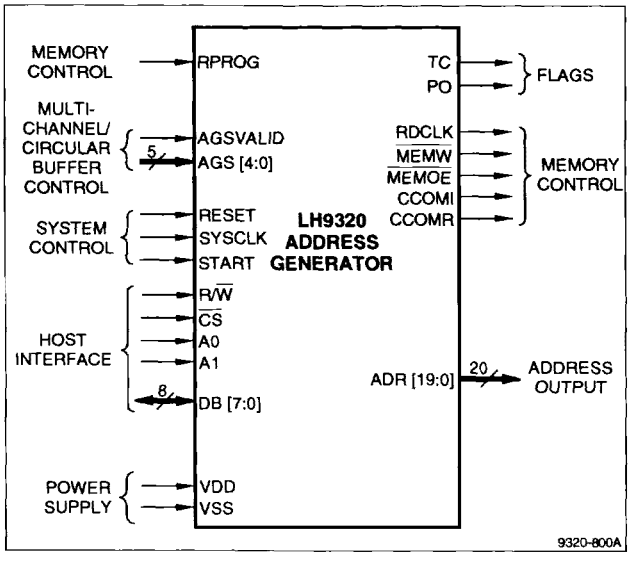
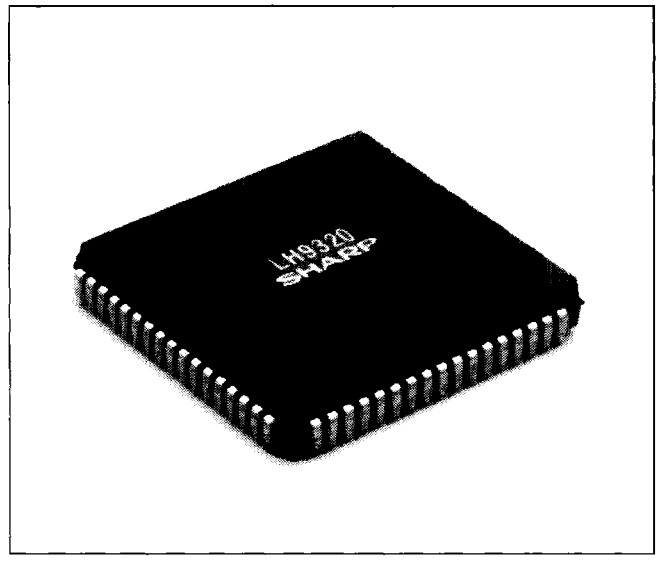
SHARP

Data Sheet

LH9320

DSP Address Generator





FEATURES

- 40 MHz Clock Rate
- Fast Convolution Overlap/Discard
- Digit Reverse on 1st Column
- Real/Complex
- Finite Impulse Response (FIR) Filters
- Interpolate (Index Filling)
- Fast Cosine Transform Separation Pass
- Index Padding
- Two-at-a-Time Real FFT Separation Pass
- Decimate
- Double Length Real FFT Separation Pass
- Radix-2, Radix-4, Radix-16, Mixed Radix, Data and Twiddle
- Multichannel Circular Buffering
- Multichannel Sample Request Arbitration
- System Level Timing Signals
- Memory Write Pulse Generation
- 68-Pin PLCC Package
- 0.8 Micron CMOS Technology

FUNCTIONAL DESCRIPTION

The LH9320 Address Generator (AG) is a high-speed memory addressing device that is uniquely designed to create address patterns for digital signal processors with real or complex memory arrays. The AG is a self-contained, small-pin-count device with virtually no computational overhead. The AG is designed to be used with SHARP's LH9124 Digital Signal Processor and other compatible execution units. The LH9320 can generate over 150 address patterns. Table 1 summarizes the AG's address pattern set.

The AG generates a series of addresses for memory arrays. It is easily programmed to generate addressing sequences between four points and one million (1 Meg) points. The AG is designed for standalone operation or as a peripheral on a host CPU bus. The LH9320 contains 32 words of program memory that can be programmed via the 8-bit data bus DB[7:0] (internal memory mode). These 32 words (instructions) of program memory are sufficient to generate algorithms for many applications. For example, a 1 mega point, radix-16 FFT can be performed using only five address patterns.

In contrast, in external memory mode, program memory may be bypassed and the LH9320 programmed in external memory mode. For example, an adaptive filtering algorithm that requires thousands of computed passes is one type of algorithm that could be performed in external mode.

Lastly, the LH9320 uses a unique circular-buffering technique for real-time multichannel DSP systems.

Table 1. LH9320 Address Pattern Set Summary

MNEMONIC	DESCRIPTION
FAST FOURIER TRANSFORMS (FFTs)	
BF2n (0 to 19)	Radix-2 data addresses
BF4n (0 to 9)	Radix-4 data addresses
BF16n (0 to 4)	Radix-16 data addresses
TF2n (0 to 19)	Radix-2 twiddle factor addresses
TF4n (0 to 9)	Radix-4 twiddle factor addresses
TF16n (0 to 4)	Radix-16 twiddle factor addresses
MXB24n (0 to 8)	Mixed radix (2,4) data addresses
MXB216n (0 to 3)	Mixed radix (2,16) data addresses
MXB416n (0 to 3)	Mixed radix (4,16) data addresses
MXB2416n (0 to 3)	Mixed radix (2,4,16) data addresses
MXT24n (0 to 8)	Mixed radix (2,4) twiddle addresses
MXT216n (0 to 3)	Mixed radix (2,6) twiddle addresses
MXT416n (0 to 3)	Mixed radix (4,16) twiddle addresses
MXT2416n (0 to 3)	Mixed radix (2,4,16) twiddle addresses
RBF0	Digit-reversed data address column 0
SEPARATION PASSES	
BRFTL	2-at-a-time real FFT separation pass, 2N, load
BRFTLS	2-at-a-time real FFT separation pass, N, load
BRFTU	2-at-a-time real FFT separation pass, 2N, unload
BRFTUS	2-at-a-time real FFT separation pass, N, unload
BFCTL	Fast cosine transform separation pass data addresses, 2N (long), load
BFCTT	Fast cosine transform separation pass twiddle addresses, 2N
BFCTUS	Fast cosine transform separation pass data addresses, N (short), load
BFCTU	Fast cosine transform separation pass data addresses, 2N, unload
BFCTUP	Fast cosine transform separation pass data addresses, 2N, unload using PO flag
BFCTUEP	Fast cosine transform separation pass data addresses, 2N, unload using early PO flag
BFCTULP	Fast cosine transform separation pass data addresses, 2N (long), unload using late PO flag
DECIMATION	
DECIM	Decimate
ADECIM	Auto decimate

MNEMONIC	DESCRIPTION
FINITE IMPULSE RESPONSE (FIR) FILTERS	
LPFIR1	Linear phase FIR, odd length, even symmetry
LPFIR2	Linear phase FIR, even length, even symmetry
LPFIR3	Linear phase FIR, odd length, odd symmetry
LPFIR4	Linear phase FIR, even length, odd symmetry
GENERAL PURPOSE ADDRESSING/UTILITIES	
MODINC	Modulo increment
MODDEC	Modulo decrement
INTER	Interpolate/index fill
INTERP	Interpolate/index fill using PO flag
INTEREP	Interpolate/index fill using early PO flag
INTERLP	Interpolate/index fill using late PO flag
PADHIGH	Pad at end of sequence
PADHIGHP	Pad at end of sequence using PO flag
PADHIGHEP	Pad at end of sequence using early PO flag
PADHIGHLP	Pad at end of sequence using late PO flag
PADLOW	Pad at start of sequence
PADLOWP	Pad at start of sequence using PO flag
PADLOWEP	Pad at start of sequence using early PO flag
PADLOWLP	Pad at start of sequence using late PO flag
OVERLAP	Overlap
DISCARD	Discard
DISCARDP	Discard using PO flag
DISCARDEP	Discard using early PO flag
DISCARDLP	Discard using late PO flag
CMAG	Square of magnitude of a complex number
INC	Index increment
NOP	No operation
CIRCULAR BUFFER ADDRESSING	
CBUFFIR	Circular buffering for FIRs
CBUFFFT	Circular buffering for FFTs
OTHER	
CLRSIG	Clear signature
VIEWSIG	View signature

INTERFACE SIGNALS

The LH9320 interface signals are shown in Figure 1 and summarized in Table 2.

Table 2. LH9320 Interface Signals

SIGNAL *	DIRECTION **	SIGNAL NAME
DB[7:0]	I/O	Host Interface Data Bus
A1, A0	I	Host Control
R/$\overline{W}$	I	Read/Write Host
$\overline{CS}$	I	Chip Select
SYSCLK	I	System Clock
START	I	Start of a Pass
TC	O	Terminal Count
ADR[19:0]	O	Address Outputs
AGS[4:0]	I	Circular Buffer Channel Sample Request
AGSVALID	I	Circular Buffer Valid Channel Flag
CCOMR	O	Coefficient Complement Real
CCOMI	O	Coefficient Complement Imaginary
RESET	I	Chip Reset
PO	O	Programmable Output
$\overline{MEMW}$	O	Memory Write
$\overline{MEMOE}$	O	Memory Output Enable
RDCLK	O	Read Clock
RPROG	I	Programmable Pulsewidth for $\overline{MEMW}$
VDD	P	+ 5 volt power supply
VSS	P	Ground for the chip

* Numbers in brackets indicate the number of bits. For example, ADR[19:0] indicates 20 bits.

** I = Input, O = Output, I/O = Input and Output, P = Power.

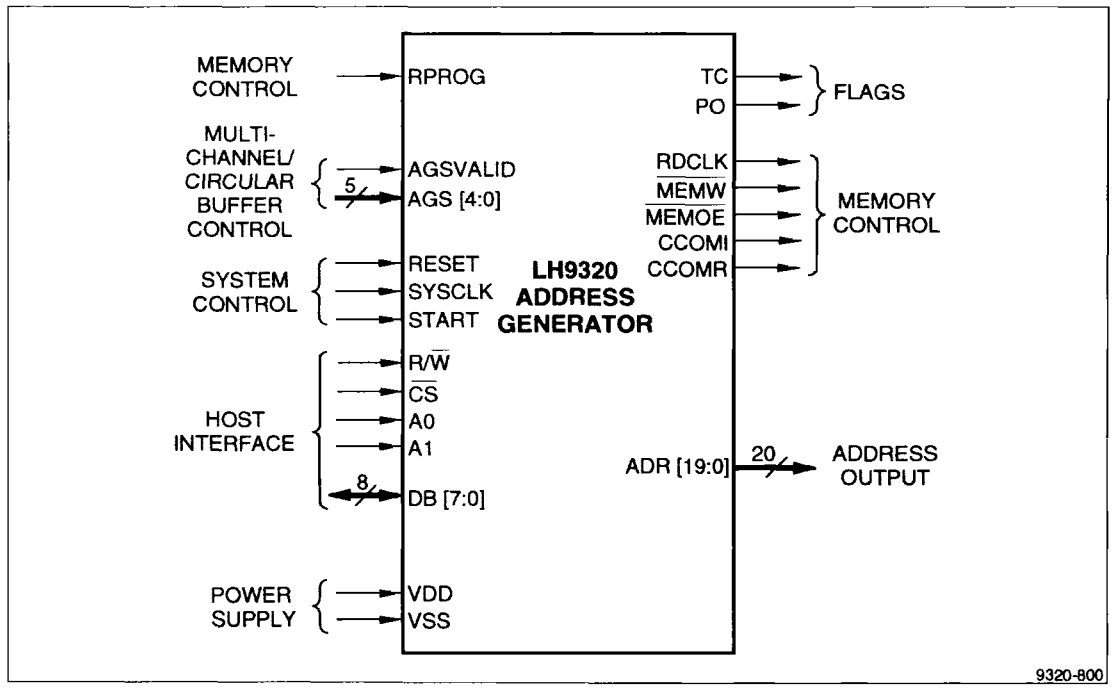


Figure 1. LH9320 Interface Signals

DB[7:0] (Host Interface Data Bus)

In internal memory mode, the 8-bit data bus, DB[7:0], is used to load LH9320 registers via a two step scheme: register address first, then register data. In external memory mode, the register address is set to one of three predefined locations based on the A1, A0 signals. In this case, the DB[7:0] data bus transfers data to the selected predefined location.

A1, A0 (Host Control)

For internal memory mode, the A1, A0 signals define the input to the DB[7:0] data bus as a register address (A1 is LOW, A0 is HIGH) or as register data (A1 is LOW, A0 is LOW).

In external memory mode, A1, A0 sets the register address to one of the following predefined address locations:

- Location 31 of program memory (Host Register Address 1F)
- Location 31 of pipeline/memory latency memory (Host Register Address 3F)
- MODE register (Host Register Address A2)

Refer to *Host Interface* and Table 6 for more information.

R/$\overline{W}$ (Read/Write Host)

R/$\overline{W}$ loads each Host register on its rising edge, effectively becoming the clock for all Host registers. For the configuration registers, the data is available after the rising edge of the R/$\overline{W}$ signal. In contrast, the program memory, pipeline/memory latency memory and channel length/overlap memory makes the data available after the falling edge of the R/$\overline{W}$ signal. In other words, a full transition (LOW to HIGH and HIGH to LOW) must be made before data from the on-chip memories can be used.

Note: R/$\overline{W}$ loads the Host registers. The SYSCLK signal controls the execution and output of the LH9320 after the Host is loaded. The R/$\overline{W}$ and SYSCLK signals do NOT have to be synchronous with one another. However, the START signal must be synchronized to the SYSCLK.

$\overline{CS}$ (Chip Select)

When set to LOW, $\overline{CS}$ enables the LH9320 to communicate with the Host to program the registers. When $\overline{CS}$ is set HIGH, DB[7:0] is disabled to a high-impedance state.

RESET (Chip Reset)

When set HIGH, RESET does a hardware reset of the LH9320 chip. When set LOW, normal operation of the LH9320 chip is enabled.

Note: RESET must be synchronized to the SYSCLK signal and be a minimum of two SYSCLK cycles long and a minimum of two R/$\overline{W}$ cycles long.

SYSCLK (System Clock)

SYSCLK controls the execution and output of the address calculation logic. A new address is generated on each rising edge of the SYSCLK signal. Refer to *Address Calculation* for more information.

Note: the START signal must be synchronized to SYSCLK.

START (Start of a Pass)

START is an edge triggered signal that initiates the generation of each addressing sequence. After the initial START, the LH9320 requires five SYSCLK cycles of latency before the first valid address is output. Each subsequent START can be set to zero latency provided START is issued before the end of the current sequence.

Note: START must be synchronized to the SYSCLK signal and must not be issued at least two SYSCLK cycles after the last write to the host memory.

TC (Terminal Count)

TC indicates when an addressing sequence has been completed. As a default, TC goes HIGH one cycle before the end of the current addressing sequence. TC remains HIGH until the next START signal causes a valid address to be generated. The LOW to HIGH transition of TC can be delayed by one cycle (to the actual end of the sequence) by setting bit 3 in the MODE register to HIGH. Refer to *MODE Register.*

Note: when circular buffering is not active, the TC signal is affected by the PAUSE register. If PAUSE is used, TC becomes active one cycle before the sequence length + PAUSE.

ADR[19:0] Address Outputs

ADR[19:0] are the address outputs from the LH9320.

AGS[4:0] Circular Buffer Channel Sample Request

These signals select a channel when using the circular buffer. These signals may either be processed by an on-chip sample arbitrator or by an off-chip approach. See *Multichannel Circular Buffer/Sample Arbitrator.*

AGSVALID Circular Buffer Channel Sample Request Valid

The AGSVALID input indicates the status of the AGS[4:0] signals in external arbitration mode. The AGS[4:0] signals are interpreted as valid channels if AGSVALID is HIGH. When set LOW, no active channels are represented by the AGS[4:0] signals in external arbitration mode. (For internal arbitration, see Table 7.)

CCOMR, CCOMI (Coefficient Complement Real and Imaginary)

These two signals are referred to as the real coefficient complement (CCOMR) and imaginary coefficient complement (CCOMI) signals.

For causal systems, zero phase is not attainable and linear phase is usually the applied design consideration. In particular, the finite impulse response sequence of a linear phase has a special kind of symmetry which stores only half as many coefficients to manipulate the convolution, thereby saving half of the memory space.

The four address patterns that generate addressing sequences for linear phase FIR filters include LPFIR1, LPFIR2, LPFIR3 and LPFIR4. The LPFIR1 to LPFIR4 address patterns use the N register to specify the length of the filter. The MEMSIZE register defines the memory leap value. In each case, the increased length for the consecutive addresses is equal to the memory size divided by the filter length.

In particular, LPFIR3 and LPFIR4 are used when the impulse response does not use mirror image symmetry, but anti-symmetry. For example, wideband differentiators and Hilbert transforms are applied examples of FIR filters with anti-symmetrical impulse response sequences.

