NCR Product Design Series

VS1500 ViGen Cell Data Sheets

Release 3.0
February 1990

ViGen
is a trademark of NCR Corporation

NCR 1.5μ ViGen LIBRARY DATA SHEETS
ALPHABETICAL CONTENTS

VS1500 ALU

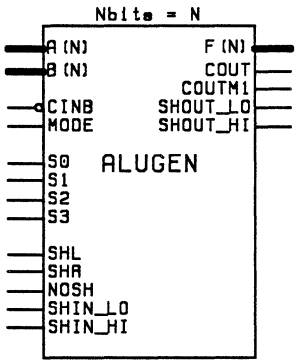
GENERAL DESCRIPTION

- Performs the same 16 logical and 16 arithmetic operations as the TTL '181 ALU
- Static operation (no clocks)
- Carry look-ahead is internally generated
- Post-shift option allows a logical or arithmetic function plus a one-bit left or right shift in a single operation
- Selectable 2- through 32-bit input and output data buses

This generator builds a fast, static ALU that is based on the TTL '181 ALU. The sixteen arithmetic and sixteen logical functions provided are the same as those of the '181. The functions are selected by four function selector lines (S3, S2, S1, and S0), the MODE input (low = arithmetic, high = logical) and the active-low carry in (CINB).

SYMBOL

The symbol for ALUGEN will be unique for each configuration. An example is given here only for reference.



INPUT PARAMETER RANGES

INPUT PARAMETER	ALLOWED RANGE	EXPLANATION
Nbits	2 – 32 even only	Number of bits in the input and output data buses.
Shift	0 or 1	Post-shift option (1=yes, 0=no).

ALUGEN

ViGen CONFIGURABLE FUNCTION

DIFFERENCES BETWEEN ALUGEN AND THE TTL '181 ALU

- Carry out is provided from the two most significant bits in ALUGEN, making overflow detection easy. The '181 provides carry out only from the most significant bit. These active high carry outs will remain high for all logical mode operations.
- A post shift option is provided in ALUGEN. When enabled, extra control lines allow selection of NOSH (no shift), SHL (shift left by one bit), or SHR (shift right by one bit). This shift is done on the result of whatever function is selected in normal operation, making it possible to do an arithmetic or logical operation followed by a one-bit shift in one cycle. Deselecting all of the shift control inputs will tristate the output bus. Serial inputs and outputs are provided for both the least and most significant bits.
- The P/ and G/ outputs of the '181, used for carry look-ahead, are not provided and are not needed, since carry look-ahead is handled internally to ALUGEN. Standard '181s are extended to larger data words by cascading devices, and if carry look-ahead is desired, it must be done externally using '182s. ALUGEN accepts any data word size (even only) as input and builds its own internal custom carry look-ahead.
- The A=B output of the '181 is not provided in ALUGEN.

INPUTS/OUTPUTS

Definitions of the ALUGEN inputs and outputs are given in the following table. Input and output pin names are listed in pin-number sequence.

PIN NAME	DEFINITION	REQ/ OPT	CAP (pF)
INPUTS:			
A (N)	A input bus	REQ	0.326
B (N)	B input bus	REQ	0.290
CINB	Carry input (active low)	REQ	0.058
MODE	Mode control (High = logical, Low = arithmetic)	REQ	0.054
S0	Function select input 0	REQ	0.055
S1	Function select input 1	REQ	0.055
S2	Function select input 2	REQ	0.055
S3	Function select input 3	REQ	0.055
SHL	Shift left by one bit control (Notes 1, 2)	OPT	0.255
SHR	Shift right by one bit control (Notes 1, 2)	OPT	0.255
NOSH	No shift (Note 1)	OPT	0.255
SHIN_LO	Shift in to least significant bit	OPT	0.118
SHIN_HI	Shift in to most significant bit	OPT	0.118
OUTPUTS:			
F (N)	Output data bus	REQ	0.230
COUT	Carry out (active high)	REQ	
COUTM1	Carry out from second most significant bit (active high)	REQ	
SHOUT_LO	Shift out from least significant bit (=F (0))	OPT	0.118
SHOUT_HI	Shift out from most significant bit (=F (Nbits-1))	OPT	0.118

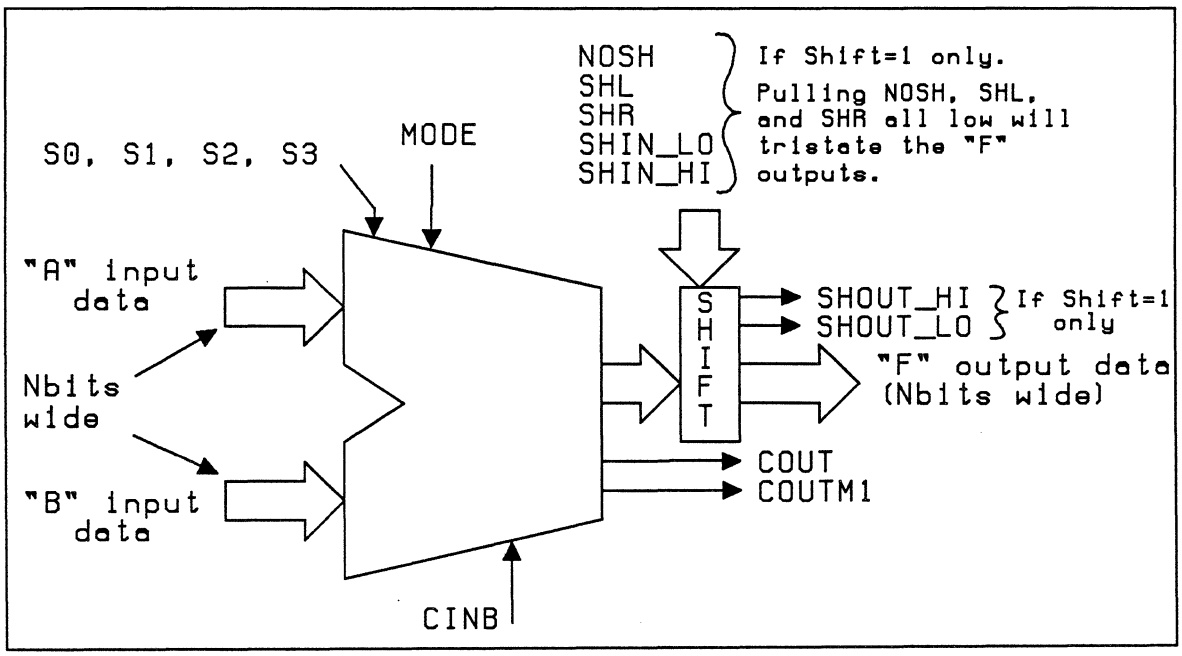
NOTE 1: Only one of SHL, SHR, and NOSH can be active at a time.

