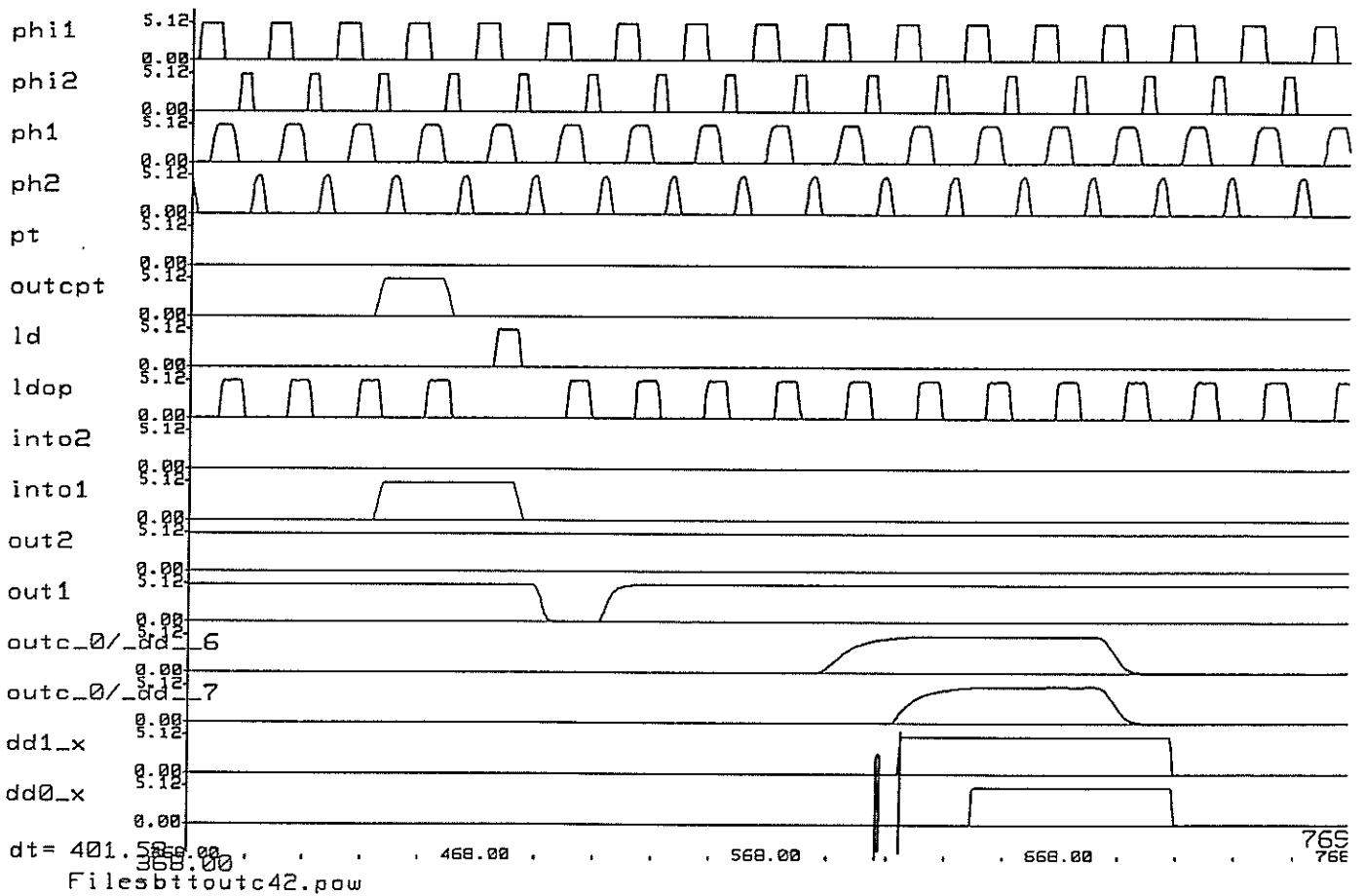


Figure 6. BTT to OUTC Interface

## 4.4. Static RAM - Write and Read

This is a set of two tests, the first involving writing a 0 to memory in the context of the BTT and reading it out, the second the same but with a 1 instead of a 0. The test is set up with a memory elements placed in a row exactly as in the BTT. The column capacitance on the *bit* lines is adjusted by editing the simulation file. All the control signals are provided artificially, but they are pessimistic compared to the real signals in the BTT. The input is provided at the input to the column driver, and is read out after being latched on the downstream side. Figure 7 provides a sample input file for this test.