Note: for the same N and MEMSIZE, LPFIR3 generates the same addressing sequence as LPFIR1. The difference between these two address patterns is the CCOMR, CCOMI signals. When CCOMR, CCOMI are set to negate the data, these signals generate zeros for both LPFIR1 and LPFIR3 in the first half of the sequence. In the second half, CCOMR, CCOMI generate zeros for LPFIR1 and ones for LPFIR3. This means, the anti-symmetry form can be obtained from the same sequence. A similar relationship occurs between LPFIR2 and LPFIR4.

The LH9320 accomplishes the conversion of real and imaginary coefficients by attaching the CCOMR and CCOMI pins to corresponding pins on the LH9124.

Note: after a reset or when CCOMR and CCOMI are not used in the current addressing sequence, these signals are set to a high-impedance state.

PO (Programmable Output)

PO indicates when an index address is being produced. When HIGH, the PO signal indicates that the actions shown in Table 3 need to be performed.

Note: after a reset or if PO is not used by the current addressing sequence, this signal is set to high-impedance.

Table 3. PO Signal Selections

MNEMONIC	USER ACTION REQUIRED WHEN PO SIGNAL IS SET ACTIVE
INTERP	Data needs to be filled.
INTEREP	Data for the next address needs to be filled.
INTERLP	Data for the previous address needs to be filled.
PADHIGHP	Data needs to be padded.
PADHIGHEP	Data for the next address needs to be padded.
PADHIGHLP	Data for the previous address needs to be padded.
PADLOWP	Data needs to be padded.
PADLOWEP	Data for the next address needs to be padded.
PADLOWLP	Data for the previous address needs to be padded.
DISCARDP	Data needs to be discarded.
DISCARDEP	Data for the next address needs to be discarded.
DISCARDLP	Data for the previous address needs to be discarded.
BFCTUP	Data for the highest frequency component needs to be written.
BFCTUEP	Data for the next address (highest frequency component) needs to be written.
BFCTULP	Data for the previous address (highest frequency component) needs to be written.

$\overline{\text{MEMW}}$ (Memory Write)

$\overline{\text{MEMW}}$ indicates when the execution unit is to write to the SRAM. This signal is controlled by bit 7 of the pipeline/memory latency memory. $\overline{\text{MEMW}}$ is pulsed LOW when data is to be written to the SRAM. $\overline{\text{MEMW}}$ is set HIGH when data is read from the SRAM.

Note: $\overline{\text{MEMW}}$ is set HIGH after a reset.

$\overline{\text{MEMOE}}$ (Memory Output Enable)

$\overline{\text{MEMOE}}$ controls the output enable of the SRAM. This signal is controlled by bit 7 of the pipeline/memory latency memory. $\overline{\text{MEMOE}}$ is set LOW when data is to be read from the SRAM. $\overline{\text{MEMOE}}$ is set HIGH when data is to be written to the SRAM.

Note: $\overline{\text{MEMOE}}$ is set HIGH after a reset.

RDCLK (Read Clock)

The rising edge of RDCLK indicates when the execution unit is to read from the SRAM. This signal is controlled by bit 7 of the pipeline/memory latency memory.

Note: RDCLK is set LOW after a reset.

RPROG (Programmable Pulsewidth for $\overline{\text{MEMW}}$)

$\overline{\text{MEMW}}$ must be set HIGH when the address changes. RPROG is used to adjust the timing of the $\overline{\text{MEMW}}$ signal and to ensure that $\overline{\text{MEMW}}$ is HIGH when the address changes. RPROG is typically connected to a 10kΩ 1% pullup resistor. This 10kΩ resistor gives the timing for $\overline{\text{MEMW}}$ shown in the AC Electrical Characteristics table.

VDD

VDD is the power supply for the chip.

VSS

VSS is the ground for the chip.

BLOCK LEVEL DESCRIPTION (Figure 2)

The LH9320 Address Generator consists of the following functional sections: Host, Address Calculation, Clock Control, Program Counter and the Multichannel Circular Buffer.

HOST

The Host enables the LH9320 to be programmed. The Host is composed of two sections as shown in Figure 3, the Host Memory and the Host Interface.

HOST MEMORY

The Host Memory uses 223 programmable memory locations. The host memory contains the following registers: Program Memory, Pipeline/Memory Latency Memory, Channel Length/Overlap Memory, and Configuration Registers. Refer to Figure 4 and Table 4 for details of the Host Memory.

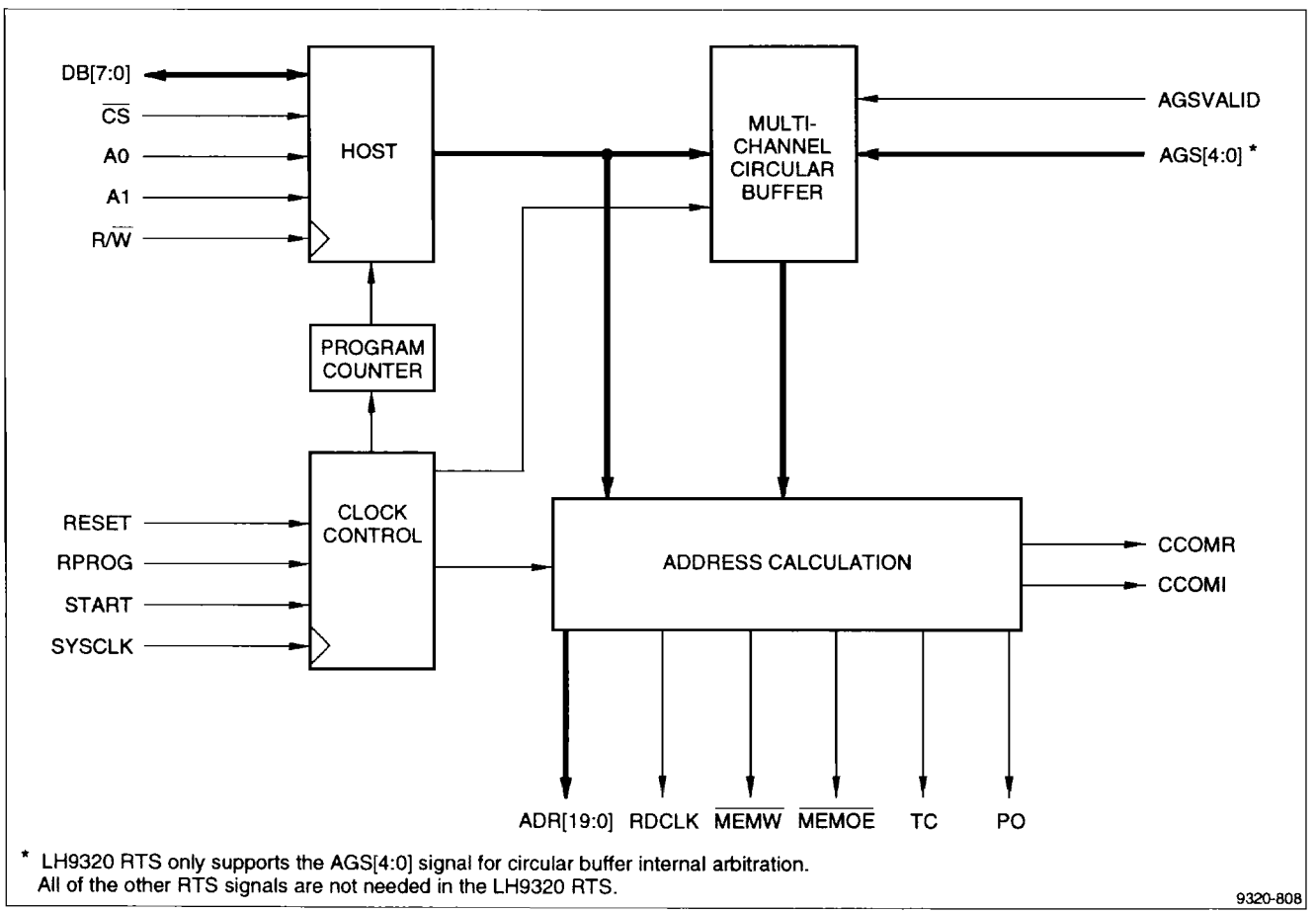
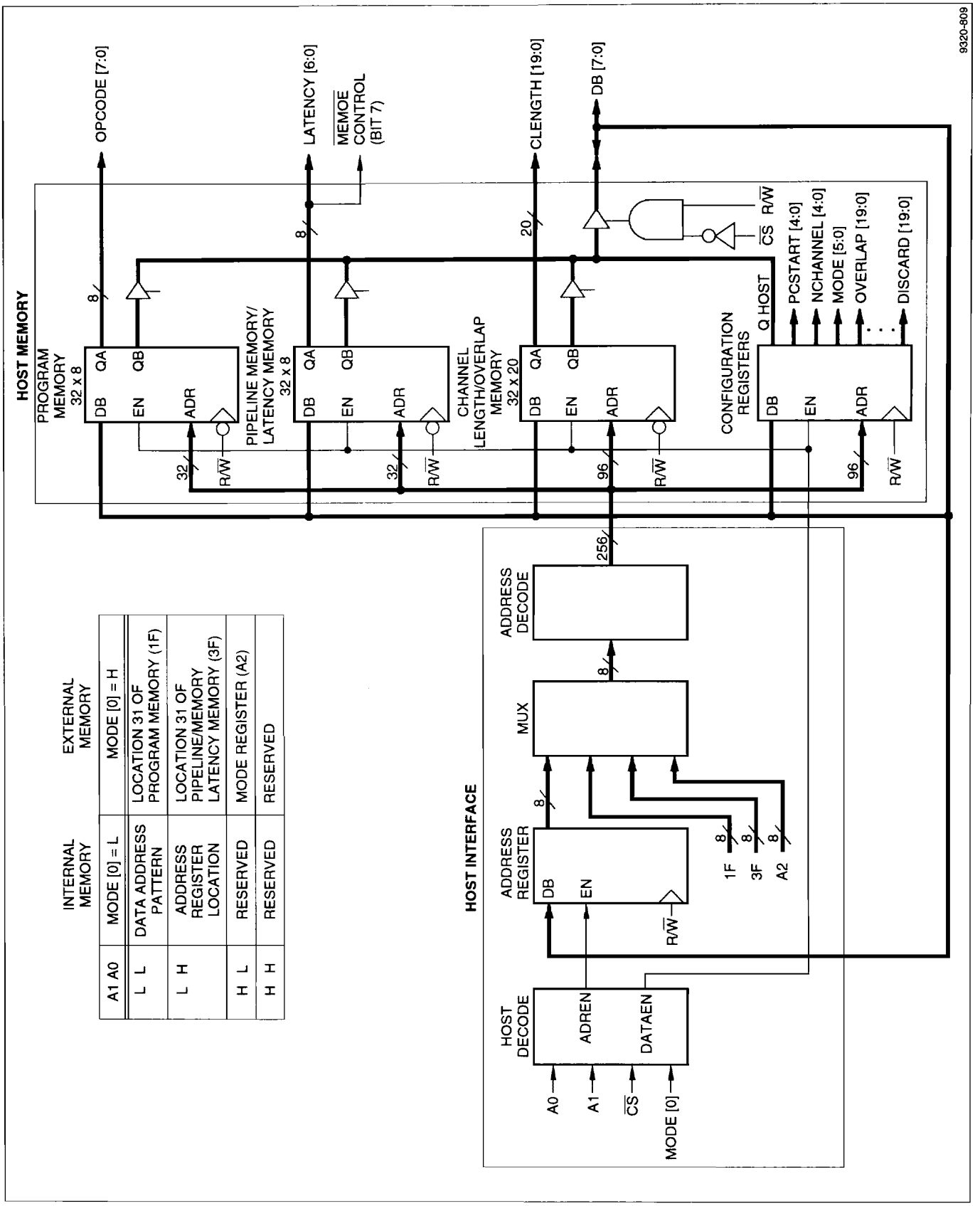


Figure 2. LH9320 Simplified Block Diagram



A1 A0	MODE [0] = L	MODE [0] = H
L L	DATA ADDRESS PATTERN	LOCATION 31 OF PROGRAM MEMORY (1F)
L H	ADDRESS REGISTER LOCATION	LOCATION 31 OF PIPELINE/MEMORY LATENCY MEMORY (3F)
H L	RESERVED	MODE REGISTER (A2)
H H	RESERVED	RESERVED

Figure 3. LH9320 Host Block

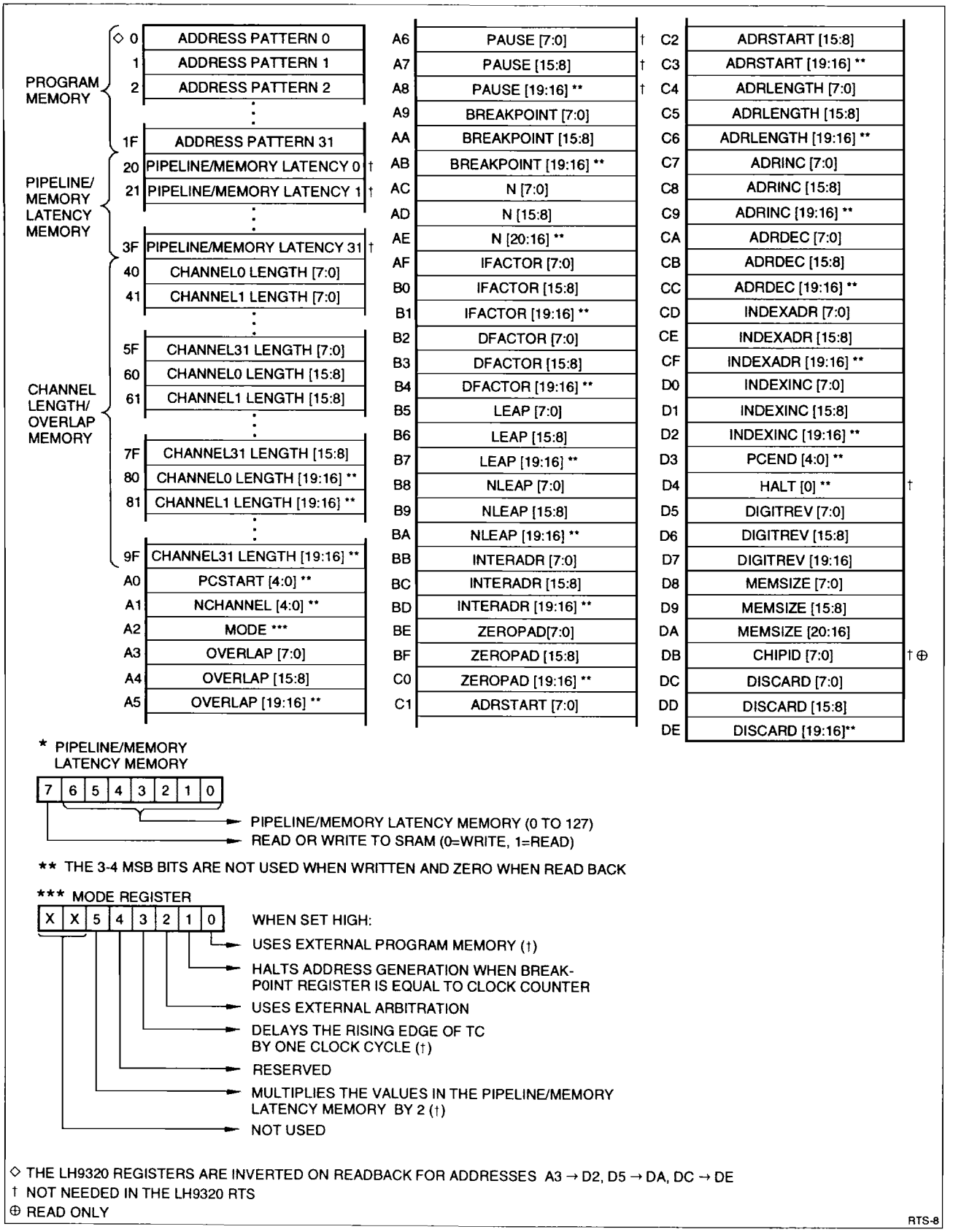


Figure 4. LH9320 Host Memory Map

Table 4. LH9320 Memory Description

ADDRESS	BITS USED	NAME	VALID RANGE (HEX)	DESCRIPTION
0-1F	8	PROGRAM MEMORY	00-FF	32, 8-bit register locations reserved to store address patterns.
20-3F	8	PIPELINE/ MEMORY LATENCY MEMORY	00-FF	32, 8-bit register locations reserved to store latency values (bits [6:0]). Bit 7 indicates if the LH9320 is reading from or writing to the SRAM.
40-9F	20	CHANNEL LENGTH/ OVERLAP MEMORY	FIR: 00004-100000 * FFT: 00000-FFFFF	32, 20-bit register locations reserved to define the channel length or channel overlap amount for each of the channels in the circular buffer. CBUFFIR: Defines the channel length. CBUFFFT: Defines the channel overlap.
A0	5	PCSTART	00-1F	For internal memory mode, points to the first address pattern to be executed in the program memory (disabled in external memory mode).
A1	5	NCHANNEL	00-1F	Identifies the number of channels to be managed by the circular buffer.
A2	6	MODE	00-3F	Defines the various modes of operation for the LH9320 as shown in Figure 4.
A3-A5	20	OVERLAP	00000-FFFFF	OVERLAP: Defines the amount of overlap for the data.
A6-A8	20	PAUSE	00000-FFFFF	Defines the number of cycles the LH9320 is to pause by immediately after an addressing sequence is completed.
A9-AB	20	BREAKPOINT	00000-FFFFF	Halts the program at a specific address when bit 1 in the MODE register is set high. When the number of addresses generated in the sequence is equal to the value in the BREAKPOINT register, the addressing sequence ends.
AC-AE	21	N	000000-100000	For most address patterns, defines the length of the addressing sequence, except those that use 2N or ADRLENGTH.
AF-B1	20	IFACTOR	00001-80000	INTER/P/EP/LP **: Defines the number of indexing addresses to be inserted between each generated address in the sequence.
B2-B4	20	DFACTOR	00000-FFFFF	DECIM: Defines the value that each address is to be incremented by each time an address is generated. ADECIM: Uses the DFACTOR value as in DECIM, but only for the first N addresses.
B5-B7	20	LEAP	00000-FFFFF	ADECIM: Defines the accumulation amount for the DFACTOR value.
B8-BA	20	NLEAP	00000-FFFFF	ADECIM: Defines the number of sets of N addresses that are to be generated.
BB-BD	20	INTERADR	00000-FFFFF	INTER/P/EP/LP: Defines the last interpolated address that was generated. (Default = FFFFF)
BE-C0	20	ZEROPAD	00000-FFFFF	PADLOW/P/EP/LP: Indicates the number of indexing addresses that are to be generated before the addressing sequence starts.