NOTE 2: SHL - bit 0 shifts to bit 1 position, etc. SHR - bit 1 shifts to bit 0 position, etc.

SELECTABLE FUNCTIONS

SELECTION				MODE = H	MODE = L (Arithmetic Operations)	
S3	S2	S1	S0	Logical Functions	CINB = H (no carry)	CINB = L (carry)
L	L	L	L	F = A/	F = A	F = A plus 1
L	L	L	H	F = (A+B)/	F = A + B	F = (A+B) plus 1
L	L	H	L	F = (A/)B	F = A + B/	F = (A+B/) plus 1
L	L	H	H	F = 0	F = all 1's	F = 0
L	H	L	L	F = (AB)/	F = A plus AB/	F = A plus AB/ plus 1
L	H	L	H	F = B/	F = (A+B) plus AB/	F = (A+B) plus AB/ plus 1
L	H	H	L	F = A exor B	F = A minus B minus 1	F = A minus B
L	H	H	H	F = AB/	F = AB/ minus 1	F = AB/
H	L	L	L	F = A/ + B	F = A plus AB	F = A plus AB plus 1
H	L	L	H	F = (A exor B)/	F = A plus B	F = A plus B plus 1
H	L	H	L	F = B	F = (A+B/) plus AB	F = (A+B/) plus AB plus 1
H	L	H	H	F = AB	F = AB minus 1	F = AB
H	H	L	L	F = all 1's	F = A plus A	F = A plus A plus 1
H	H	L	H	F = A + B/	F = (A+B) plus A	F = (A+B) plus A plus 1
H	H	H	L	F = A + B	F = (A+B/) plus A	F = (A+B/) plus A plus 1
H	H	H	H	F = A	F = A minus 1	F = A
H = high L = low					+ = logical OR	

FUNCTIONAL BLOCK DIAGRAM



TIMING PARAMETERS AND CELL SIZE AS FUNCTIONS OF INPUT PARAMETERS

Timing parameters are specified for nominal process, Vdd = 5.0 V, and temperature = 25°C. See the *NCR ASIC Data Book* for information on derating factors for process, voltage, and temperature. Input parameters are Nbits and Shift. CL is output capacitance in pF.

PARAM	DESCRIPTION	TYPICAL VALUE (ns)
t_{ifa}	Any A[i] or B[i] input to any F[i] or SHOUT output, arithmetic mode	t_{ifa} = 7.05 + 0.20*Nbits + 0.64*Shift + (0.90 + 0.45*Shift)*CL
t_{ica}	Any A[i] or B[i] input to COUT or COUTM1, arithmetic mode	t_{ica} = 7.30 + 0.20*Nbits + 1.77*CL
t_{cfa}	CINB to any F[i] or SHOUT output, arithmetic mode	t_{cfa} = 4.45 + 0.20*Nbits + 0.64*Shift + (0.90 + 0.45*Shift)*CL
t_{cca}	CINB to COUT or COUTM1, arithmetic mode	t_{cca} = 4.95 + 0.20*Nbits + 1.77*CL
t_{ifl}	Any A[i] or B[i] input to any F[i], or SHOUT output, logic mode	t_{ifl} = 3.57 + 0.55*Shift + (0.90 + 0.45*Shift)*CL
t_{sf}	S0,S1,S2,S3 or MODE to any F[i], COUT, COUTM1 or SHOUT output	t_{sf} = 8.45 + 0.068*Nbits + 0.77*Shift + (0.90 + 0.45*Shift)*CL
t_{shf}	NOSH, SHL, SHR or SHIN to any F[i] or SHOUT output	t_{shf} = 0.80 + 0.039*Nbits + 1.35*CL

Cell Width (mils) = 21.67 + 3.31*Shift
Cell Height (mils) = 2.94 + 2.07*Nbits + 0.78*Shift

TIMING EXAMPLES

Timing for a 16–Bit ALU Without Post Shift

SYMBOL	PARAMETER	NOMINAL VDD=5V		WORST CASE VDD=4.5V						UNITS
		TA=25C		TA=70C		TA=85C		TA=125C		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t_{ifa}	From A, B to F, SHOUT Arithmetic Mode		11		22		23		26	ns
t_{ica}	From A, B to COUT, COUTM1 Arithmetic Mode		11		23		24		27	ns
t_{cfa}	From CINB to F, SHOUT Arithmetic Mode		8		17		17		19	ns
t_{cca}	From CINB to COUT, COUTM1 Arithmetic Mode		9		19		19		22	ns
t_{ifl}	From A, B to F, SHOUT Logic Mode		4		8		9		10	ns
t_{sf}	From S0-S3, MODE to F, SHOUT		10		20		21		24	ns

Switching Characteristics (Input tr,tf=1.4ns, CL=0.5pF)

Timing for an 8–Bit ALU With Post Shift

SYMBOL	PARAMETER	NOMINAL VDD=5V		WORST CASE VDD=4.5V						UNITS
		TA=25C		TA=70C		TA=85C		TA=125C		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t_{ifa}	From A, B to F, SHOUT Arithmetic Mode		10		20		21		24	ns
t_{ica}	From A, B to COUT, COUTM1 Arithmetic Mode		10		21		22		25	ns
t_{cfa}	From CINB to F, SHOUT Arithmetic Mode		7		15		16		18	ns
t_{cca}	From CINB to COUT, COUTM1 Arithmetic Mode		8		17		17		19	ns
t_{ifl}	From A, B to F, SHOUT Logic Mode		5		10		10		12	ns
t_{sf}	From S0-S3, MODE to F, SHOUT		10		21		22		25	ns
t_{shf}	From NOSH, SHL, SHR, SHIN to F, SHOUT		2		4		4		4	ns

Switching Characteristics (Input tr,tf=1.4ns, CL=0.5pF)

APPLICATION NOTES

Use Of The Shift Function

The optional shift function allows more complicated functions than are possible with just the '181 compatible operations.