---

```
/* the first 29 ticks are the write cycle, the read cycle is the last 23 */
/* for write start at 0, for read start is at tick 11 */

wave ph1 bit {000001111000} pw=2.0n off=0.0 on=5.0 rt=1.0n ft=1.0n
wave ph2 bit {110000000000} pw=2.0n off=0.0 on=5.0 rt=1.0n ft=1.0n
wave pre' bit   {11(1) 3(0) 12(1)} pw=48.0n off=0.0 on=5.0 rt=48.0n ft=48.0n
wave dis  bit   {01 11(0)} pw=96.0n off=0.0 on=5.0 rt=48.0n ft=48.0n
wave mil  bit   {103(0) 1111 517(0)} pw=2.0n off=0.0 on=5.0 rt=2.0n ft=2.0n
wave mil' bit   {103(1) 0000 517(1)} pw=2.0n off=0.0 on=5.0 rt=2.0n ft=2.0n
wave eval bit   {11(0) 8(1) 33(0)} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave eval' bit  {11(1) 8(0) 33(1)} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n
wave ld   bit   {415(0) 1111 313(0)} pw=2.0n off=0.0 on=5.0 rt=2.0n ft=2.0n
wave ld'  bit   {415(1) 0000 313(1)} pw=2.0n off=0.0 on=5.0 rt=2.0n ft=2.0n
wave ldop bit   {34(000000011110) 000000000000 26(000000011110)} pw=2.0n off=0.0 on=5.0 rt=2.0n ft=2.0n
wave ldop' bit  {34(111111100001) 111111111111 26(111111100001)} pw=2.0n off=0.0 on=5.0 rt=2.0n ft=2.0n
wave word0 bit  {15(0) 0011 10(0) 0111 19(0)} pw=24.0n off=0.0 on=5.0 rt=10.0n ft=10.0n
wave sysin bit  {10(0) 18(1) 24(0)} pw=24.0n off=0.0 on=5.0 rt=3.5n ft=3.5n

volt vdd vss 5.0
options reltol=0.001

plot {ph1 ph2 pre' dis mil eval ld ldop word0 sysin bit bitb bitstore0 bitstore0b sysout}

transient 960.0n 1.0n powerup
```