* Because the LH9320 is a 20-bit unit, 100000 = 00000.

** Indicates the address pattern and related extensions, such as INTER, INTERP, INTEREP, INTERLP.

Table 4 (cont'd). LH9320 Memory Description

ADDRESS	BITS USED	NAME	VALID RANGE (HEX)	DESCRIPTION
BE-C0 (cont'd)	20	ZEROPAD	00000-FFFFF	PADHIGH/P/EP/LP **: Indicates when the indexing address is to be generated to replace the current sequence.
C1-C3	20	ADRSTART	00000-FFFFF	PADLOW/P/EP/LP, PADHIGH/P/EP/LP, MODINC, MODDEC and INC: Defines the starting address in the current sequence.
C4-C6	20	ADRLENGTH	00004-100000 *	PADLOW/P/EP/LP, PADHIGH/P/EP/LP, MODINC, MODDEC and INC: Indicates the number of points in the current sequence.
C7-C9	20	ADRINC	00000-FFFFF	PADLOW/P/EP/LP, PADHIGH/P/EP/LP, MODINC and INC: Indicates the amount each subsequent address is to be incremented by when generated.
CA-CC	20	ADRDEC	00000-FFFFF	MODDEC: Indicates the amount each subsequent address pattern is to be decremented by when generated.
CD-CF	20	INDEXADR	00000-FFFFF	PADLOW/P/EP/LP, PADHIGH/P/EP/LP, INTER/P/EP/LP, DISCARD/P/EP/LP, BFCTU/P/EP/LP: Points to an address space outside of the working space memory for a particular address pattern.
D0-D2	20	INDEXINC	00000-FFFFF	PADLOW/P/EP/LP, PADHIGH/P/EP/LP, INTER/P/EP/LP, DISCARD/P/EP/LP: Identifies the amount to increment the INDEXADR pointer value by each time an indexing address is generated.
D3	5	PCEND	00-1F	For internal memory mode, points to the last address pattern to be executed in the program memory (disabled in external memory mode).
D4	1	HALT	0-1	Halts the current addressing sequence.
D5-D7	20	DIGITREV	00000-FFFFF	Calculates the digit reverse sequence for the first column of an FFT based on the product of the radix combination.
D8-DA	21	MEMSIZE	000000-100000	CBUFFIR and CBUFFFT: Defines the physical memory size for the generated addressing sequence in a circular buffering operation. TF2n (0 to 19), TF4n (0 to 9), TF16n (0 to 4), MXT24n (0 to 8), MXT216n (0 to 3), MXT416n (0 to 3), MXT2416n (0 to 3), DECIM, ADECIM, LPFIR1 to LPFIR4, BFCTT: Defines the physical memory size that contains 360 degrees of coefficients for address patterns on the coefficient side of the execution unit.
DB	8	CHIPID	41	Chip identification/revision. (Read only.)
DC-DE	20	DISCARD	00000-FFFFF	DISCARD/P/EP/LP: Defines the number of addresses to be discarded.

* Because the LH9320 is a 20-bit unit, 100000 = 00000

** Indicates the address pattern and related extensions, such as INTER, INTERP, INTEREP, INTERLP.

Program Memory

The program memory contains 32, 8-bit registers (locations 0-1F) that can store up to 32 address patterns.

Pipeline/Memory Latency Memory

The pipeline/memory latency memory contains 32, 8-bit registers (locations 20-3F) that store the latency value for each address pattern. The latency value defines when the related address pattern is to be generated. Memory latency values can range from 0 to 127 (bits [6:0]). However, by setting bit 5 of the MODE register to HIGH, the latency numbers can be multiplied by 2, thus extending the latency range to 254. Bit 7 of the pipeline/memory latency memory indicates that the LH9320 is reading from or writing to the SRAM. Refer to *$\overline{\text{MEMOE}}$ (Memory Output Enable) Signal*, *$\overline{\text{MEMW}}$ (Memory Write) Signal*, and to *MODE Register*.

Channel Length/Overlap Memory

The channel length/overlap memory contains 32, 20-bit registers (locations 40-9F) that define the channel length or the channel overlap amount, dependent on the addressing sequence. In either case, each 20-bit register contains two 8-bit registers and one 4-bit register (8 + 8 + 4 = 20), to a total of 96 registers for 32 channels. Refer to *Multichannel Circular Buffer*.

Note: the address pattern CBUFFIR uses this memory to define the channel length, while CBUFFFT uses the memory to define the channel overlap amount.

Configuration Registers

The configuration registers (A0-DE) comprise the remainder of the LH9320's Host Memory.

PCSTART Register

The 5-bit PCSTART register (A0) points to the first address pattern to be executed in program memory. In external memory mode, PCSTART is disabled and the first address pattern to be executed points to location 31 of the program memory.

PCEND Register

The 5-bit PCEND register (D3) points to the last address pattern to be executed in program memory. In external memory mode, PCEND is disabled and the last address pattern to be executed points to location 31 of the program memory.

NCHANNEL Register

The 5-bit NCHANNEL register (A1) identifies the number of channels to be managed by the circular buffer (binary 31 indicates 32 channels; binary 0 indicates 1 channel).

MODE Register

The 8-bit MODE register (A2) defines the various modes of operation of the LH9320. The bits for the MODE register shown in Figure 4 are defined as follows:

Bit 0 – Determines when the LH9320 is in internal or external memory mode (internal is LOW, external is HIGH).

Bit 1 – Enables the BREAKPOINT register (enable is HIGH, disable is LOW).

Bit 2 – Enables the AGS[4:0] signals to directly control the addressing to the circular buffer memory or to use the internal priority encoder (direct AGS[4:0] addressing is HIGH, internal priority encoder is LOW).

Bit 3 – Delays the low-to-high transition of TC by 1 cycle to the actual end of the sequence (1 cycle delay is HIGH, no delay is LOW). If delayed, TC goes HIGH at the same time that the last address is generated. Bit 3 does not affect the falling edge of TC.

Bit 4 – Reserved. It is recommended that Bit 4 be programmed to zero.

Bit 5 – Indicates when the values in the pipeline/memory latency memory are multiplied by 2. (Multiplied by 2 is HIGH.)

Bits 6 and 7 – Not used.

OVERLAP Register

The 20-bit (two 8-bit and one 4-bit register) OVERLAP register (A3-A5) is used by the OVERLAP address pattern to determine the amount of overlap (00000 H to FFFFF H) for the data.

DISCARD Register

The 20-bit (two 8-bit and one 4-bit register) DISCARD register (DC-DE) is used by DISCARD, DISCARDP, DISCARDEP and DISCARDLP address patterns to indicate the number of addresses to be discarded (00000 H to FFFFF H).

PAUSE Register

The 20-bit (two 8-bit and one 4-bit register) PAUSE register (A6-A8) defines the number of cycles the LH9320 is to pause immediately after an addressing sequence is completed. If used, the effective length (N, ADRLENGTH or 2N) is changed to the sequence length + PAUSE. In contrast to the pipeline/memory latency memory which has a unique value for each address pattern, PAUSE has a global value for all address patterns.

The TC signal is affected by the PAUSE register. If PAUSE is used, TC becomes active one cycle before sequence length + PAUSE.

Note: if MODE register bit 3 is set HIGH, TC transitions at the same time as sequence length + PAUSE. The circular buffer address patterns, CBUFFIR and CBUFFFT, are not affected by the PAUSE Register

BREAKPOINT Register

The 20-bit (two 8-bit and one 4-bit register) BREAKPOINT register (A9-AB) defines the break point for an addressing sequence (halts at a specific number of addresses). The BREAKPOINT register is used only when bit 1 in the MODE register is set HIGH.

If used, the value in the BREAKPOINT register becomes the new length of the current sequence (instead of N, ADRLENGTH, or 2N). When the number of addresses generated in the sequence reaches the value in the BREAKPOINT register, the addressing sequence ends. In turn, the TC signal becomes active one cycle before the value in the BREAKPOINT register is reached. The BREAKPOINT stops the current addressing sequence provided that the sequence has not already ended and has not entered a PAUSE state.

Note: the circular buffer address patterns, CBUFFIR and CBUFFFT, are not affected by the BREAKPOINT register.

N Register

The 21-bit (two 8-bit and one 5-bit register) N register (AC-AE) defines the length of the addressing sequence for most address patterns. In contrast, some address patterns use a sequence length of 2N, such as BRFTL and BRFTU, or use a value in the ADRLENGTH register instead of N, such as MODINC and MODDEC.

Note: because internally the LH9320 requires the use of all values of N which are divisible by powers of $2\left(\frac{N}{2}, \frac{N}{4}, \frac{N}{8}, \ldots\right)$, the N register is 21 bits wide. The multiples of N are generated internally using a shift right. N must always be a power of 2 between 000004 H and 100000 H.

IFACTOR (Interpolate Factor) Register

The 20-bit (two 8-bit and one 4-bit register) IFACTOR register (AF-B1) is used by the INTER, INTERP, INTEREP and INTERLP address patterns. The IFACTOR register defines the number of indexing addresses that are to be inserted (filled) between each generated address in the sequence. The IFACTOR register value has to be an even multiple of N, this means:

$$\frac{N}{IFACTOR} = \text{integer.}$$

In addition, 0 is not a legal value for the interpolate factor.

DFACTOR (Decimate Factor) Register

The 20-bit (two 8-bit and one 4-bit register) DFACTOR register (B2-B4) is used by the DECIM and ADECIM address patterns as follows:

DECIM – The addressing sequence will start at address 0, and will continue accumulating by the amount given by the DFACTOR. The sequence will stop when the number of addresses generated reaches N.

ADECIM – Uses the DFACTOR value as in DECIM, but only for the first N addresses. Then, DFACTOR will be internally accumulated (to DFACTOR = DFACTOR + LEAP) for the next N addresses. Then, DFACTOR will be internally accumulated again (to DFACTOR = DFACTOR + LEAP) for the next N address, etc.

LEAP Register

The 20-bit (two 8-bit and one 4-bit register) LEAP register (B5-B7) is used by the ADECIM address pattern to define the accumulation amount for the DFACTOR value. The DFACTOR value is increased by LEAP for each set of N addresses.

NLEAP Register

The 20-bit (two 8-bit and one 4-bit register) NLEAP register (B8-BA) is used by the ADECIM address pattern to define the number of sets of N addresses that are to be generated.

INTERADR (Interpolate Address) Register

The 20-bit (two 8-bit and one 4-bit register) INTERADR register (BB-BD) is used by the INTER, INTERP, INTEREP and INTERLP address patterns to define the starting address (INTERADR + 1) for these sequences. INTERADR = –1 (FFFFF) after RESET, then it keeps the last address generated by the above sequences.

Note: Figure 5 shows INTERADR is selected (SEL = 1) only the first time the INTER, INTERP, INTEREP or INTERLP address pattern is used after a reset. The ACL INTERADR register (not accessible by the user) is used by all subsequent address patterns (i.e., when SEL = 0).

ZEROPAD Register

The 20-bit (two 8-bit and one 4-bit register) ZEROPAD register (BE-C0) is used by the following address patterns:

PADLOW, PADLOWP, PADLOWEP, PADLOWLP – Indicates the number of indexing addresses that are to be generated before the actual sequence starts.

PADHIGH, PADHIGHP, PADHIGHEP, PADHIGHLP – Indicates when the indexing address is to be generated to replace the current sequence.

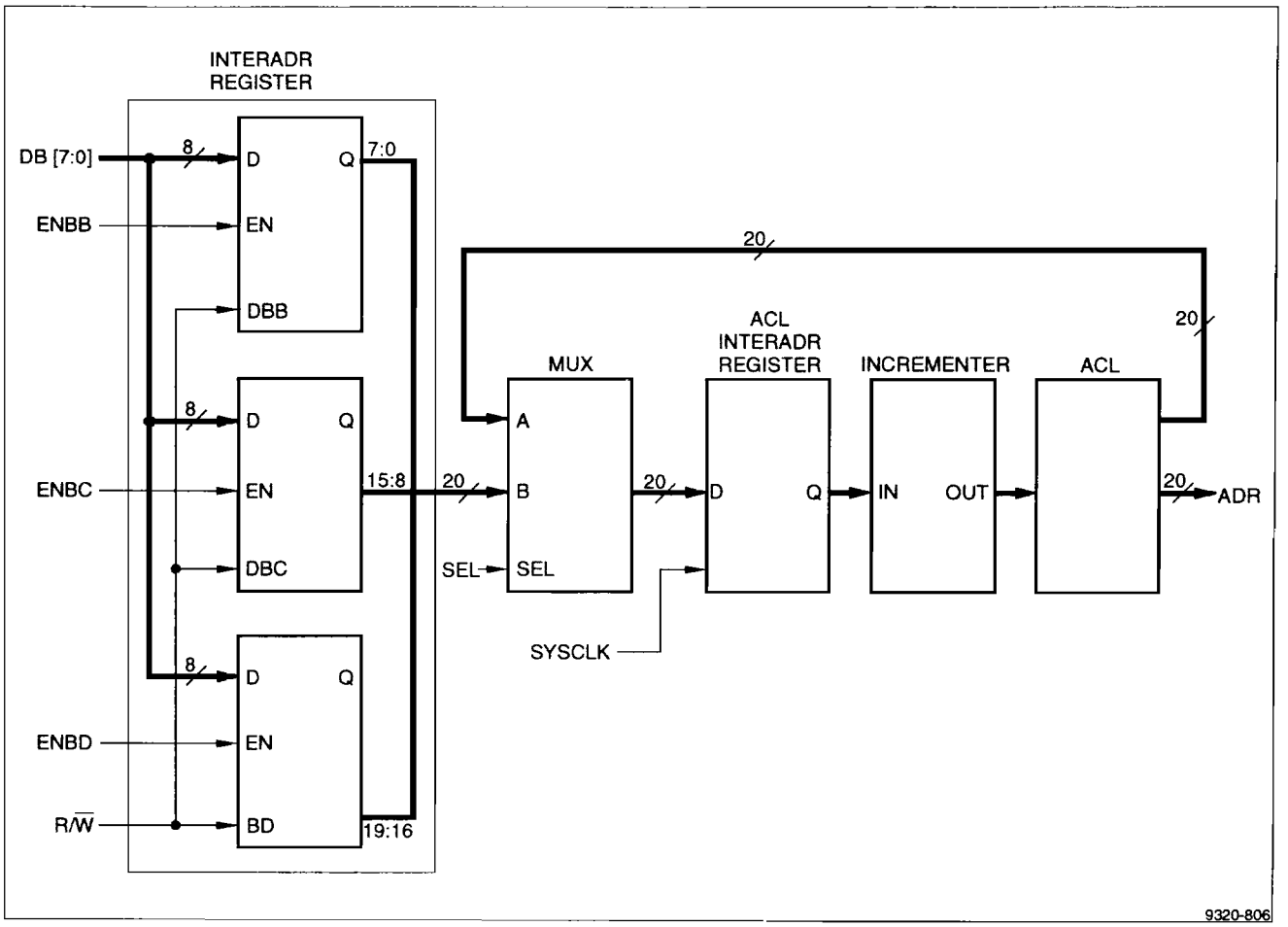


Figure 5. Reading the INTERADR Register

Note: the indexing address can be any value between 00000 H to FFFFF H and does not have to be less than the address length.

ADRSTART Register

The 20-bit (two 8-bit and one 4-bit register) ADRSTART register (C1-C3) is used by PADLOW, PADLOWP, PADLOWEP, PADLOWLP, PADHIGH, PADHIGHP, PADHIGHEP, PADHIGHLP, MODINC, MODDEC and INC to define the starting address in the sequence.