The three shift control lines (SHL, SHR, and NOSH) control the shift operation. Only one of these lines may be asserted active high at a time. The physical implementation uses three CMOS transmission gates per output. The three control lines select one of the three transmission gates per output. If all three control lines are pulled low, the output bus will be tristated.

The SHIN and SHOUT lines allow shift-in and shift-out at the ends of the output bus. For instance, when shifting left using the SHL control, the signal level on the SHIN_LO will appear on F[0], and the most significant bit of the ALU operation will appear on SHOUT_HI. The SHOUT_LO output will be tristated in this case. Similarly, a shift right operation will take the signal from SHIN_HI to F[msb], and the least significant ALU bit will appear on SHOUT_LO.

These operations are shown in Figure A-1.

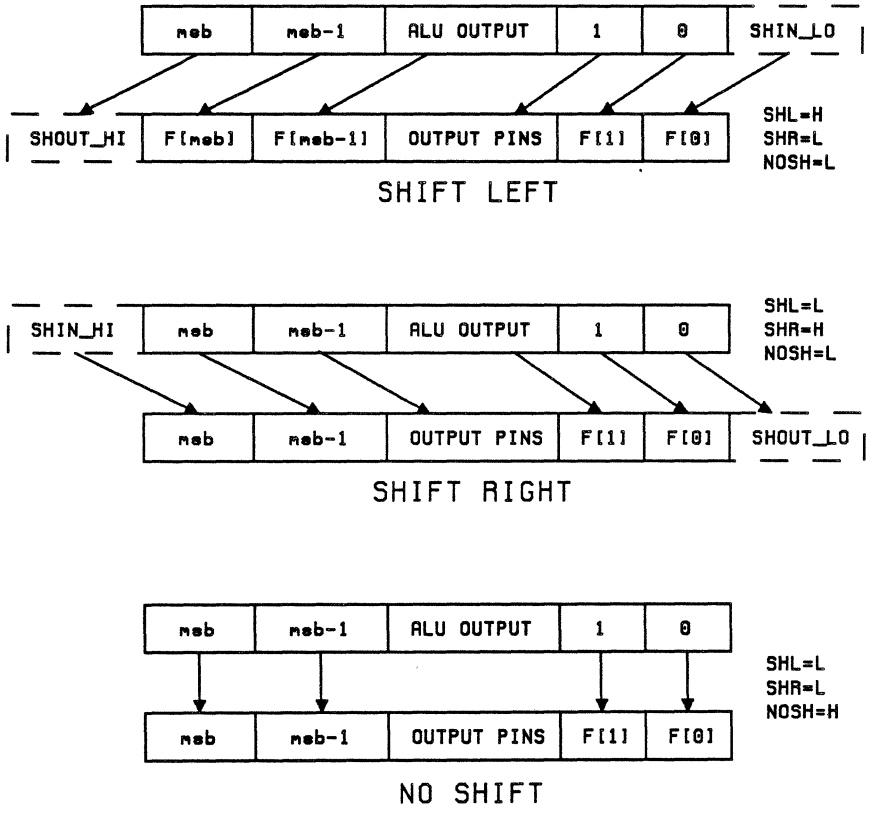


Figure A-1 Shift operations

Limitations

QUICKSIM Save and Restore functions will not currently operate with generated cells.

VS1500 Binary Counter

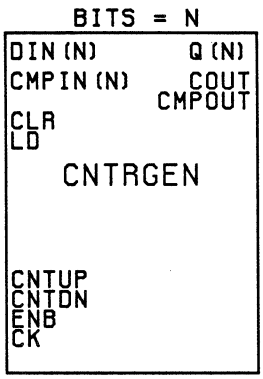
GENERAL DESCRIPTION

- The counter is an n-bit synchronous (positive edge triggered) counter
- Allowed functions include count up, count down, load, clear and hold
- Optionally tristatable outputs
- An optional comparison circuit can be included to output a high level when the output bus matches an external compare bus

CNTRGEN is a positive edge triggered binary counter. The load, count up and count down operations occur synchronously with the positive clock edge. The clear operation is asynchronous. Carry out and compare out signals are generated from the Q outputs and will have an additional delay from the positive clock edge. The compare and carry out signals can be fed back as load or clear signals, provided the minimum clock period and setup times are satisfied.

SYMBOL

The symbol for CNTRGEN will be unique for each configuration. An example is given here only for reference.



INPUT PARAMETER RANGES

INPUT PARAMETER	ALLOWED RANGE	EXPLANATION
BITS	4 – 32	Number of bits in counter.
CNTUP_OPT	0 or 1	0: No count up option 1: Count up option*
CNTDN_OPT	0 or 1	0: No count down option 1: Count down option*
LOAD_OPT	0 or 1	0: No parallel data load 1: Parallel data load
COMPARE_OPT	0 or 1	0: No compare function 1: Compare output to external bus option
TRISTATE_OPT	0 or 1	0: Always driving outputs 1: Tristatable outputs