Figure 7. Input file to Write/Read from Memory

---

Figure 8 and Figure 9 show sample signals for the test performed with a 0 and 1 written to and read from memory respectively. Notice that in Figure 8, the *bitstore* nodes already had the correct value and we notice a small glitch in the *bitstore* 0 node of about 0.8 volts while reading the value 0 out. In Figure 9, we notice that the *bitstore* nodes change their value from the initial 0 to a 1, and we see a glitch on the *bitstore* 0*b* node while reading the value out. This is due to charge sharing when the word line first comes on.
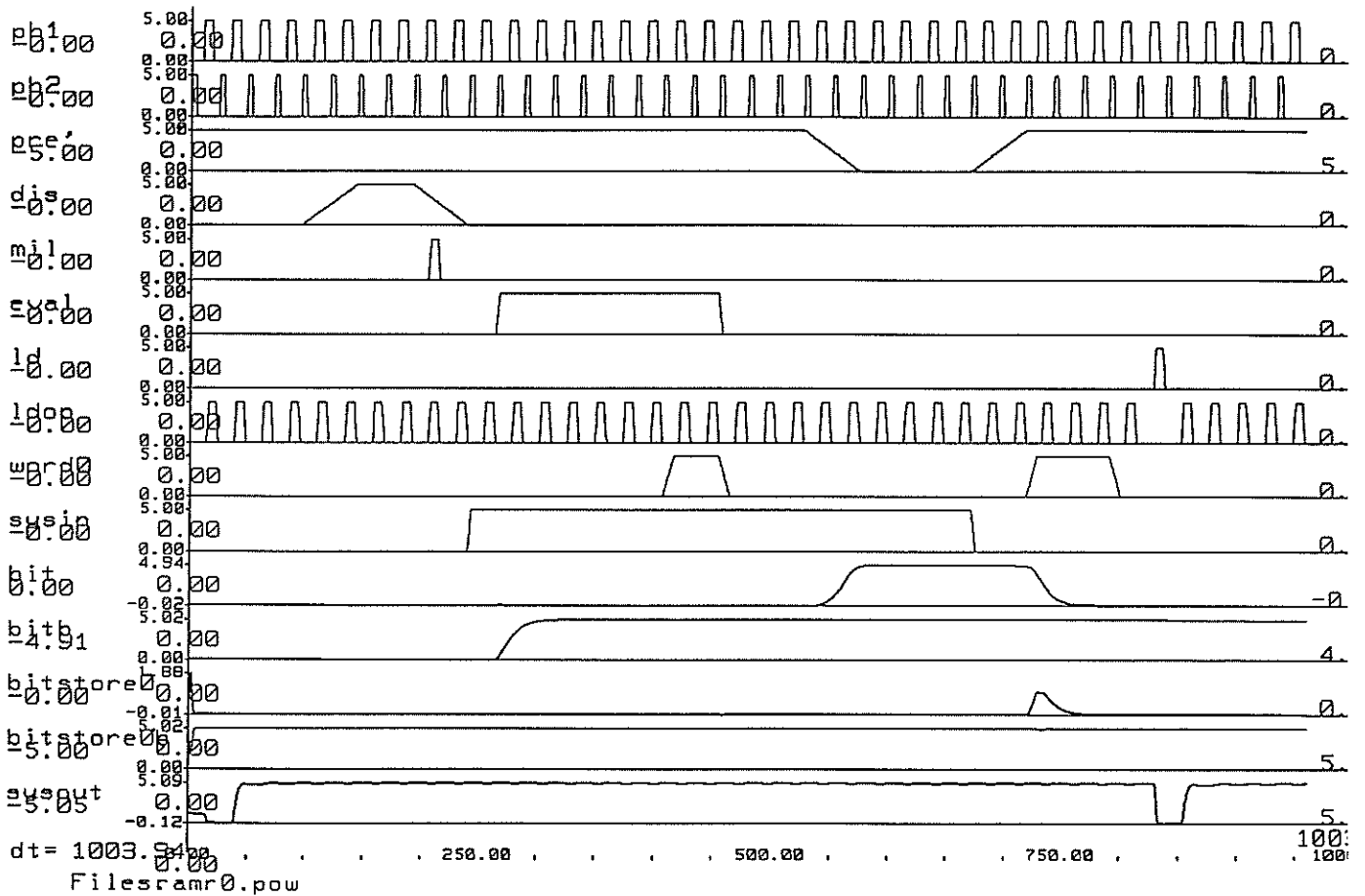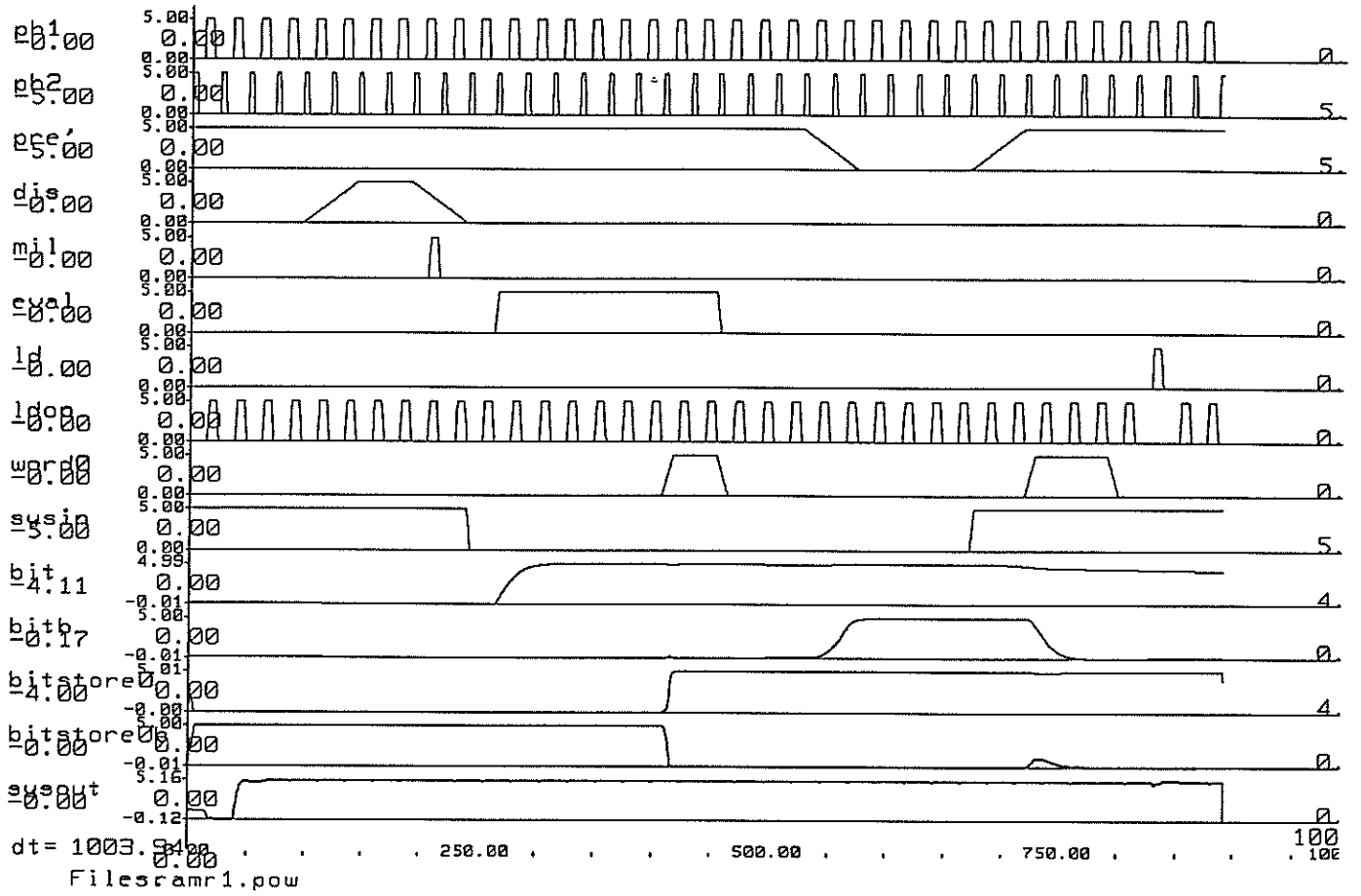