ADRLENGTH Register

The 20-bit (two 8-bit and one 4-bit register) ADRLENGTH register (C4-C6) is used by PADLOW, PADLOWP, PADLOWEP, PADLOWLP, PADHIGH, PADHIGHP, PADHIGHEP, PADHIGHLP, MODINC, MODDEC and INC to indicate the number of points the sequence contains. Unlike the N register, ADRLENGTH does not have to be a power of 2.

Note: if a length of 1 million (1 mega word) is required, the ADRLENGTH register must be set to zero.

ADRINC (Address Increment) Register

The 20-bit (two 8-bit and one 4-bit register) ADRINC register (C7-C9) is used by PADLOW, PADLOWP, PADLOWEP, PADLOWLP, PADHIGH, PADHIGHP, PADHIGHEP, PADHIGHLP, MODINC and INC to indicate the amount by which each subsequent address is to be incremented.

ADRDEC (Address Decrement) Register

The 20-bit (two 8-bit and one 4-bit register) ADRDEC register (CA-CC) is used by the MODDEC address pattern to indicate the amount by which each subsequent address is to be decremented.

INDEXADR (Index Address) Register

The 20-bit (two 8-bit and one 4-bit register) INDEXADR register (CD-CF) is used by the PADLOW, PADLOWP, PADLOWEP, PADLOWLP, PADHIGH, PADHIGHP, PADHIGHEP, PADHIGHLP, INTER, INTERP, INTEREP, INTERLP, DISCARD, DISCARDP, DISCARDEP, DISCARDLP, BFCTU, BFCTUP, BFCTUEP and BFCTULP address patterns. While the exact application for each address pattern differs, the INDEXADR value (00000 H to FFFFF H) can point to an address space outside of the working space memory for a particular address pattern.

Note: if the address patterns that use the indexing address are used on the input side of the execution unit, data must be written to the INDEXADR location prior to running the address pattern. If INDEXADR is used on the output side of the execution unit, the PO signal must be used in conjunction with the indexing address to determine when the data is to be written to the RAM.

INDEXINC Register

The 20-bit (two 8-bit and one 4-bit register) INDEXINC register (D0-D2) is used by the PADLOW, PADLOWP, PADLOWEP, PADLOWLP, PADHIGH, PADHIGHP, PADHIGHEP, PADHIGHLP, INTER, INTERP, INTEREP and INTERLP address patterns. The INDEXINC value (00000 H to FFFFF H) identifies the amount by which the INDEXADR value is to be incremented each time an indexing address is generated.

HALT Register

The 1-bit HALT register (D4) halts the current addressing sequence when set HIGH. If halted, the inherent four SYSCLK cycle latency will occur after the next START signal and before the first address is generated.

DIGITREV Register

The 20-bit (two 8-bit and one 4-bit register) DIGITREV register (D5-D7) is used to calculate the digit reverse sequence for the first column of an FFT. The DIGITREV value is determined based on the product of the radix combination. For example, an 8-point FFT could contain three different digit reverse sequences depending on the radix combination being used as follows:

Radix-2 × Radix-2 × Radix-2

Radix-2 × Radix-4

Radix-4 × Radix-2

Refer to Table 5 for more information.

Note: DIGITREV is the only register that requires the user to provide information concerning the entire algorithm being performed, rather than a specified pass.

Table 5. Digit Reverse Addressing for an 8-Point Input Sequence

REFERENCE ADDRESS NO.	DIGIT REVERSE SEQUENCE		
	2×2×2	2×4	4×2
0 0 0	0 0 0	0 (0 0)	(0 0) 0
0 0 1	1 0 0	1 (0 0)	(0 1) 0
0 1 0	0 1 0	0 (0 1)	(1 0) 0
0 1 1	1 1 0	1 (0 1)	(1 1) 0
1 0 0	0 0 1	0 (1 0)	(0 0) 1
1 0 1	1 0 1	1 (1 0)	(0 1) 1
1 1 0	0 1 1	0 (1 1)	(1 0) 1
1 1 1	1 1 1	1 (1 1)	(1 1) 1

MEMSIZE Register

The 21-bit (two 8-bit and one 5-bit register) MEMSIZE register (D8-DA) is used by the following address patterns:

CBUFFIR, CBUFFFT – Defines the physical memory size used by the circular buffer.

TF2n (0 to 19), TF4n (0 to 9), TF16n (0 to 4), MXT24n (0 to 8), MXT216n (0 to 3), MXT416n (0 to 3), MXT2416n (0 to 3), DECIM, ADECIM, LPFIR1 to LPFIR4, BFCTT – Defines the physical memory size that contains 360 degrees of coefficient data for address patterns on the coefficient side of the execution unit. MEMSIZE permits a transform to be completed using a coefficient array of any size when MEMSIZE is greater or equal to N, and both MEMSIZE and N are a power of 2.

CHIPID Register

This 8-bit, read only register (DB) contains chip revision identification:

Bit[3:0] – Last digit for the year

Bit[7:4] – Number for the month

For example, CHIPID = 31H would indicate that the chip version was updated during March, 1991.

Host Interface

The host interface enables the LH9320 to be programmed in internal or external memory mode via the 8-bit data bus, DB[7:0] as was shown in Figure 3. In particular, the host decode enables the data to be written to a specified address location depending on the A1, A0 signals as shown in Table 6.

Table 6. Host Decoder Truth Table

A1	A0	MODE [0] = L INTERNAL MEMORY PROGRAMMING	MODE [0] = H EXTERNAL MEMORY PROGRAMMING
L	L	Data	Location 31 of program memory (1F)
L	H	Address register	Location 31 of pipeline/memory latency memory (3F)
H	L	Reserved	MODE register (A2)
H	H	Reserved	Reserved

Internal Memory Mode

For internal memory mode, the A1 and A0 signals generate the ADREN output of the host decode when A1 is LOW and A0 is HIGH (see Figure 3). The ADREN output is connected to the enable input of the address register. A1 and A0 also generate the DATAEN output of the host decode when A1 is LOW and A0 is LOW. The DATAEN output is connected to the enable input of the host memory. In this case, the host decode loads the LH9320 registers through a two step scheme: register address and register data. First, the DB[7:0] data bus points to the address of the register that is to receive the data, and then loads the data to the indicated address location.

External Memory Mode

In external memory mode, A1 and A0 permits the data to flow to the following predefined address locations:

- Location 31 of program memory (1F)
- Location 31 of pipeline/memory latency memory (3F)
- MODE register (A2)

In this case, the address register is bypassed and the DB[7:0] data bus transfers data to the predefined address (1F or 3F) as assigned by the A1 and A0 signals. In addition, the MODE register address location (A2) can be accessed externally to toggle back and forth between the internal and external memory modes. See Figure 3.

Note: because the address register is bypassed, programming in external memory mode requires only half the program length that would be utilized in internal memory mode.

ADDRESS CALCULATION

The address calculation section produces most of the LH9320 address patterns using the following logic:

- Opcode decode
- Seed
- Counter
- Datapath Control and Logic
- Terminal Count

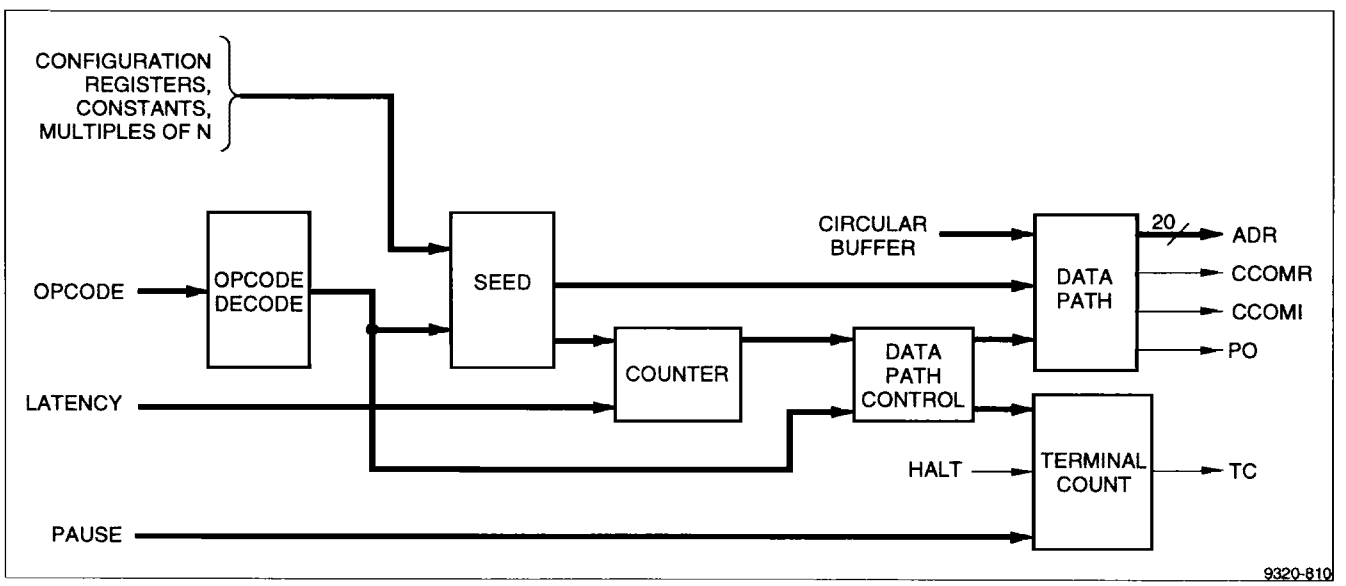


Figure 6. Address Calculation

Table 7. LH9320L Address Pattern/Register Cross Reference

MNEMONIC *	PIPELINE/MEMORY LATENCY		MODE [0]	MODE [1]	MODE [2]	MODE [3]	MODE [4]	MODE [5]		PAUSE	BREAKPOINT	HALT		CHAN. LENGTH/OVERLAP MEM.	NCHANNEL	MEMSIZE	OVERLAP	DISCARD	N	IFACTOR	DFACTOR	LEAP	NLEAP	INTERADR	ZEROPAD	ADRSTART	ADRLENGTH	ADRINC	ADRDEC	INDEXADR	INDEXINC	DIGITREV
FAST FOURIER TRANSFORMS (FFTs)																																
BF2n (0 to 19)	X			X		X		X		X	X	X							♦													
BF4n (0 to 9)	X			X		X		X		X	X	X							♦													
BF16n (0 to 4)	X			X		X		X		X	X	X							♦													
TF2n (0 to 19)	X			X		X		X		X	X	X				♦			♦													
TF4n (0 to 9)	X			X		X		X		X	X	X				♦			♦													
TF16n (0 to 4)	X			X		X		X		X	X	X				♦			♦													
MXB24n (0 to 8)	X			X		X		X		X	X	X							♦													
MXB216n (0 to 3)	X			X		X		X		X	X	X							♦													
MXB416n (0 to 3)	X			X		X		X		X	X	X							♦													
MXB2416n (0 to 3)	X			X		X		X		X	X	X							♦													
MXT24n (0 to 8)	X			X		X		X		X	X	X				♦			♦													
MXT216n (0 to 3)	X			X		X		X		X	X	X				♦			♦													
MXT416n (0 to 3)	X			X		X		X		X	X	X				♦			♦													
MXT2416n (0 to 3)	X			X		X		X		X	X	X				♦			♦													
RBF0	X			X		X		X		X	X	X							♦													♦
SEPARATION PASSES																																
BRFTL	X			X		X		X		X	X	X							♦													
BRFTLS	X			X		X		X		X	X	X							♦													
BRFTU	X			X		X		X		X	X	X							♦													
BRFTUS	X			X		X		X		X	X	X							♦													
BFCTL	X			X		X		X		X	X	X							♦													
BFCTT	X			X		X		X		X	X	X				♦			♦													
BFCTUS	X			X		X		X		X	X	X							♦													
BFCTU	X			X		X		X		X	X	X							♦											♦		
BFCTUP	X			X		X		X		X	X	X							♦											♦		
BFCTUEP	X			X		X		X		X	X	X							♦											♦		
BFCTULP	X			X		X		X		X	X	X							♦											♦		
DECIMATION																																
DECIM	X			X		X		X		X	X	X				♦			♦		♦											
ADECIM	X			X		X		X		X	X	X				♦			♦		♦	♦	♦									

* PCSTART and PCEND must be programmed for each execution.
X Programming is optional.
♦ Programming is required.
■ Used only on the first generated addressing sequence.

Table 7 (cont'd). LH9320L Address Pattern/Register Cross Reference

MNEMONIC *	PIPELINE/MEMORY LATENCY		MODE [0]	MODE [1]	MODE [2]	MODE [3]	MODE [4]	MODE [5]		PAUSE	BREAKPOINT	HALT		CHAN. LENGTH/OVERLAP MEM.	NCHANNEL	MEMSIZE	OVERLAP	DISCARD	N	IFACTOR	DFACTOR	LEAP	NLEAP	INTERADR	ZEROPAD	ADRSTART	ADRLENGTH	ADRINC	ADRDEC	INDEXADR	INDEXINC	DIGITREV
FINITE IMPULSE RESPONSE (FIR) FILTERS																																
LPFIR1	X			X		X		X		X	X	X				♦			♦													
LPFIR2	X			X		X		X		X	X	X				♦			♦													
LPFIR3	X			X		X		X		X	X	X				♦			♦													
LPFIR4	X			X		X		X		X	X	X				♦			♦													
GENERAL PURPOSE ADDRESSING/UTILITIES																																
MODINC	X			X		X		X		X	X	X														♦	♦	♦				
MODDEC	X			X		X		X		X	X	X														♦	♦		♦			
INTER	X			X		X		X		X	X	X							♦	♦				■						♦	♦	
INTERP	X			X		X		X		X	X	X							♦	♦				■						♦	♦	
INTEREP	X			X		X		X		X	X	X							♦	♦				■						♦	♦	
INTERLP	X			X		X		X		X	X	X							♦	♦				■						♦	♦	
PADHIGH	X			X		X		X		X	X	X													♦	♦	♦	♦		♦	♦	
PADHIGHP	X			X		X		X		X	X	X													♦	♦	♦	♦		♦	♦	
PADHIGHEP	X			X		X		X		X	X	X													♦	♦	♦	♦		♦	♦	
PADHIGHLP	X			X		X		X		X	X	X													♦	♦	♦	♦		♦	♦	
PADLOW	X			X		X		X		X	X	X													♦	♦	♦	♦		♦	♦	
PADLOWP	X			X		X		X		X	X	X													♦	♦	♦	♦		♦	♦	
PADLOWEP	X			X		X		X		X	X	X													♦	♦	♦	♦		♦	♦	
PADLOWLP	X			X		X		X		X	X	X													♦	♦	♦	♦		♦	♦	
OVERLAP	X			X		X		X		X	X	X					♦		♦													
DISCARD	X			X		X		X		X	X	X						♦	♦											♦	♦	
DISCARDP	X			X		X		X		X	X	X						♦	♦											♦	♦	
DISCARDEP	X			X		X		X		X	X	X						♦	♦											♦	♦	
DISCARDLP	X			X		X		X		X	X	X						♦	♦											♦	♦	
CMAG	X			X		X		X		X	X	X							♦													
INC	X			X		X		X		X	X	X														♦	♦	♦				
NOP	X			X		X		X		X	X	X							♦													
CIRCULAR BUFFER ADDRESSING																																
CBUFFIR					X	X						X		♦	♦	♦			♦													
CBUFFFT					X	X						X		♦	♦	♦			♦													

* PCSTART and PCEND must be programmed for each execution.
X Programming is optional.
♦ Programming is required.
■ Used only on the first generated addressing sequence.

ADDRESS CALCULATION (cont'd)

The address calculation section does not generate the circular buffer address patterns, CBUFFIR and CBUFFFT, as shown in Figure 6. Refer to *Multichannel Circular Buffer*.

The address calculation section decodes each address pattern and selects a set of seeds which, in turn, control the datapath and the counter. Based on the count initiated from the seeds, a particular dataflow within the datapath is selected to produce the desired addressing sequence.

Opcode Decode

The opcode is decoded in a straightforward manner. The decode outputs addresses that select a particular set of seeds for each address pattern.

Seed

The seed logic is essentially a large multiplexer as shown in Figure 7. The inputs for the multiplexer consist of the following:

- All powers of 2 up to 1 mega word; for example, 2^0, 2^1, 2^2, ... 2^{19}
- All values of N that are also divisible by powers of 2, such as $\left(\frac{N}{2^0}, \frac{N}{2^1}, \ldots \frac{N}{2^{19}}\right)$
- Configuration registers

The seeds are selected according to the address pattern and are sent to both the counter logic and the datapath logic.