* One or both of CNTUP_OPT and CNTDN_OPT must be selected

CNTRGEN

ViGen CONFIGURABLE FUNCTION

INPUTS/OUTPUTS

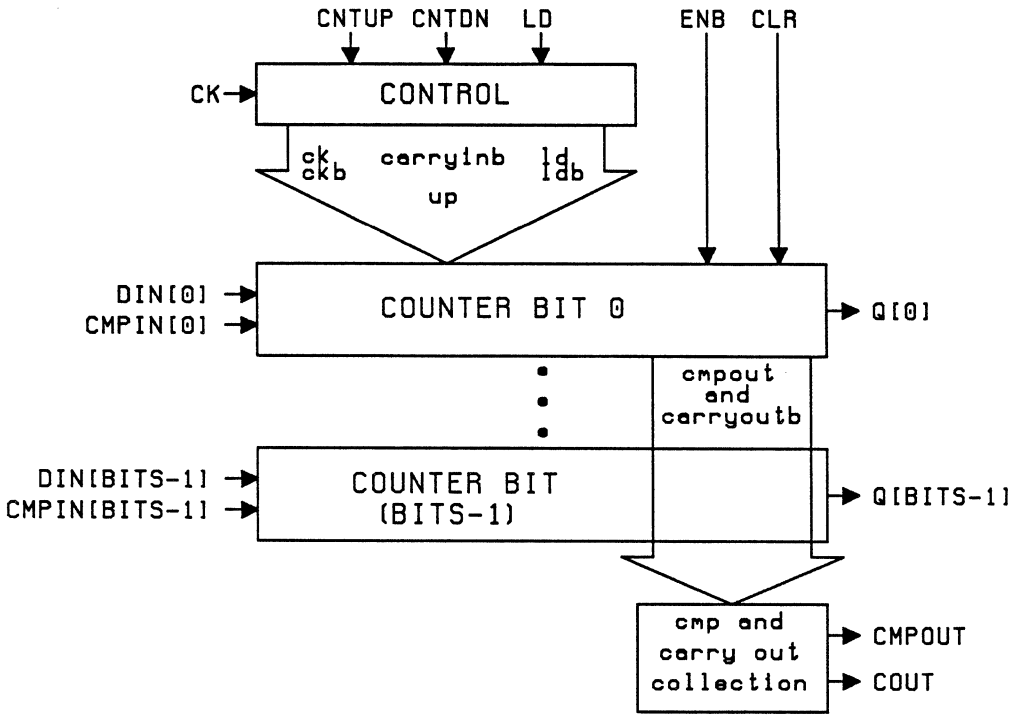
PIN NAME	FUNCTION	REQ/ OPT	CAP (pF)
INPUTS:			
DIN (N)	Parallel load input data bus	OPT	0.118
CMPIN (N)	Compare input bus	OPT	0.096
CLR	Clear – resets all Q outputs to 0	REQ	0.054
LD	Load from DIN (N) on rising edge of CK	OPT	0.054
CNTUP	Count up on rising edge of CK	OPT*	0.060# 0.138&
CNTDN	Count down on rising edge of CK	OPT*	0.080
ENB	Output enable, tristates Q when high	OPT	0.054
CK	Clock	REQ	0.054
OUTPUTS:			
Q (IN)	Output bus	REQ	0.271
COUT	Carry out from most significant bit	REQ	
CMPOUT	Output compare (high if CMPIN bus matches Q bus)	OPT	

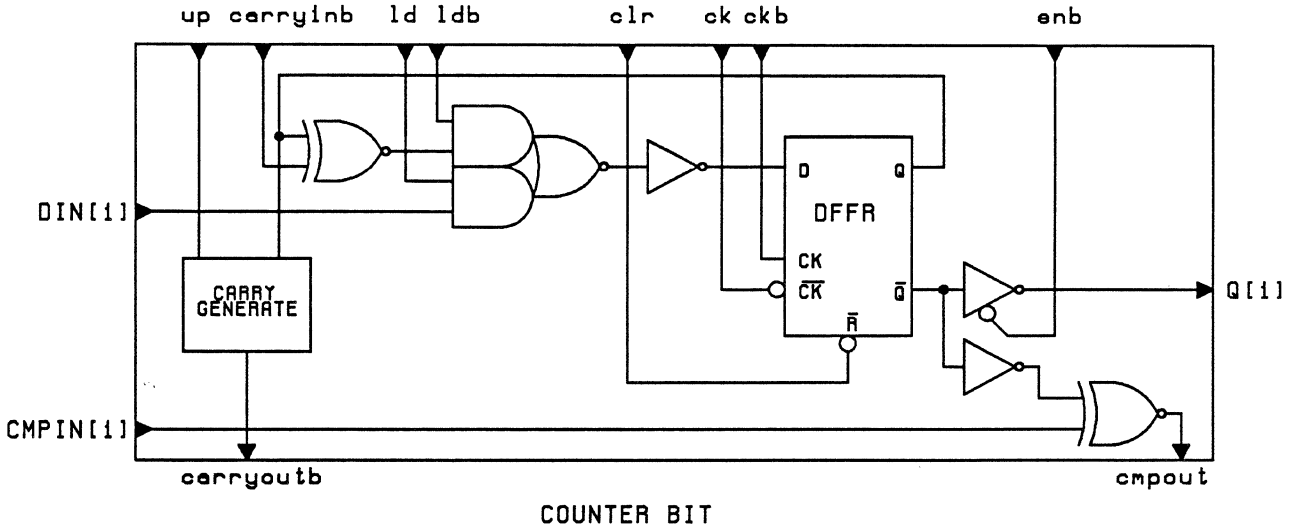
* One or both of CNTUP_OPT and CNTDN_OPT is required