Figure 8. SRAM Write/Read 0

Figure 9. SRAM Write/Read 1

## 5. Device Testing

The objective here is to create a single push-button test that would yield a "Run Passed" or "Run Failed" message on the LV500 indicating whether or not the particular device is acceptable. The pattern section of the test set-up on the LV500 is very close to the output that ESIM generates. Thus, the patterns on the test files are generated by modifying ESIM output files. These are concatenated to create a single test file for the tester.

### 5.1. Test Templates

One may want to watch different acquisition pins at different points during the test. This is done by having different templates for each section of the test and by masking off the acquisition pods for signals that are irrelevant for different parts of the test. There were three templates created for the global push-button test each of which would run at 25MHz. Each template had the tap acquisition pins masked off because this is not critical to the functionality of the circuit. Table 2 summarizes the behavior of the templates.

Table 2. Summary of Test Templates

| Template | Downstream Data | Parity Bit | Tap Outputs |
|---|---|---|---|
| Template 0 | Compare | Masked | Masked |
| Template 2 | Masked | Masked | Masked |
| Template 3 | Compare | Compare | Masked |

Template 2 is used at startup time, Template 3 is used during the parity error detection mechanism test, and Template 0 is used during the bulk of the test. The timing information for a template at a speed of 25MHz appears in Figure 10. Table 3 provides template timing information for various operational frequencies. All timing values are in nanoseconds.