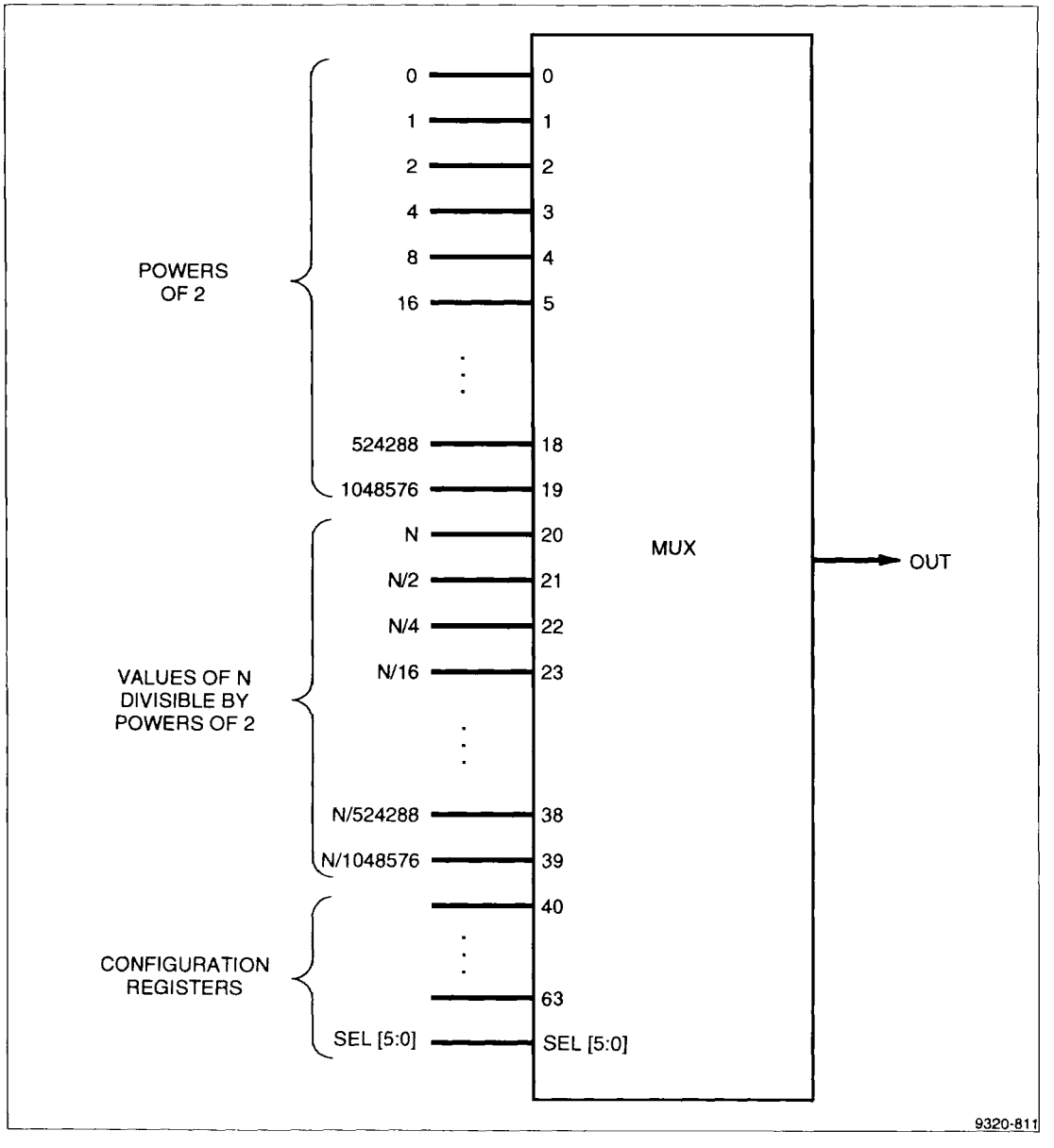


Figure 7. Seed Logic

Counter

The counter logic is set up to perform the following algorithm:

```
for (latency=0; latency<nlatency; latency++)
{
    /* do nothing */
}
for (loop1=0; loop1<nloop1; loop1++)
{
    /* do arithmetic operations */
    for (loop2=0; loop2<nloop2; loop2++)
    {
        /* do arithmetic operations */
        for (loop3a=0; loop3a<nloop3a;
            loop3a++)
        {
            /* do arithmetic operations */
        }
        for (loop3b=0; loop3b<nloop3b;
            loop3b++)
        {
            /* do arithmetic operations */
            for (loop4b=0; loop4b<nloop4b;
                loop4b++)
            {
                /* do arithmetic operations */
            }
        }
    }
}
```

Note: the complexity of the algorithm shown is determined by the radix-16 algorithm. All other address patterns are generated using subsets of the above algorithm.

Each of the counters generates three flags to indicate when a specific count ends, as shown in Figure 8. In turn, the counter gets decoded in the datapath control according to the specified address pattern and a datapath flow selected.

Datapath Control and Logic

The datapath control selects the dataflow for the datapath logic of each address pattern. The datapath logic is composed of a series of accumulators that perform arithmetic operations and manipulate data for a specified algorithm as shown in Figure 9. The datapath logic also performs additional special operations.

Specifically, the datapath contains digit reverse logic which translates a straight sequence into its digit reverse version.

To help solve testability issues, the address calculation logic contains a signature analyzer that includes a primitive polynomial of degree 31, as shown in Figure 10. The signals that are connected to the signature analyzer are ADR[19:0], CCOMR, CCOMI, PO, CCOMR enable, CCOMI enable, and PO enable.

Note: the LH9320 contains two instructions (opcodes) that are reserved for clearing and viewing the signature analyzer via the 20-bit address output. CLRSIG is the instruction for clearing the signature to zero. (The signature is always cleared after a reset.) VIEWSIG is the instruction that displays the signature contents since the last CLRSIG instruction.

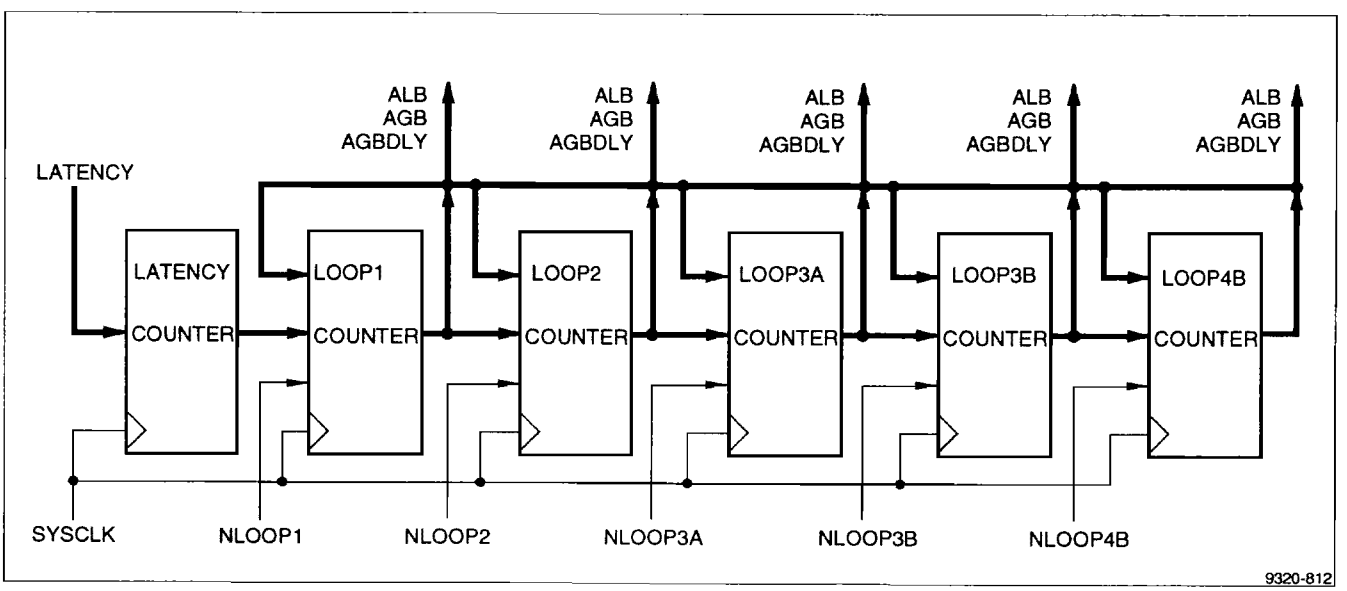


Figure 8. Counter Logic

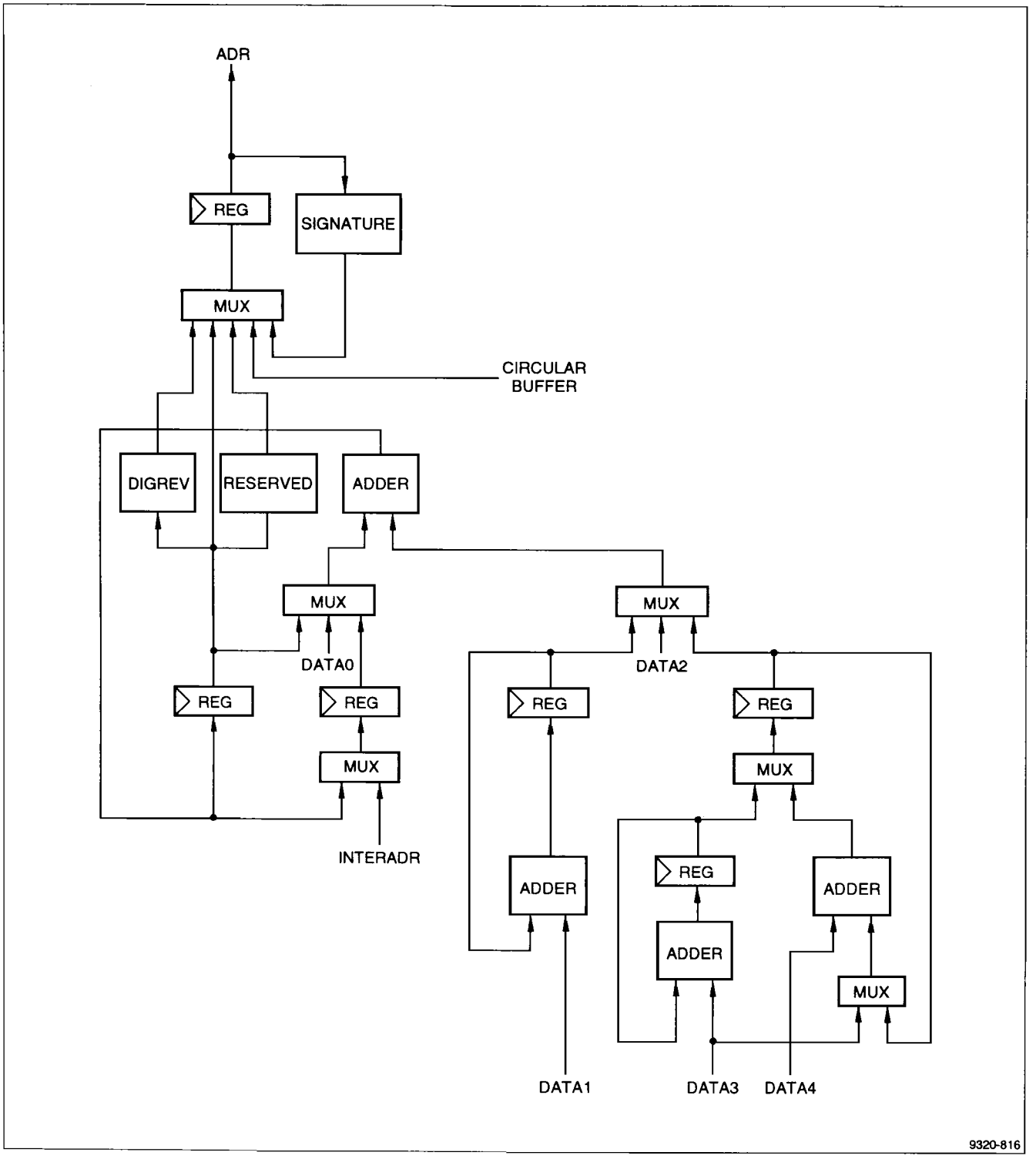


Figure 9. Datapath Logic

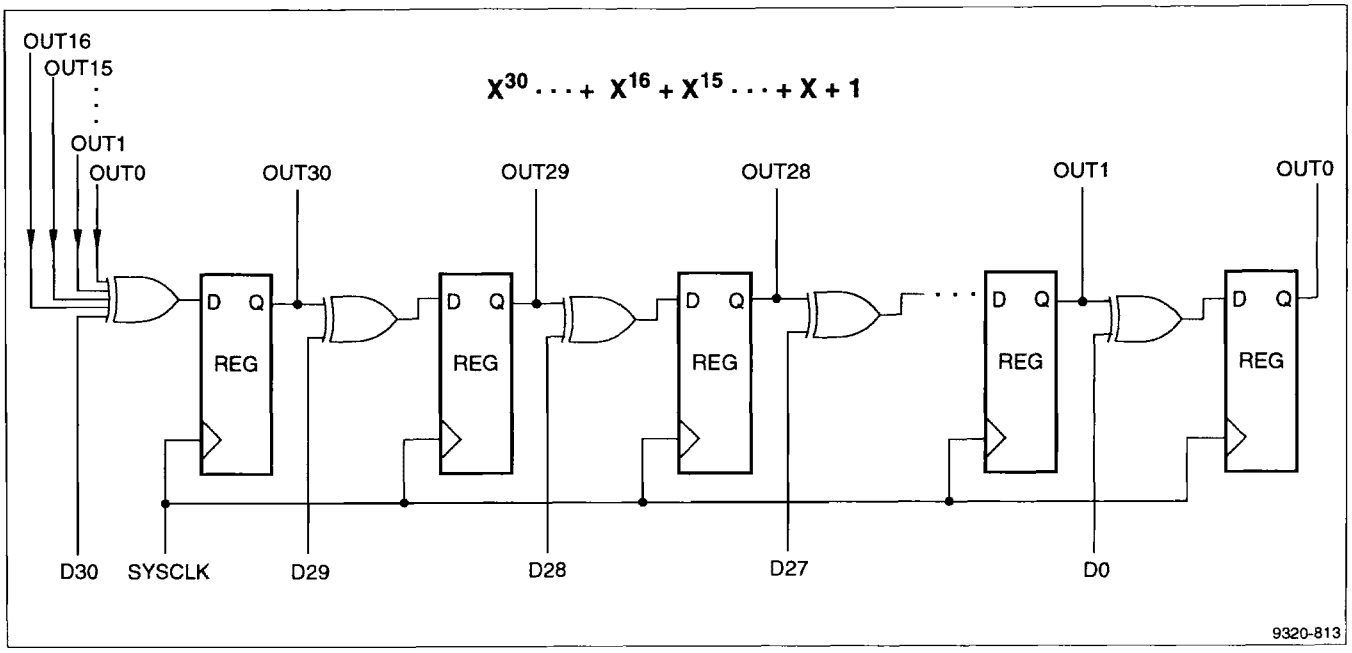


Figure 10. Signature Analyzer

Terminal Count

The terminal count (TC) logic as shown in Figure 11 produces the TC signal which indicates when an address pattern has been completed. As a default, TC goes HIGH one cycle before the current addressing sequence ends. TC remains HIGH until the next START signal causes a valid address to be generated. The low-to-high transition of TC can be delayed by one cycle (to actual end of sequence) by setting bit 3 of the MODE register to HIGH. Refer to *MODE Register* for additional information.

The end of the sequence can be delayed by an additional amount via the PAUSE register. If a pause is used, TC becomes active one cycle before sequence length + PAUSE.

CLOCK CONTROL

The clock control logic as shown in Figure 12 detects the rising edge of the START signal, which initiates the execution of the specified address pattern. The START signal causes the program counter to be incremented. Refer to *START (Start of a Pass) Signal.*

PROGRAM COUNTER

The program counter as shown in Figure 13 is a 5 bit counter whose starting address is given by the PCSTART register and ending address is given by the PCEND register. The next address is again started by PCSTART, and this process continues until the next reset occurs. The program counter is incremented by one each time the rising edge of the START signal is detected.

Note: for external memory mode, PCSTART and PCEND are bypassed and the data is transferred to the predefined address (1F, 3F, or A2) assigned by the A0 and A1 signals.