CNTUP_OPT only

& CNTUP_OPT and CNTDN_OPT both selected

FUNCTIONAL BLOCK DIAGRAM

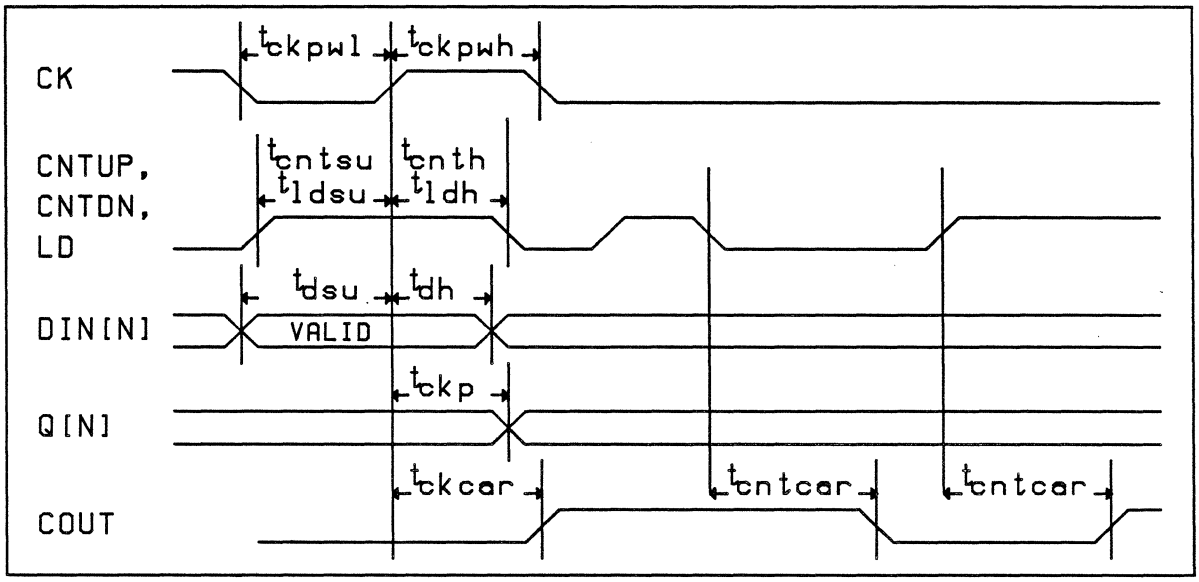




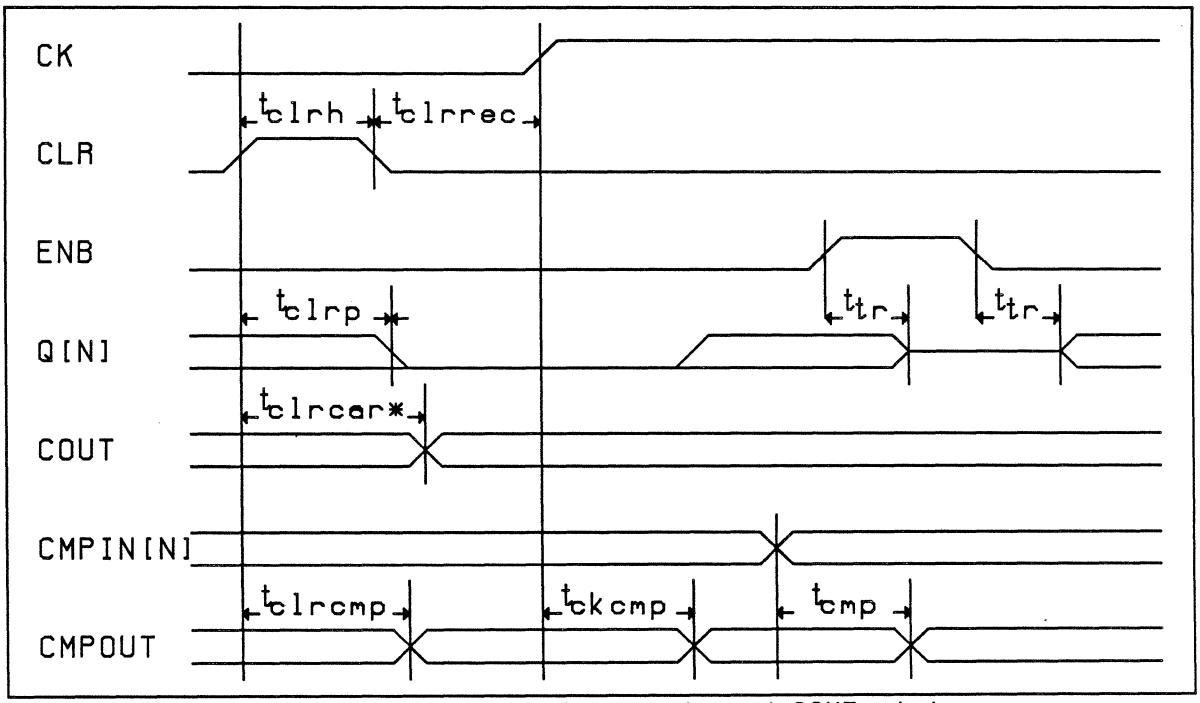
COUNTER BIT

AC WAVEFORMS

Counting and Loading



Clear, Compare and Tristate Operations



* COUT falling applies in count up mode and COUT rising applies in count down mode.

TIMING PARAMETERS AND CELL SIZE AS FUNCTIONS OF INPUT PARAMETERS

1. The input parameters are: BITS, CNTUP_OPT, CNTDN_OPT, LOAD_OPT, COMPARE_OPT, and TRISTATE_OPT. CL is output capacitance in pF.
2. Timing parameters are specified for nominal process, Vdd=5.0 volts, and temperature = 25°C. See the NCR ASIC Data Book for information on derating factors for process, voltage and temperature. The (t_{plh}) notation refers to the output switching from low to high, and (t_{phl}) from high to low.

PARAM	DESCRIPTION	TYPICAL VALUE (ns)
t_{cntsu}	CNTUP, CNTDN set up time	5.13 + 0.258*BITS + 5.74*CNTDN_OPT
t_{cnth}	CNTUP, CNTDN hold time	0
t_{ldsu}	LD setup time	5.34
t_{ldh}	LD hold time	0
t_{dsu}	DIN setup time	5.12
t_{dh}	DIN hold time	0
t_{ckpwl}	Min CK pulse width (low)	6.66 + 0.137*BITS + 0.65*CNTDN_OPT
t_{ckpwh}	Min CK pulse width (high)	6.66 + 0.137*BITS + 0.65*CNTDN_OPT
t_{clrh}	Min CLR pulse width (high)	6.66 + 0.137*BITS + 0.65*CNTDN_OPT
t_{clrrec}	CLR recovery time (CLR release to rising CK edge)	6.66 + 0.137*BITS + 0.65*CNTDN_OPT
t_{ckp}	CK to Q delay	t_{plh} = 5.06 + 0.022*BITS + 0.18*COMPARE_OPT + 0.80*TRISTATE_OPT + (0.80 + 1.22*TRISTATE_OPT)*CL t_{phl} = 5.39 + 0.021*BITS + 0.25*COMPARE_OPT + 1.01*TRISTATE_OPT + (0.86 + 0.83*TRISTATE_OPT)*CL
t_{tr}	ENB to tristate on or off delay	t_{plh} = 2.53 + 0.020*BITS + 2.13*CL t_{phl} = 2.86 + 0.020*BITS + 1.51*CL
t_{ckcar}	CK to carry out delay	t_{plh} = 8.90 + 0.308*BITS + 0.32*TRISTATE_OPT + 2.38*CNTDN_OPT + 1.01*CL t_{phl} = 8.90 + 0.042*BITS + 0.54*TRISTATE_OPT + 1.19*CNTDN_OPT + 0.83*CL
t_{cmp}	CMPIN to CMPOUT delay	t_{plh} = 2.92 + 0.116*BITS + 3.18*CL t_{phl} = 3.70 + 0.135*BITS + 0.57*CL