Table 3. Template Timing

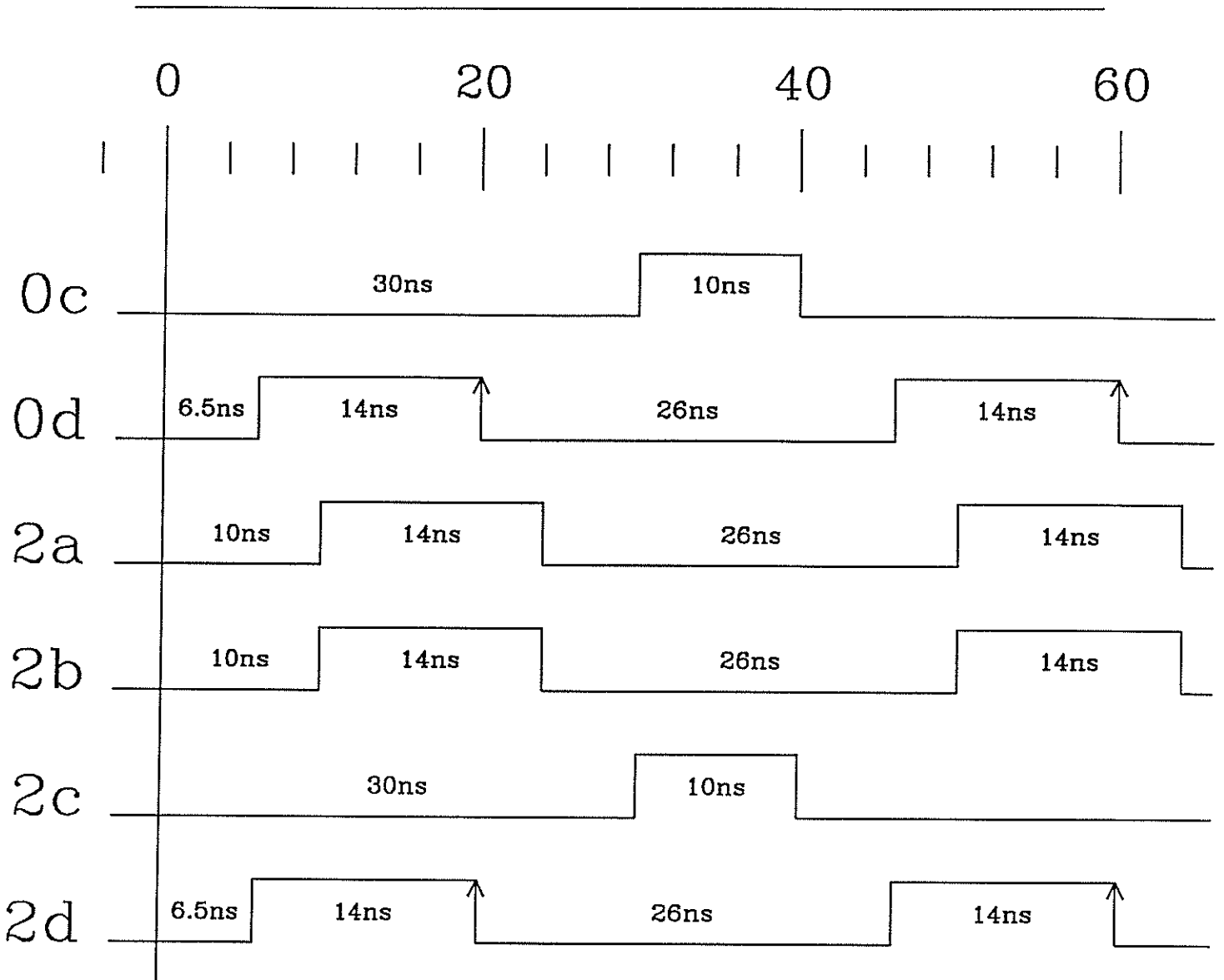| Phase | Clock | T = 96ns | | T = 48ns | | T = 40ns | | T = 32ns | |
|---|---|---|---|---|---|---|---|---|---|
| | | Width | Delay | Width | Delay | Width | Delay | Width | Delay |
| 0c | phi2 | 0.0 | 72.0 | 0.0 | 36.0 | 0.0 | 30.0 | 0.0 | 24.0 |
| 0d | acquisition | 16.0 | 32.0 | 8.0 | 16.0 | 6.5 | 14.0 | 5.5 | 10.0 |
| 2a | phi1 | 24.0 | 32.0 | 12.0 | 16.0 | 10.0 | 14.0 | 8.0 | 10.0 |
| 2b | phi1 | 24.0 | 32.0 | 12.0 | 16.0 | 10.0 | 14.0 | 8.0 | 10.0 |
| 2c | phi2 | 0.0 | 72.0 | 0.0 | 36.0 | 0.0 | 30.0 | 0.0 | 24.0 |
| 2d | acquisition | 16.0 | 32.0 | 8.0 | 16.0 | 6.5 | 14.0 | 5.5 | 10.0 |