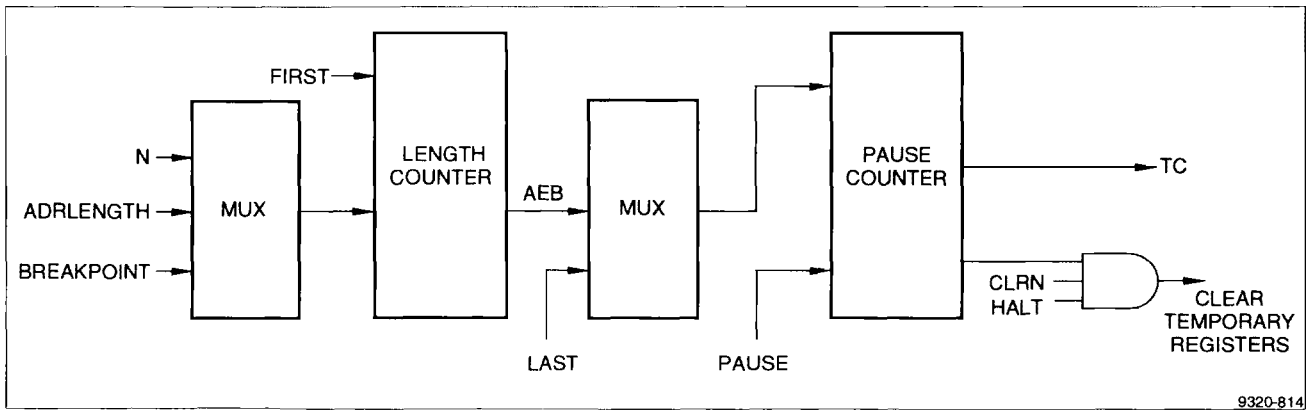


Figure 11. Terminal Count

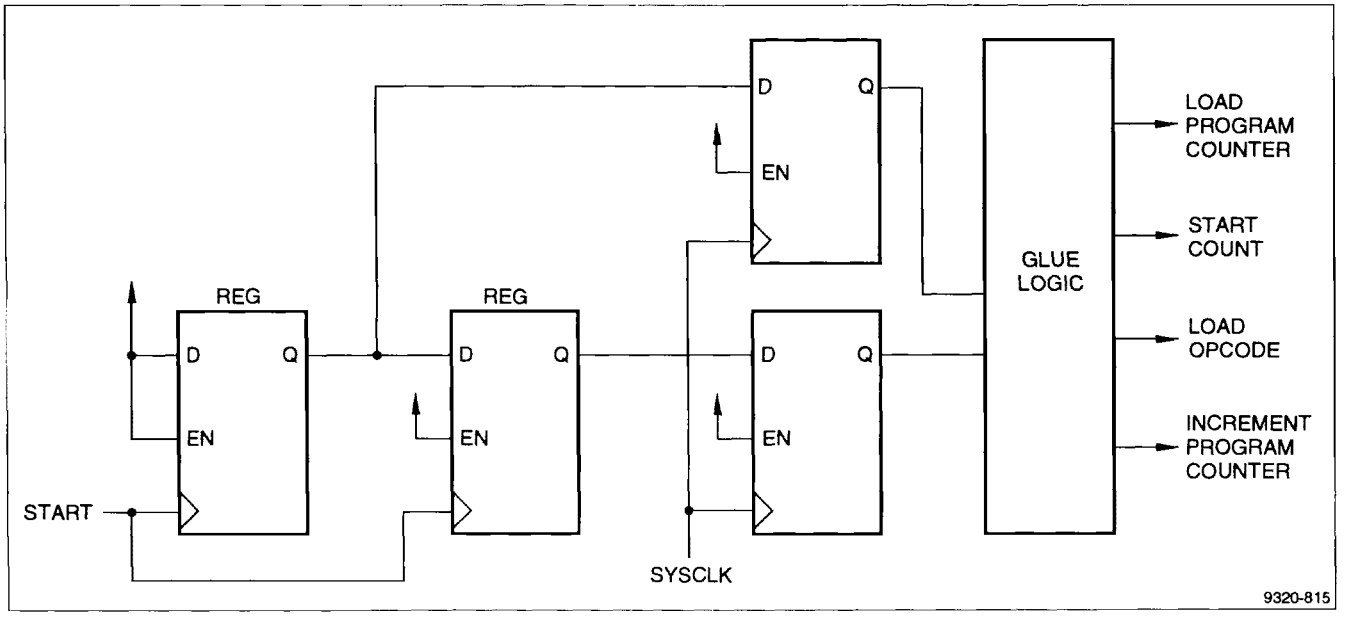


Figure 12. Clock Control

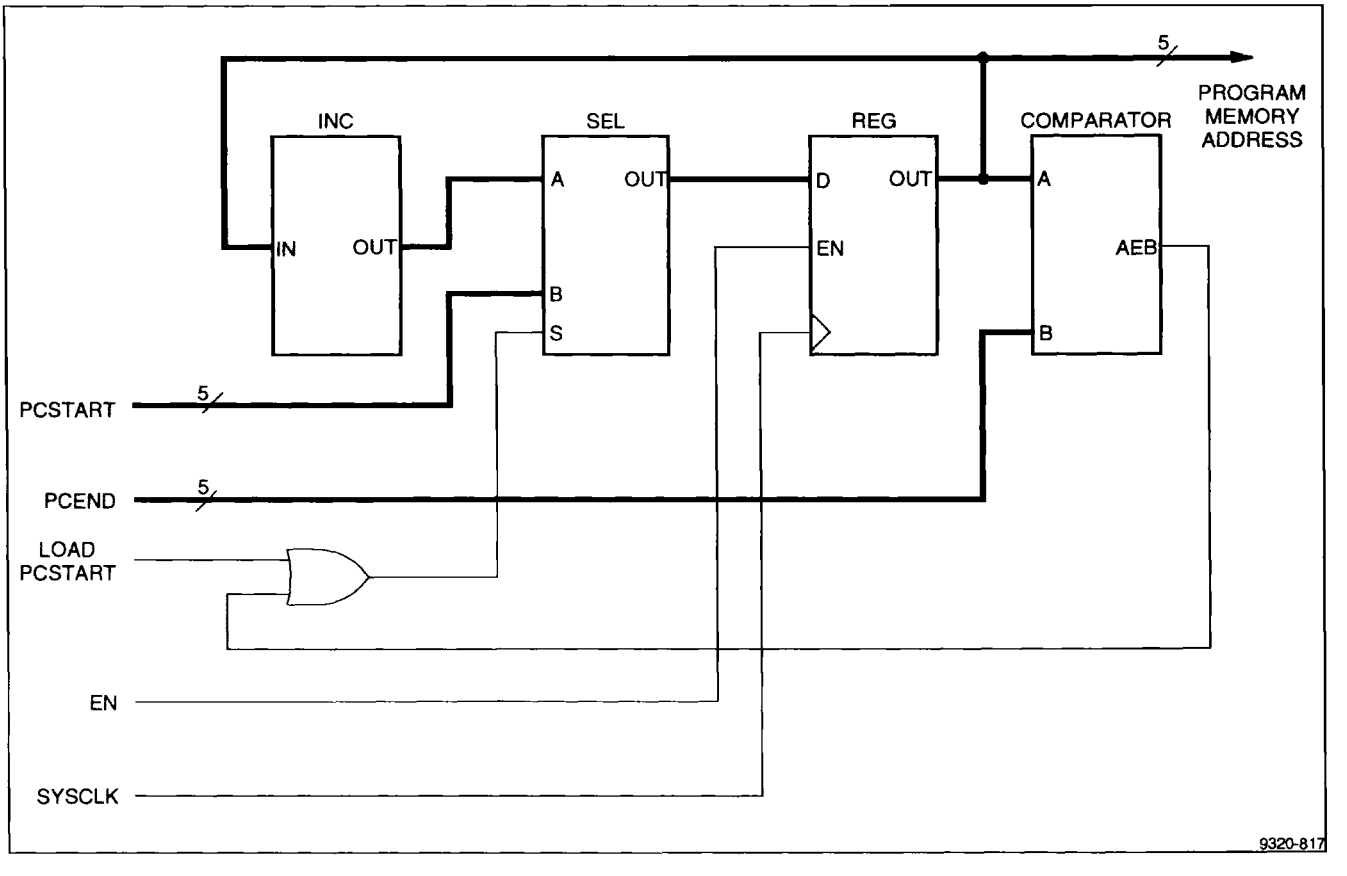


Figure 13. Program Counter

MULTICHANNEL CIRCULAR BUFFER

The LH9320 utilizes a unique circular-buffering technique for real-time DSP systems that use a multichannel system configuration. For example, circular buffering is used in algorithms for circular convolutions (if the input data from frame to frame is contiguous), FIR structures and time-domain correlations.

The LH9320 supports two types of circular buffer address patterns; CBUFFIR for multichannel FIR filter operations, and CBUFFFT for circular buffer FFT processing. The only difference between the two is the use of the channel length/overlap memory.

In CBUFFIR, the value in the channel length/overlap memory indicates the length of the addressing sequence for each channel. At the start of each sequence, CBUFFIR is decremented by one and then counts forward until the number of addresses generated is equal to the channel length. In CBUFFFT, the value in channel length/overlap memory indicates the overlap value, using N as the length of the addressing sequence generated. In this case, the start of each addressing sequence is decremented by the overlap amount and then counts forward until the number of addresses generated equals N.

The circular buffer block contains the following logic as shown in Figure 14:

- Sample arbitrator
- Pointer memory
- Channel length/overlap memory
- Circular buffer counter
- Decrementer
- Channel ID

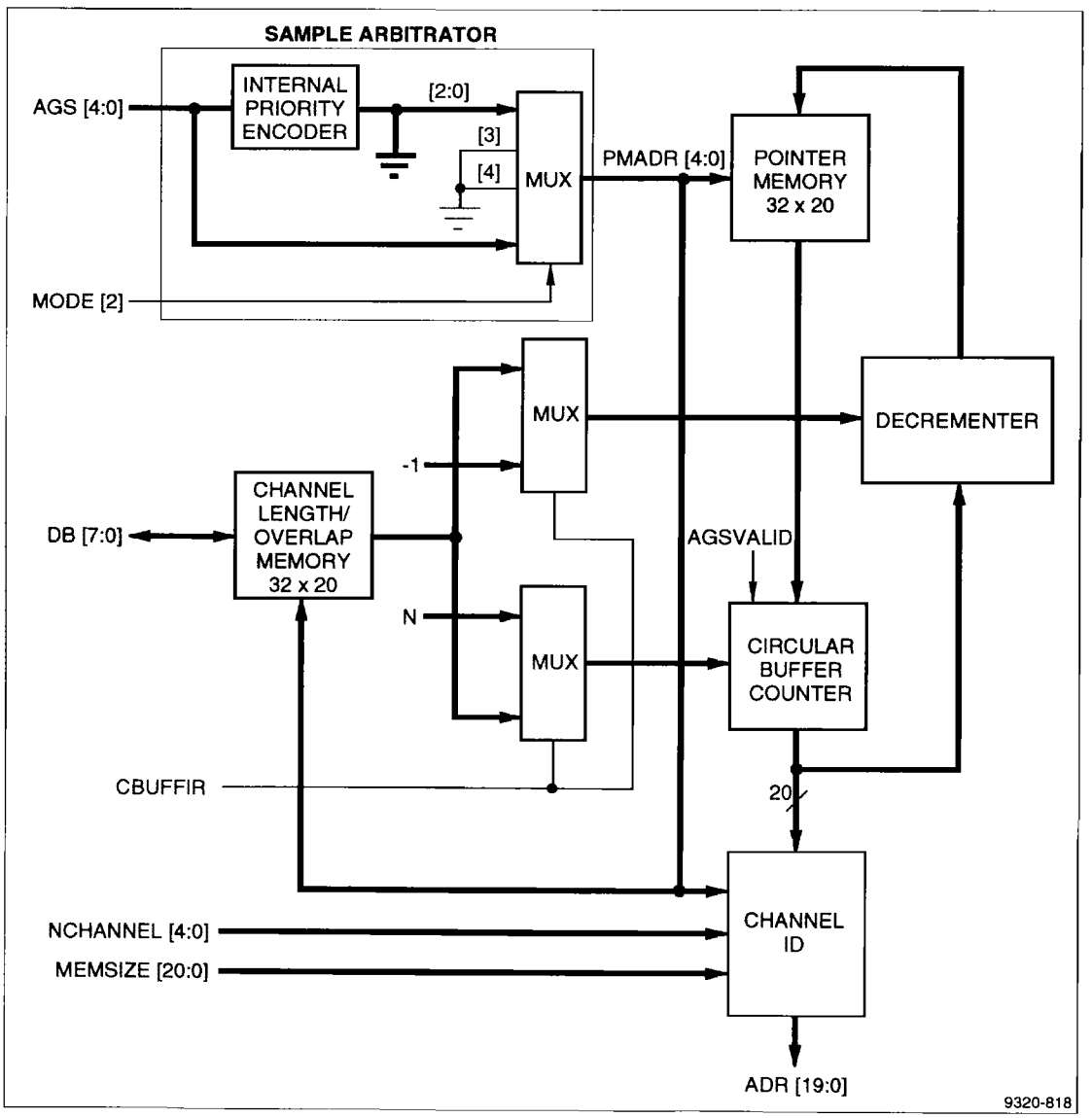


Figure 14. Multichannel Circular Buffer

MULTICHANNEL CIRCULAR BUFFER (cont'd)

The circular buffer generates an addressing sequence that originates from the 32 × 20 pointer memory which points to one of the channels defined by the NCHANNEL register. The selection of the channel depends on AGS[4:0] signals, which can handle a maximum of 32 encoded channels for external arbitration or up to five non-encoded channels for internal arbitration.

The output of the pointer memory contains the starting address for a particular channel. The ADR[19:0] outputs initially contain the starting address (plus the channel information). Then the ADR[19:0] outputs perform the circular buffer operation until the number of addresses generated reaches the value of the circular buffer length.

The (internal) starting address is then decremented by the specified overlap amount and is stored back in the same pointer file location. The circular buffer sequence continues indefinitely until the HALT register is set HIGH. At this time, all outputs are held at their previous values, except for $\overline{\text{MEMW}}$, $\overline{\text{MEMOE}}$ and RDCLK which change to HIGH, HIGH and LOW respectively. In turn, the LH9320 waits until the next START signal to execute the next address pattern.

The most significant bits of the output address (where the location of the most significant bit in ADR[19:0] is determined by MEMSIZE) are determined by the value in the NCHANNEL register. The most significant bits can be used to detect which circular buffer channel is currently being processed. Figure 15 shows how the user memory is partitioned for both the CBUFFIR and CBUFFFT address patterns.

Sample Arbitrator

The output of the sample arbitrator is the address to the pointer memory. Bit 2 of the MODE register controls whether the output of the sample arbitrator originates from the internal priority encoder (MODE bit 2 is LOW) or from the AGS[4:0] signals directly (MODE bit 2 is HIGH).

For internal arbitration, up to five nonencoded channels can be selected by the user. Table 8 shows how the internal arbitration is performed. The AGSVALID signal is only used in external arbitration.

Note: PMADR (pointer memory address) = 00 represents the highest priority. The AGSVALID input indicates the status of the AGS[4:0] signals. The AGS[4:0] are interpreted as valid channels if AGSVALID is HIGH. When AGSVALID is LOW, no active channels are represented by the AGS[4:0] signals.

For arbitration above five channels (MODE bit 2 is set HIGH), the internal priority encoder is bypassed. This permits the user to arbitrate externally using any type of priority encoder, such as the 74LS148, to increase the number of channels to a maximum of 32.

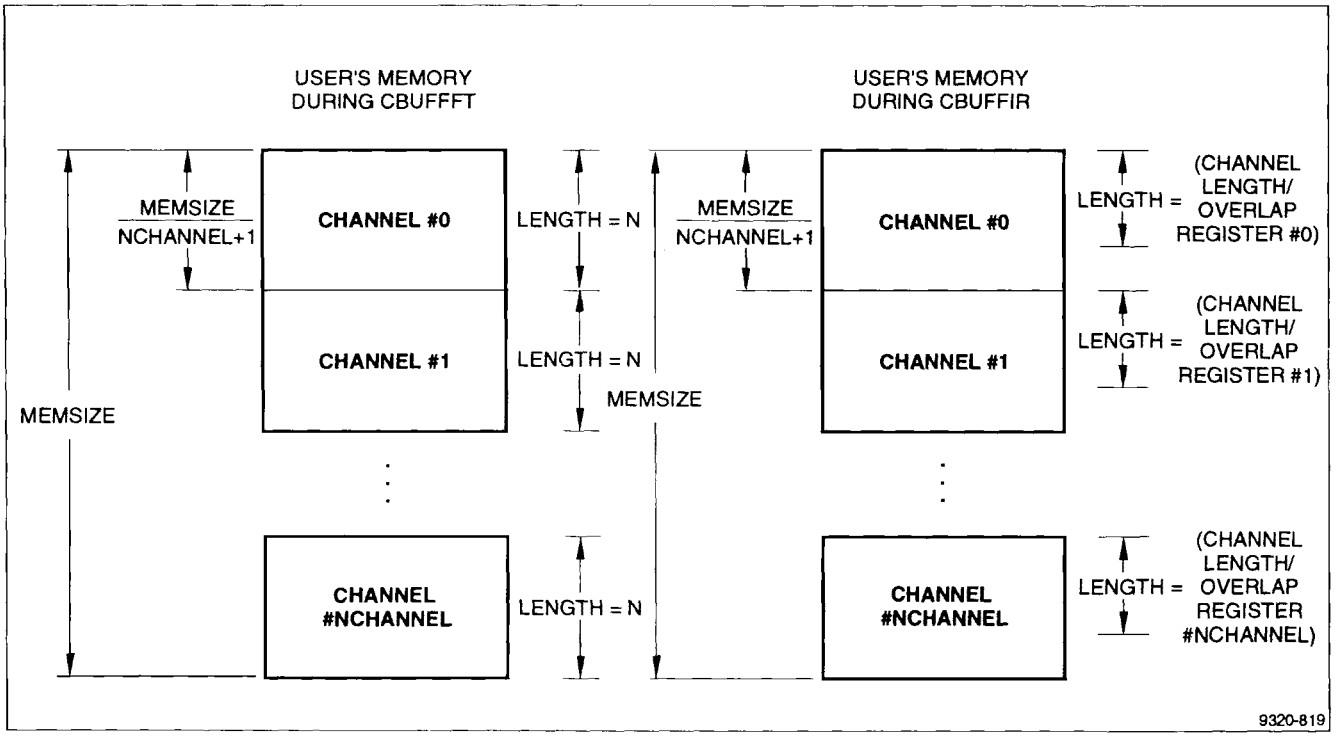


Figure 15. User Memory Partition for CBUFFIR and CBUFFFT

Table 8. Internal Arbitration*

AGS[4]	AGS[3]	AGS[2]	AGS[1]	AGS[0]	PMADR**[4]	PMADR[3]	PMADR[2]	PMADR[1]	PMADR[0]	CHANNEL
H	H	H	H	H	H	H	H	H	H	***
X	X	X	X	L	L	L	L	L	L	0
X	X	X	L	H	L	L	L	L	H	1
X	X	L	H	H	L	L	L	H	L	2
X	L	H	H	H	L	L	L	H	H	3
L	H	H	H	H	L	L	H	L	L	4

* H = High-logic level, L = Low-logic level, X = Irrelevant

** Pointer memory address (PMADR)

*** LH9320 is in idle mode, awaiting sample request.

Pointer Memory

The 32 × 20 pointer memory contains the starting address for each of the 32 possible channels in the circular buffer. The data input for the pointer memory is the same as the output after the overlap amount has been subtracted. In turn, this data input becomes the starting address the next time the specified channel is selected.