CNTRGEN

ViGen CONFIGURABLE FUNCTION

t_{ckcmp}	CK to CMPOUT delay	
	TRISTATE_OPT = 0:	$t_{plh} = t_{ckp}$ (Low to High) + 3.44 + 0.68*BITS + 3.18*CL
		$t_{phl} = t_{ckp}$ (High to Low) + 1.10 + 0.035*BITS + 0.57*CL
	TRISTATE_OPT = 1:	t_{plh} = 8.44 + 0.190*BITS + 3.18*CL
		t_{phl} = 6.85 + 0.069*BITS + 0.57*CL
t_{clrp}	CLR to Q delay	t_{phl} = 4.31 + 0.083*BITS + 0.67*COMPARE_OPT + 1.44*TRISTATE_OPT + (0.80 + 1.22*TRISTATE_OPT)*CL
t_{clrcar}	CLR to COUT delay	t_{plh} = 11.30 + 0.336*BITS + 0.47*TRISTATE_OPT + 1.01*CL
		t_{phl} = 10.39 + 0.031*BITS + 0.47*TRISTATE_OPT + 1.37*CNTDN_OPT + 0.83*CL
t_{clrcmp}	CLR to CMPOUT delay	$t_{plh} = t_{clrp}$ + 2.14 + 0.162*BITS + 3.18*CL
		$t_{phl} = t_{clrp}$ + 0.97 + 0.038*BITS + 0.57*CL
t_{cntcar}	CNTUP to COUT delay	t_{plh} = 2.47 + 0.353*BITS + 6.22*CNTDN_OPT + 1.01*CL
		t_{phl} = 2.73 + 0.097*BITS + 1.16*CNTDN_OPT + 0.83*CL
t_{cntcar}	CNTDN to COUT delay	t_{plh} = 4.55 + 0.360*BITS + 1.01*CL
		t_{phl} = 4.29 + 0.093*BITS + 0.83*CL

Cell Width (mils)

for BITS ≤ 16:

9.7 + 2.1*LOAD_OPT + 2.6*COMPARE_OPT + 1.0*TRISTATE_OPT + (offset)

offset = (0 CNTUP_OPT only)
(0.5 CNTDN_OPT only)
(1.3 CNTUP_OPT and CNTDN_OPT)

for BITS > 16:

19.2 + 4.1*LOAD_OPT + 5.2*COMPARE_OPT + 2.1*TRISTATE_OPT + (offset)

offset = (0 CNTUP_OPT only)
(1.0 CNTDN_OPT only)
(2.6 CNTUP_OPT + CNTDN_OPT)

Cell Height (mils) = 3.6 + 3.13*BITS for BITS ≤ 16
= 3.6 + 1.56*BITS for even BITS > 16
= 3.6 + 1.56*(BITS + 1) for odd BITS > 16

TIMING EXAMPLES:

8–Bit Up/Down Counter with Compare and Tristate

8 BIT UP/DOWN COUNTER with COMPARE_OPT and TRISTATE_OPT			NOMINAL VDD=5V		WORST CASE VDD=4.5V						UNITS
			TA=25C		TA=70C		TA=85C		TA=125C		
SYMBOL	PARAMETER		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t_{ckp}	CK to Q delay	LH		7.1		14.5		15.1		17.0	ns
		HL		7.7		15.7		16.4		18.4	
t_{ckcar}	CK to COUT delay	LH		14.6		29.9		31.2		35.0	ns
		HL		11.4		23.3		24.4		27.3	
t_{ckcmp}	CK to CMPOUT delay	LH		11.6		23.7		24.7		27.7	ns
		HL		7.7		15.8		16.5		18.4	
t_{cmp}	CMPIN to CMPOUT delay	LH		5.4		11.1		11.6		13.1	ns
		HL		5.1		10.4		10.8		12.2	
t_{tr}	ENB to Q (on or off)	LH		3.8		7.7		8.0		9.0	ns
		HL		3.8		7.7		8.1		9.1	
t_{clrp}	CLR to Q delay	HL		8.1		16.6		17.3		19.4	ns
t_{clrcar}	CLR to COUT delay	LH		15.0		30.7		32.0		35.9	ns
		HL		12.9		26.4		27.6		30.9	
t_{clrcmp}	CLR to CMPOUT delay	LH		13.1		26.9		28.1		31.5	ns
		HL		9.7		19.8		20.7		23.2	
t_{entcar}	CNTUP to COUT delay	LH		12.0		24.6		25.7		28.8	ns
		HL		5.1		10.4		10.9		12.2	
t_{entcar}	CNTDN to COUT delay	LH		7.9		16.3		17.0		19.0	ns
		HL		5.4		11.2		11.7		13.1	
t_{entsu}	CNTUP/CNTDN setup time		12.9		26.5		27.7		31.0		ns
t_{enth}	CNTUP/CNTDN hold time		0		0		0		0		ns
t_{clrh}	Min CLR pulse width (high)		8.4		17.2		18.0		20.2		ns
t_{clrrec}	CLR recovery time		8.4		17.2		18.0		20.2		ns
t_{ckpwl}	Min CK pulse width (low)		8.4		17.2		18.0		20.2		ns
t_{ckpwh}	Min CK pulse width (high)		8.4		17.2		18.0		20.2		ns

Switching Characteristics (Input tr, tf=1.4ns, CL=0.5pF)