Figure 10. Template Timing

## 5.2. Grouping Information

The active pins on the BTC are divided into 11 groups for testing purposes. Table 4 summarizes the grouping information used on the LV500.

Table 4. Summary of Grouping Information

| Group | Signals | Group Type | Clock Phase |
|-------|---------|------------|-------------|
| phi1 | phi1 | force r0 | 2b |
| phi2 | phi2 | force r1 | 2c |
| phi3 | phi3 | force r0 | 2a |
| reset | tm4, hrst, srst | force dnrz_1 | 0c, 2c |
| err | err | compare edge_t | 0d |
| ud | ud0..8 | force dnrz_1 | 2c |
| dd | dd0..8 | compare edge_t | 0d |
| tapsel | tbAs0..1, tbBs0..1, tbCs0..1 | force dnrz_1 | 2c |
| tapA | tbA0..9 | compare edge_t | 2d |
| tapB | tbB0..9 | compare edge_t | 2d |
| tapC | tbC0..9 | compare edge_t | 2d |

## 5.3. Pin & LV500 Location Chart

This section provides a complete list of the pins and their connections to respective sector locations on the LV500.

Table 5. Pin and Sector Location Chart

| Pin Number | Signal Name | Sector | Channel |
|---|---|---|---|
| 2 | phi1 | b | 5 |
| 3 | phi2 | b | 2 |
| 4 | phi3 | b | 6 |
| 5 | tm4 | b | 3 |
| 6 | hrst | 0 | f |
| 7 | srst | 0 | e |
| 8 | err | 0 | d |
| 99..107 | ud8..0 | b | f..7 |
| 38..31 | dd8..1 | 1 | f..8 |
| 30 | dd0 | 1 | 7 |
| 88..87 | tbAs1..0 | a | 6..7 |
| 90..89 | tbBs1..0 | a | 4..5 |
| 92..91 | tbCs1..0 | a | 2..3 |
| 85 | tbA9 | a | 9 |
| 84..83 | tbA8..7 | a | b..c |
| 81..80 | tbA6..5 | 9 | a..9 |
| 79 | tbA4 | a | d |
| 78 | tbA3 | a | a |
| 77..75 | tbA2..0 | 9 | 8..6 |
| 72 | tbB9 | 9 | 3 |
| 71..69 | tbB8..6 | 9 | 0..2 |
| 68..63 | tbB5..0 | 8 | a..5 |
| 61..59 | tbC9..7 | 8 | 3..1 |
| 58 | tbC6 | 2 | b |
| 57 | tbC5 | 8 | 0 |
| 56 | tbC4 | 2 | c |
| 54 | tbC3 | 2 | d |
| 53..51 | tbC2..0 | 2 | a..8 |

## 5.4. Pattern Section

As described before, the pattern is generated by concatenating the output pattern from all the ESIM tests described in section 3. The hardware reset sequence is performed only in the beginning of the test. Also, the first packet is omitted from each of the input decks used to test the BTT. The tests are sequenced as given below, and the entire test consists of 470 packets.

1. Parity error detection mechanism test with 2 packets
2. RC idle with 8 packets
3. RC point-to-point with 6 packets
4. RC multpoint with 6 packets
5. 7 packets for each of the 64 addresses in the BTT

## 5.5. Test Performance

The test was conducted at a speed of 21$MHz$ on the fabricated version of the BTT and proved to be successful implying that it would be easy to test version 2 when it is received from the fabricator. The ring oscillator frequency for version 1 was 31$MHz$ for 31 stages implying that a gate delay was about 0.5$ns$. The rise and fall times from the output pads were approximately 5.5$ns$ while on the LV500. These were measured to be approximately 2.3$ns$ while on a mock board.