Channel Length/Overlap Memory

The 32 × 20 channel length/overlap memory is part of the host memory. The channel length/overlap memory contains either the channel length (CBUFFIR) or the overlap amount (CBUFFFT). As Figure 14 shows, the CBUFFIR address pattern selects the output of the channel length/overlap memory as the circular buffer length for the specified channel and a value of one for the decrement amount. CBUFFFT selects a circular buffer length equal to N and a decrement amount equal to the amount given by the channel length/overlap memory.

Circular Buffer Counter

The circular buffer counter is a 20-bit counter whose starting address is given by the output of the pointer memory. The length of the counter is the value given by the channel length/overlap memory (CBUFFIR) or by N (CBUFFFT). The increment amount for the counter is equal to one regardless of the type of address pattern, CBUFFIR or CBUFFFT.

Decrementer

The 20-bit decrementer subtracts the overlap amount from the starting address given by the pointer memory. The starting address is decremented by the amount given in the channel length/overlap memory (CBUFFFT) or by –1 (CBUFFIR).

$\overline{\text{MEMW}}$

The $\overline{\text{MEMW}}$ signal is inactive (HIGH) during all read sequences to the RAM except for the CBUFFIR and CBUFFFT instructions. During these instructions the $\overline{\text{MEMW}}$ signal is active on the first address of the sequence, and then inactive until the end. This can be used by external logic to gate data onto the data bus to be written to the RAM and the LH9124 at the same time.

Channel ID

The channel ID modifies the address generated by the circular buffer counter to point to the segment in memory mapped to a particular channel. Table 9 shows that the NCHANNEL register is part of the most significant bits of the address that is generated. These bits can be used to indicate which channel is accessed at any given time.

Note: the total number of partitions in memory must be equal to a power of two. For this reason, the number of partitions must either be greater or equal to the number of desired channels. For example, if five channels are desired, the user must partition the memory into eight channels (NCHANNEL = 07H), the closest power of two.

Table 9. MSBs of NCHANNEL Register

NCHANNEL REGISTER	NUMBER OF MSBs IN THE OUTPUT ADDRESS THAT FORM THE CHANNEL ID	MAX CHANNELS
00H	0	1
01H	1	2
03H	2	4
07H	3	8
0FH	4	16
1FH	5	32

Inactive Channel

Detection of an inactive channel will halt circular buffering when the currently running channel is done. Inactive channels are detected in external arbitration mode (mode bit 2 is HIGH) by AGSVALID set to HIGH. In internal arbitration mode (mode bit 2 is LOW), AGSVALID is ignored, and the invalid condition is detected by evaluating the AGS[4:0] inputs (see Table 7). An invalid channel is detected external to the chip by noting that TC remains HIGH when the currently running channel is done. When a valid channel is finally detected, TC will drop LOW, as it normally does at the start of a new circular buffer sequence.

USING THE LH9320

INITIALIZING THE LH9320

The AG begins initialization when RESET is set to HIGH. All the internal counters and registers are reset to initial conditions. In order to reset all internal counters and registers, the RESET signal should be HIGH for at least two cycles of R/$\overline{W}$ and one cycle of SYSCLK inputs. (R/$\overline{W}$ clocks the host registers and SYSCLK is for all other registers.)

After the AG is reset, all internal mode and configuration registers must be loaded with the appropriate system information if different than the reset values (see the next section, Programming the LH9320). Internal programmable memory and configuration registers only need to be set if the user's program needs them.

The LH9320 can be considered a simple decoded system peripheral. When used in conjunction with the LH9124 DSP, the AG can be initialized by the LH9124's scheduler. When the AG's Chip Select ($\overline{CS}$) signal is asserted LOW, the scheduler selects the AG. Internally, the AG connects DB0-DB7 to the internal register fields and program memory.

There are two levels of decode within the AG. To load any memory or register location the user must first load the internal address register with the location of the desired 8-bit field to be written, then load the internal data register with the value.

STARTING EXECUTION

Address generation begins when the START signal is asserted HIGH. All mode, memory, and configuration information must be properly loaded before the START signal is issued. For the first start after a reset, there will be five pipeline latencies. However, if START is asserted HIGH at least (4) four cycles before the terminal count (TC) of the current address pattern (TC goes HIGH one cycle before the end of the sequence), the AG completes calculating the current addressing sequence and immediately begins processing the next address pattern code without any pipeline latency.

When the LH9320 is configured with the scheduler, START is usually generated by the scheduler and it must be synchronous to the system clock (SYSCLK). For other systems, START may be derived from the AG's TC signal (except for very first START signal). For additional information, refer to the next section, Programming the LH9320.

STOPPING EXECUTION

The HALT register can be set to '1' by the scheduler at any time during AG execution. When this bit is set to '1,' the next system clock will cause the AG execution to stop immediately. The START signal will be disabled for as long as the halt bit is set HIGH. Once the halt bit is set LOW and a new start is asserted the user must wait five pipeline latencies before the first valid address is produced.

The AG will stop generating address outputs and stay at the last generated address state when the programmed address pattern is finished. The AG then enters a waiting state. The number of cycles that the AG waits depends on the value in the PAUSE register. The PAUSE register is a 20-bit register, which means that the AG can be paused up to 1mega word cycles.

TC will go HIGH one cycle before the end of each sequence, and will stay HIGH until the next START signal causes a valid address to be produced. The low-to-high transition of TC can be delayed by one cycle by setting MODE[3] to one. TC will then go HIGH with the last address. TC will always go LOW when the first valid address is produced.

PROGRAMMING THE LH9320

The LH9320 can be programmed in conjunction with a scheduler. The AG is similar to a programmable peripheral device with minimized decoding through a two step address scheme.

INTERFACE BETWEEN THE SCHEDULER AND THE AG

Programming the AG involves loading the address patterns and latency codes into memory and then configuring the control registers. Loading is done by the scheduler. Table 6 shows how the two pins of the LH9320, A0 and A1, select the address and data registers. The user activates the AG's internal interface by asserting $\overline{CS}$ LOW and activating the A0 and A1 signals with the appropriate logic levels to select one of the internal interface registers.

Figure 3 shows the A0 and A1 signals controlling both the address register and the data register. All the internal memory shown is accessed through the 8-bit data bus, DB0-DB7. The internal memory is separated into the four areas: program memory, multichannel circular buffer lengths/overlaps, pipeline and memory latency, and configuration registers.

Each AG has separate lines to the scheduler for $\overline{CS}$ and TC. This allows each AG to be programmed individually by asserting $\overline{CS}$ LOW. All AGs share a common read and write (R/$\overline{W}$) line. Whether R/$\overline{W}$ is HIGH or LOW, only the AG with $\overline{CS}$ asserted LOW is activated and the data is read to or written from the AG through the common data bus.

The address pattern code is programmed into the AG by the user. The simple programming procedure consists of loading the address pattern code in the AG memory and setting the AG registers. The AG has two modes, internal memory mode and external memory mode. In internal memory mode the user programs the address pattern codes in the AG's internal memory (program memory). The program memory can handle address codes of up to 32 patterns. In external memory mode the user programs the address pattern code into external memory. This mode is useful when the 32-word space in the local memory is not sufficient for processing. The address pattern code in external memory is accessed by the AG through the data bus, DB0-DB7. In both internal and external memory mode, the user must set the appropriate values in all control registers needed by the program.

ABSOLUTE MAXIMUM RATINGS [1,2]

PARAMETER	RATING
Supply Voltage to VSS Potential	−0.5 V to 7 V
Signal Pin Voltage to VSS Potential	−0.5 V to VDD + 0.5 V (not to exceed 7 V)
DC Output Current [3]	± 50 mA
Storage Temperature Range	−65°C to 150°C
Power Dissipation (Package Limit)	2.0 W
DC Voltage Applied to Outputs in High-Z State	−0.5 V to VDD + 0.5 V (not to exceed 7 V)

OPERATING RANGE [1]

SYMBOL	DESCRIPTION	MIN	MAX	UNIT
T_A	Temperature, Ambient	0.0	70	°C
VDD	Supply Voltage	4.75	5.25	V
VSS	Supply Voltage	0.0	0.0	V
V_{IL}	Logic '0' Input Voltage [5]	−0.5	0.8	V
V_{IH}	Logic '1' Input High Voltage	2.2	VDD + 0.5	V

DC ELECTRICAL CHARACTERISTICS [1] (Over Operating Range)

SYMBOL	DESCRIPTION	TEST CONDITIONS	MIN	MAX	UNIT
I_{LPUI} [6]	Pull-Up Input Leakage Current	VDD = 5.25 V, V_{IN} = 0 V	−70	−15	µA
		V_{IN} = VDD	−10	10	
I_{LPDI} [7]	Pull-Down Input Leakage Current	VDD = 5.25 V, V_{IN} = VDD	15	70	
		V_{IN} = VSS	−10	10	
I_{LI}	Input Leakage Current	VDD = 5.25 V, V_{IN} = 0 V to VDD	−10	10	
V_{OH}	Output High Voltage	I_{OH} = −6.0 mA	2.4		V
V_{OL}	Output Low Voltage	I_{OL} = 6.0 mA		0.4	V
I_{DD}	Average Supply Current	Measured at t_{CYC} (40 MHz)		250	mA
I_{DD2}	Average Standby Current [8]	All inputs = V_{IH} (2.2 V), except AGSVALID = V_{IL} (0.8 V). 2 SYSCLKS occur to enable RESET of CCOMR & CCOMI. Excludes output load current & RPROG current.		20	mA
I_{DD3}	Quiescent Standby Current	All inputs = VDD (−0.2 V), except AGSVALID = VSS (0.2 V). 2 SYSCLKS occur to enable RESET of CCOMR & CCOMI. Excludes output load current & RPROG current.		1	mA

See notes on Page 29.

AC TEST CONDITIONS [1]

PARAMETER	RATING
Input Pulse Levels	VSS to 3 V
Input Rise & Fall Times (10% to 90%)	3.0 ns (Figure 16a)
Input Timing Reference Levels	1.5 V
Output Reference Levels	1.5 V
Output Load, Timing Tests	Figure 16b

CAPACITANCE [1,4]

SYMBOL	DESCRIPTION	TEST CONDITIONS	RATING
C_{IN}	Input Capacitance	T_A = 25°C, F = 1MHz, V_{DD} = 4.75 V	10 pF
C_{OUT}	Output Capacitance	T_A = 25°C, F = 1MHz, V_{DD} = 4.75 V	10 pF

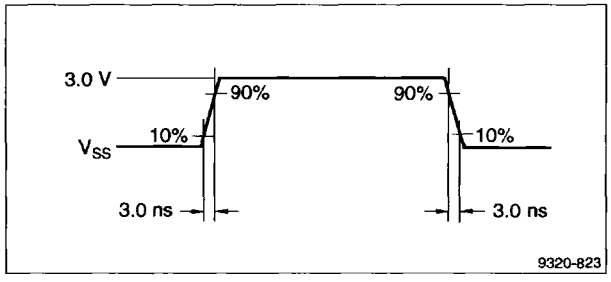


Figure 16a. Input Rise and Fall Times

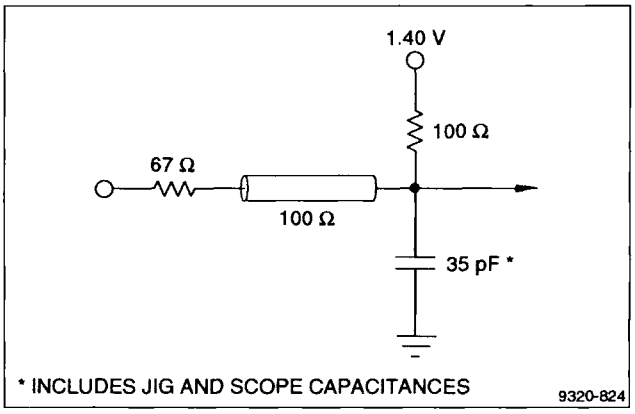


Figure 16b. Output Load Circuit

NOTES:

1. All voltages are measured with respect to VSS.
2. Stresses greater than those listed under 'Absolute Maximum Ratings' may cause permanent damage to the device. This is a stress rating for transient conditions only. Functional operation of the device at these or any other conditions above those indicated in the 'Operating Range' of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.
3. Outputs should not be shorted for more than 30 seconds. No more than one output should be shorted at any time.
4. Sample tested only.
5. LH9320 inputs are able to withstand a −1.5 V undershoot for less than 10 ns per cycle.
6. An internal pull-up resistor is attached to DB[7:0] and AGS[4:0] pins.
7. An internal pull-down resistor is attached to the CCOMR, CCOMI, and AGSVALID pins.
8. I_{DD} is dependent upon actual output loading and cycle rates. Specified values are with outputs open.

AC ELECTRICAL CHARACTERISTICS

SIGNAL	DESCRIPTION	MIN	TYP	MAX	UNITS
t_{CYC}	SYSCLK Cycle Time	25			ns
t_{CLSY}	Clock Low Time (SYSCLK)	10		15	ns
t_{CHSY}	Clock High Time (SYSCLK)	10		15	ns
t_{ODRD}	RDCLK, Output Delay Time (SYSCLK)	2 *	5	10	ns
t_{OHMW}	MEMWN, Rising Edge, Output Delay Time (SYSCLK)		5	9	ns
t_{OLMW}	MEMWN, Falling Edge, Output Delay Time (SYSCLK)		5	10	ns
t_{ODADR}	ADR[19:0] Output Valid Delay Time (SYSCLK)		16	20	ns
$t_{RDSKADRX}$	RDCLK to ADR[19:0] Invalid Rising Edge Skew	1	3		ns
t_{ODMOE}	MEMOEN, Output Delay Time (SYSCLK)		13	18	ns
t_{ODTC}	TC, Rising Edge, Output Delay Time (SYSCLK)		16	20	ns
t_{ODPO}	PO, Rising Edge, Output Delay Time (SYSCLK)		17	20	ns
t_{OHZPO}	PO, High Impedence, Output Delay Time (SYSCLK) *			15	ns
t_{OLZPO}	PO, Low Impedence, Output Delay Time (SYSCLK) *			15	ns
$t_{ODCCOMR}$	CCOMR, Rising Edge, Output Delay Time (SYSCLK)		17	20	ns
$t_{OHZCCOMR}$	CCOMR, High Impedance, Output Delay Time (SYSCLK) *			15	ns
$t_{OLZCCOMR}$	CCOMR, Low Impedance, Output Delay Time (SYSCLK) *			15	ns
$t_{ODCCOMI}$	CCOMI, Rising Edge, Output Delay Time (SYSCLK)		17	20	ns
$t_{OHZCCOMI}$	CCOMI, High Impedance, Output Delay Time (SYSCLK) *			15	ns
$t_{OLZCCOMI}$	CCOMI, Low Impedance, Output Delay Time (SYSCLK) *			15	ns
t_{SAGS}	AGS[4:0] Setup Time (SYSCLK)	7	4		ns
t_{HAGS}	AGS[4:0] Hold Time (SYSCLK)			0	ns
t_{SAGSV}	AGSVALID Setup Time (SYSCLK)	7	4		ns
t_{HAGSV}	AGSVALID Hold Time (SYSCLK)			0	ns
t_{SSTART}	START Setup Time (SYSCLK)	7	4		ns
t_{HSTART}	START Hold Time (SYSCLK)			0	ns
t_{LRW}	RWN Low Time	25	15		ns
t_{HRW}	RWN High Time	50	40		ns
t_{RWSCSN}	CSN Setup Time (RWN)	10			ns
t_{RWHCSN}	CSN Hold Time (RWN)			0	ns
t_{RWSDB}	DB[7:0] Setup Time (RWN)	10	7		ns
t_{RWHDB}	DB[7:0] Hold Time (RWN)			0	ns
t_{RWODDB}	DB[7:0] Output Delay (RWN)	45	38		ns
t_{RWLZDB}	DB[7:0] Output Delay (RWN) High to DB[7:0] Low-Z *			15	ns
t_{RWHZDB}	DB[7:0] Output Time (RWN) Low to DB[7:0] High-Z *			15	ns
$t_{RWSA1A0}$	A1A0 Setup Time (RWN)	16	13		ns
$t_{RWHA1A0}$	A1A0 Hold Time (RWN)			0	ns
t_{SRESET}	RESET Setup Time (SYSCLK)	11	9		ns
t_{HRESET}	RESET Hold Time (SYSCLK)			0	ns
$t_{RWSRESET}$	RESET Setup Time (RWN)	11	9		ns
$t_{RWHRESET}$	RESET Hold Time (RWN)			0	ns
$t_{CSNHZDB}$	CSN High to DB[7:0] High-Z *			15	ns
$t_{CSNLZDB}$	CSN Low to DB[7:0] Low-Z *			15	ns
t_{ODMOEH}	MEMOEN, Output Delay, Rising Edge	5 *	13		ns

* Specification guaranteed by design.