16–Bit Up Counter with Load

16 BIT UP COUNTER with LOAD_OPT			NOMINAL VDD=5V		WORST CASE VDD=4.5V						UNITS
			TA=25C		TA=70C		TA=85C		TA=125C		
SYMBOL	PARAMETER		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t_{ckp}	CK to Q output delay	LH		5.8		11.9		12.4		13.9	ns
		HL		6.2		12.6		13.2		14.8	
t_{ckcar}	CK to COUT delay	LH		14.3		29.4		30.7		34.4	ns
		HL		10.0		20.5		21.4		24.0	
t_{clrp}	CLR to Q delay	HL		6.0		12.4		12.9		14.5	ns
t_{clrcar}	CLR to COUT delay	HL		11.3		23.2		24.2		27.1	ns
t_{cntcar}	CNTUP to COUT delay	LH		8.6		17.7		18.5		20.7	ns
		HL		4.7		9.6		10.1		11.3	
t_{dsu}	DIN setup time		5.1		10.5		11.0		12.3		ns
t_{dh}	DIN hold time		0		0		0		0		ns
t_{cntsu}	CNTUP set up time		9.3		19.0		19.8		22.2		ns
t_{cnth}	CNTUP hold time		0		0		0		0		ns
t_{ldsu}	LD setup time		5.3		10.9		11.4		12.8		ns
t_{ldh}	LD hold time		0		0		0		0		ns
t_{clrrec}	CLR recovery time		8.9		18.1		18.9		21.2		ns
t_{clrh}	Min CLR pulse width (high)		8.9		18.1		18.9		21.2		ns
t_{ckpwl}	Min CK pulse width (low)		8.9		18.1		18.9		21.2		ns
t_{ckpwh}	Min CK pulse width (high)		8.9		18.1		18.9		21.2		ns

Switching Characteristics (Input tr,tf=1.4ns, CL=0.5pF)

APPLICATIONS NOTES

Primary Uses

CNTRGEN generates synchronous n–bit binary counters. Count up, count down and load functions occur synchronously with the positive clock edge. If none of these control signals are active during a rising clock edge, the counter state will remain unchanged. LD will take priority over CNTUP or CNTDN if LD and either CNTUP or CNTDN are active. If both CNTUP and CNTDN are active, the cell will operate as if it is in count up mode. The clear function is asynchronous and sets all Q outputs low regardless of the state of clock or any control signal. The compare function is implemented with EXNOR gates comparing the Q output to the CMPIN input for each bit. These signals are then collected with an AND function to produce the CMPOUT signal. Note that the CMPIN input is not clocked and, therefore, changes on the CMPIN bus can produce changes on the CMPOUT pin regardless of the state of CK.

Note that if TRISTATE_OPT = 0, then CMPOUT is a dependent output and the CK to CMPOUT delay is dependent upon the loading and delay to the Q outputs. To calculate the CK to CMPOUT delay, the delay to Q must be calculated first. For a more complete explanation of dependent outputs, refer to the Timing Information section for the VS2000 standard cell library in the NCR ASIC data book.

Naming Conventions

ViGen will automatically create a default cell name for each unique CNTRGEN configuration. The name of each configurations is encoded in the following manner:

CNT *bits X load_opt cntup_opt cntdn_opt compare_opt tristate_opt*

Therefore, the default names for the two example counters would be:

CNT8X01111
CNT16X11000

Limitations

QUICKSIM Save and Restore functions will not currently operate with generated cells.

DPRGEN

ViGen CONFIGURABLE FUNCTION

VS1500 Dual Port RAM

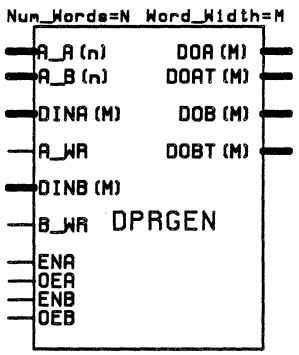
GENERAL DESCRIPTION

- Variable size RAM array with two independent, bidirectional ports (A,B).
- Read-only option on port A saves area.
- Tristate and always-driving outputs available on both ports.
- Variable number of words and word size, up to 16K bits.

The dual port RAM generator produces a RAM array with two fully independent read/write ports. Each port uses clocked operation to reduce complexity and operating power. Each port is precharged when its enable input is low and can read or write data when its enable input is high. To reduce logic circuitry and save cell area, Port A can be programmed to be a read-only port. Ports must be precharged between successive reads and writes.

SYMBOL

The symbol for DPRGEN will be unique for each configuration. An example is given here only for reference.



INPUT PARAMETER RANGES

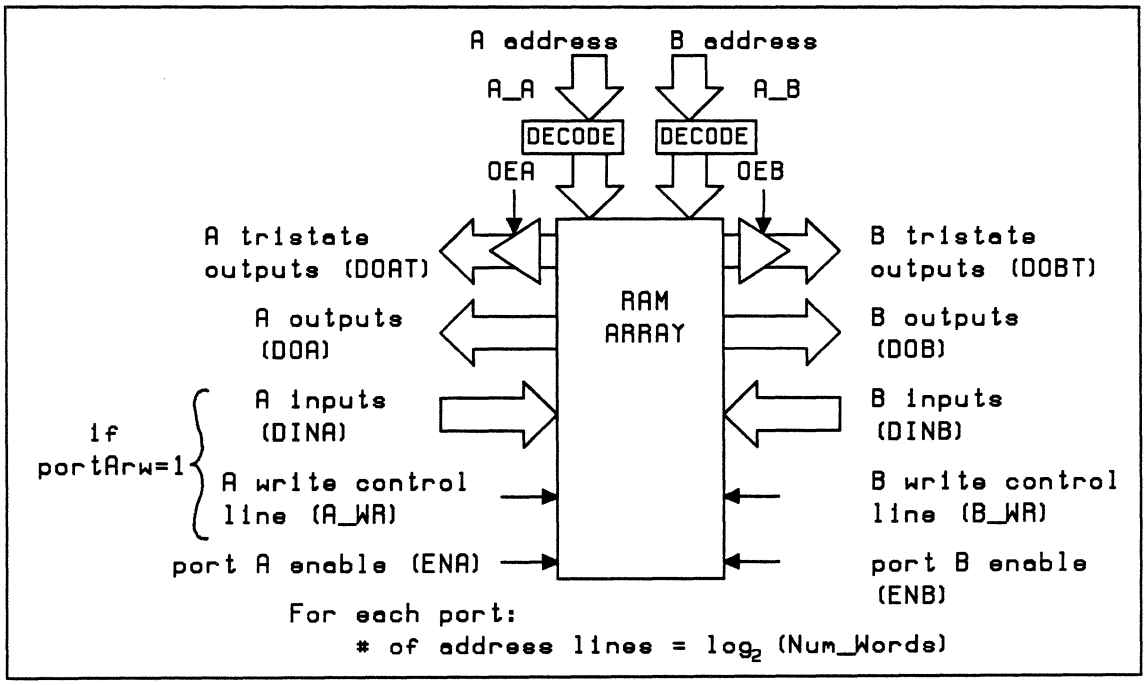
INPUT PARAMETER	ALLOWED RANGE	EXPLANATION
Num_Words	2–2048, even only	Number of data words in the RAM array. The number of address lines for each port will be n = $\log_2$ (Num_Words). Number of bits = Num_Words*Word_Width must be ≤16384.
Word_Width	2 – 32	Number of bits in a data word. This affects the number of DINA, DINB, DOA, DOAT, DOB, and DOBT pins. Number of bits = Num_Words*Word_Width must be ≤16384.
PortArw	0 or 1	0: port "A" is read-only. 1: port "A" is read/write. If port "A" is read-only, then the DINA and A_WR pins are removed from the schematic symbol.