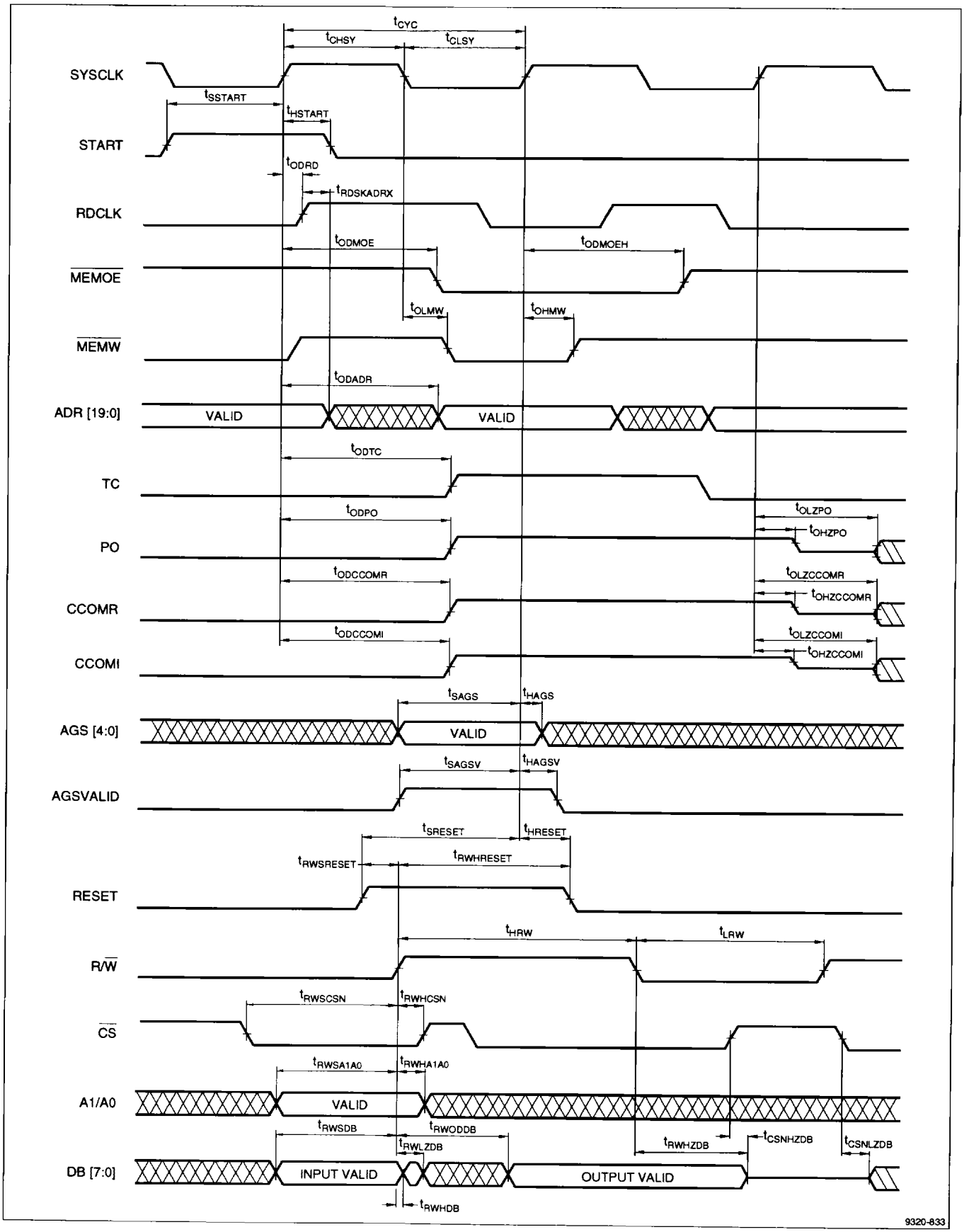


Figure 17. Timing Diagram

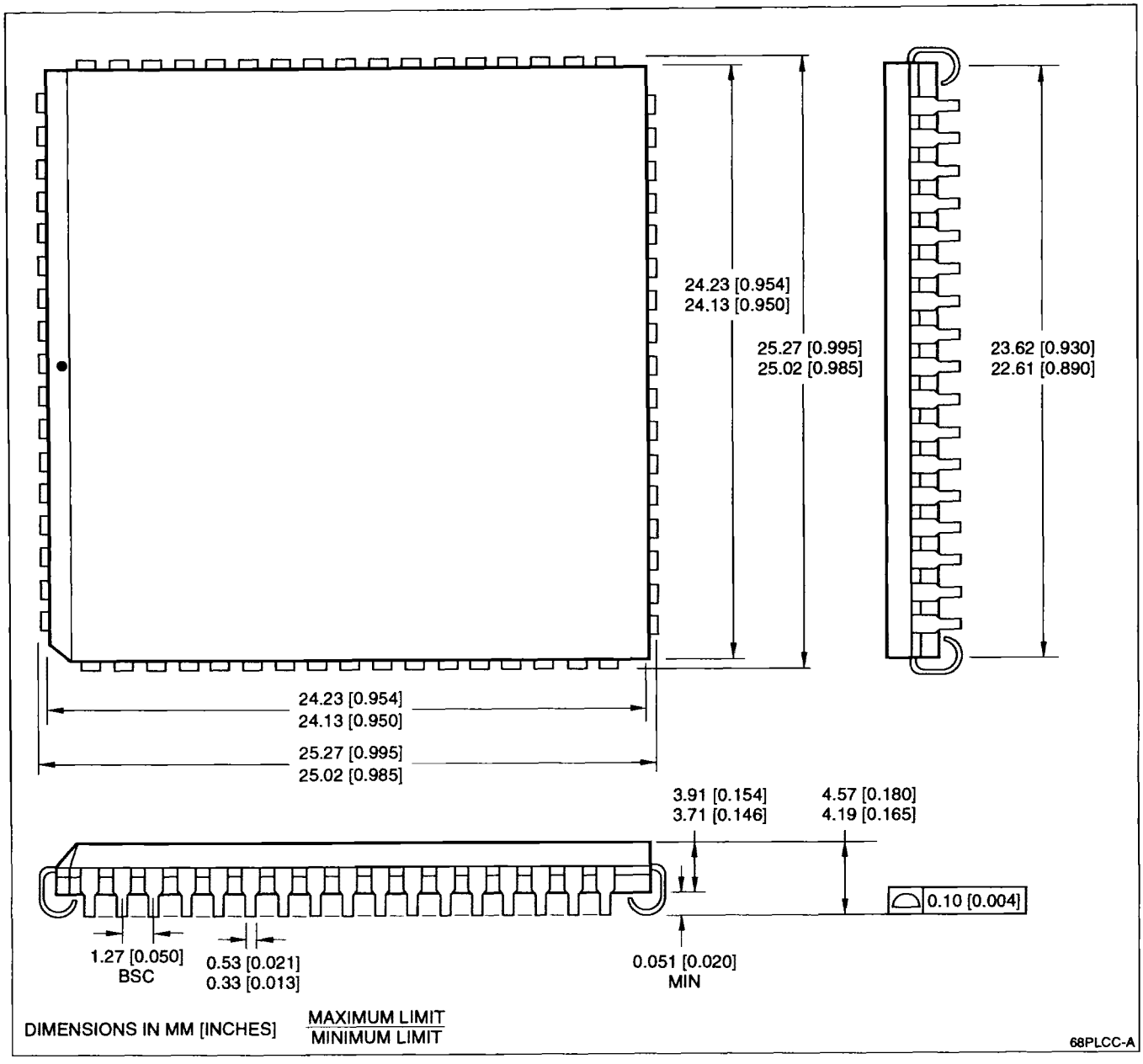
PIN LIST

PIN	SIGNAL
1	VSS
2	ADR16
3	ADR17
4	ADR18
5	AD419
6	VDD
7	VSS
8	PO
9	TC
10	CCOMR
11	CCOMI
12	VSS
13	RDCLK
14	VDD
15	VSS
16	$\overline{\text{MEMOE}}$
17	$\overline{\text{MEMW}}$
18	VSS
19	RPROG
20	VDD
21	SYSCLK
22	R/$\overline{\text{W}}$
23	START

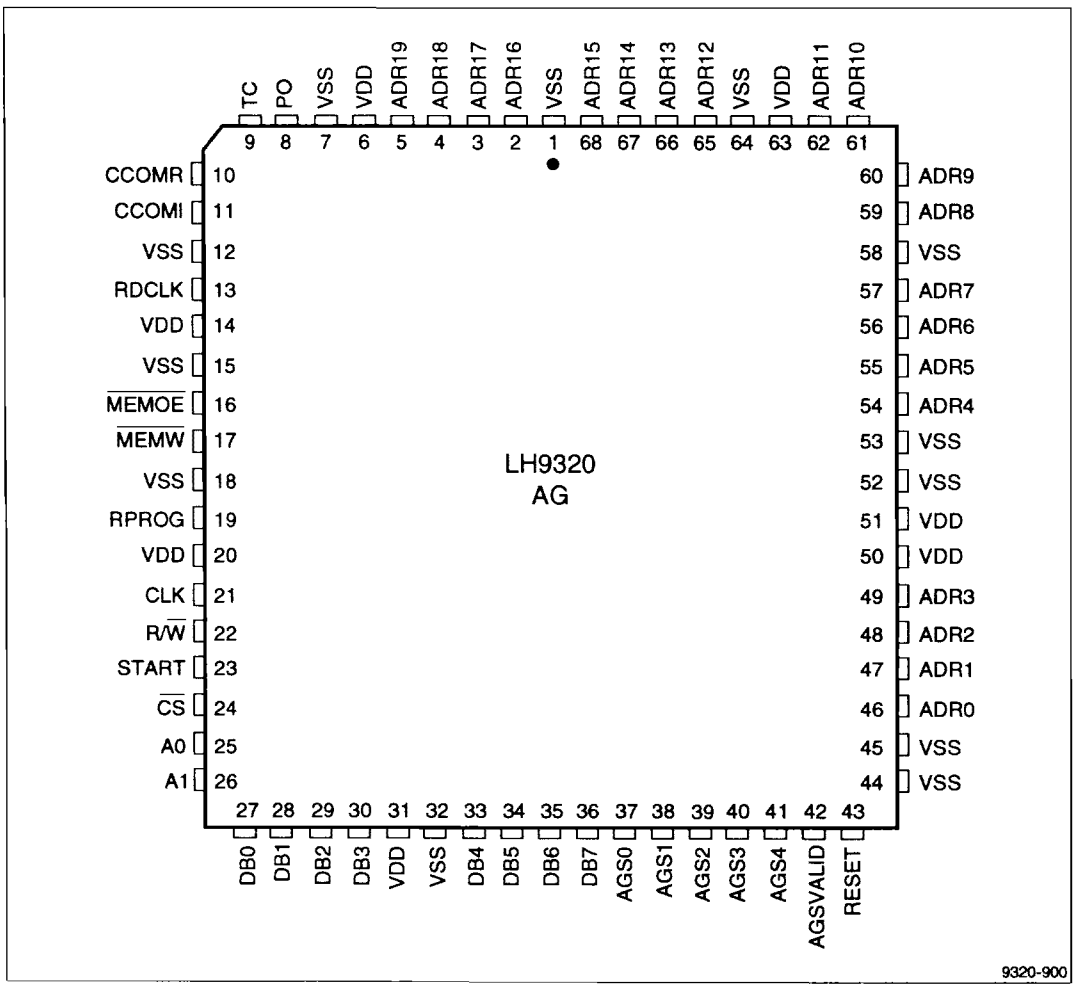
PIN	SIGNAL
24	$\overline{\text{CS}}$
25	A0
26	A1
27	DB0
28	DB1
29	DB2
30	DB3
31	VDD
32	VSS
33	DB4
34	DB5
35	DB6
36	DB7
37	AGS0
38	AGS1
39	AGS2
40	AGS3
41	AGS4
42	AGSVALID
43	RESET
44	VSS
45	VSS
46	ADR0

PIN	SIGNAL
47	ADR1
48	ADR2
49	ADR3
50	VDD
51	VDD
52	VSS
53	VSS
54	ADR4
55	ADR5
56	ADR6
57	ADR7
58	VSS
59	ADR8
60	ADR9
61	ADR10
62	ADR11
63	VDD
64	VSS
65	ADR12
66	ADR13
67	ADR14
68	ADR15

PACKAGE DIAGRAM: 68-PIN, PLCC

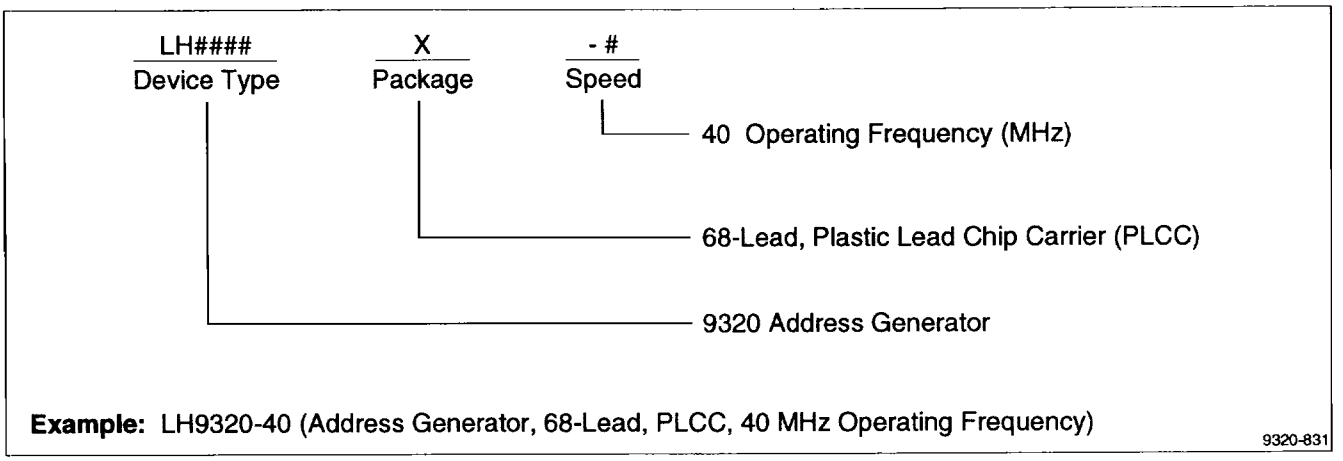


PACKAGE DRAWING: 68-PIN PLCC PIN CONNECTIONS



NOTES

ORDERING INFORMATION



SHARP

NORTH AMERICA

SHARP Electronics Corporation
Microelectronics Group
5700 NW Pacific Rim Blvd., M/S 20
Camas, WA 98607, U.S.A.
Phone: (206) 834-2500
Telex: 49608472 (SHARPCAM)
Facsimile: (206) 834-8903

EUROPE

SHARP Electronics (Europe) GmbH
Microelectronics Division
Sonninstraße 3
20097 Hamburg, Germany
Phone: (49) 40 2376-2286
Telex: 2161867 (HEEG D)
Facsimile: (49) 40 2376-2232

ASIA

SHARP Corporation
Integrated Circuits Group
2613-1 Ichinomoto-Cho
Tenri City, Nara 632, Japan
Phone: (07436) 5-1321
Telex: LABOMETA-B J63428
Facsimile: (07436) 5-1532

Reference SMT90062 © SHARP Corp. NOV92 Printed and bound in the U.S.A.
Rev A, 9/30/92; Rev. B, 6/16/93