INPUTS/OUTPUTS

Definitions of the DPRGEN inputs and outputs are given in the following table. Input and output pin names are listed in pin-number sequence.

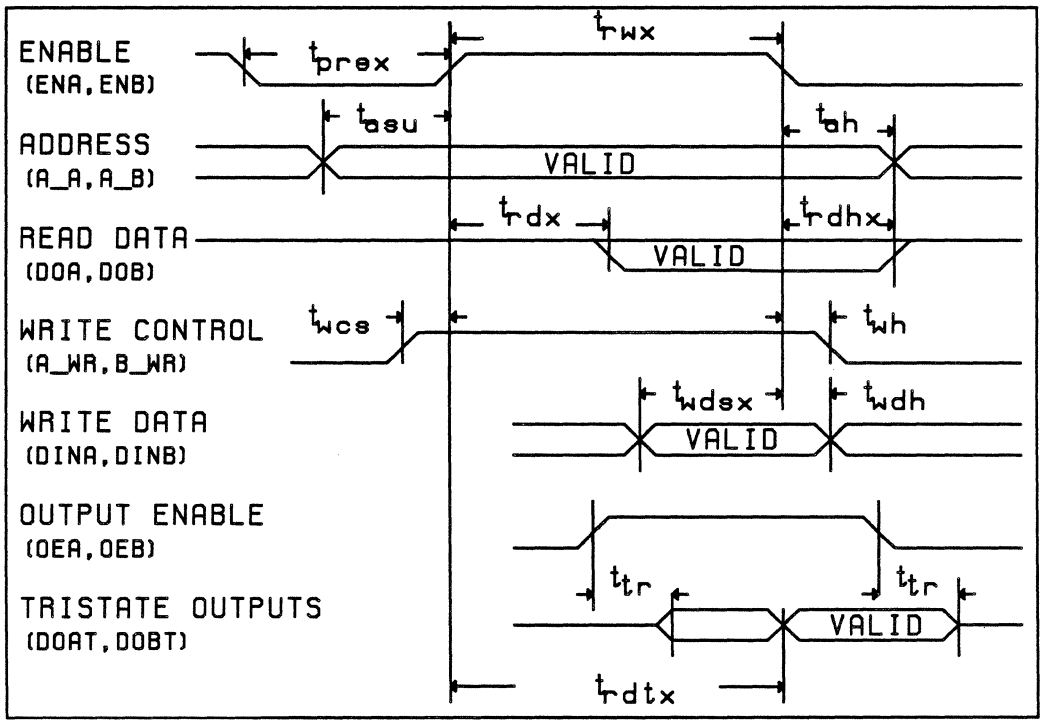
PIN NAME	DEFINITION	REQ/ OPT	CAP
INPUTS:			
A_A (n)	A port address bus (n = $\log_2$ (Num_Words))	REQ	0.159 pF
A_B (n)	B port address bus (n = $\log_2$ (Num_Words))	REQ	0.159 pF
DINA (M)	A port data in bus (M = Word_Width)	OPT	0.113 pF
A_WR	A port read/write control (active high write)	OPT	0.058 pF
DINB (M)	B port data in bus (M = Word_Width)	REQ	0.113 pF
B_WR	B port read/write control (active high write)	REQ	0.058 pF
ENA	A port enable pin	REQ	0.167 pF
OEA	DOAT (M) tristate enable pin	REQ	0.058 pF
ENB	B port enable pin	REQ	0.173 pF
OEB	DOBT (M) tristate enable pin	REQ	0.058 pF
OUTPUTS:			
DOA (M)	Always-driving A port output bus	REQ	
DOAT (M)	Tristate A port output bus	REQ	0.105 pF
DOB (M)	Always-driving B port output bus	REQ	
DOBT (M)	Tristate B port output bus	REQ	0.105 pF

FUNCTIONAL BLOCK DIAGRAM



AC WAVEFORMS

The following waveforms apply equally to port A or port B. If the read only option is chosen for port A (portArw = 0), then ignore waveforms referring to write lines. A suffix of "x" in a timing parameter name should be substituted with "a" or "b" for the appropriate port.



TIMING PARAMETERS

t_{prex}	Minimum precharge time for port x
t_{asu}	Minimum address setup before rising enable (either port)
t_{ah}	Minimum address hold after falling enable (either port)
t_{rdx}	Maximum read access from rising enable on port x
t_{rdtx}	Maximum read access to tristate outputs from rising enable on port x
t_{rdhx}	Maximum read data hold after falling enable on port x
t_{wcs}	Write control setup before rising enable (either port)
t_{wh}	Write control hold after falling enable (either port)
t_{wdsx}*	Write data setup before falling enable of port x
t_{wdh}	Write data hold after falling enable (either port)
t_{tr}	Output enable to tristate on or off delay (either port)
t_{rwx}	Minimum enable high time of port x for read or write

* See Applications Note titled *Port Contention* which follows.

TIMING PARAMETERS AND CELL SIZE

1. The input parameters are: Num_Words, Word_Width, portArw. CL is output capacitance in pF.
2. Internal column decode ("coldec") must be determined first. Follow these steps (these are the same steps followed by the layout compiler for DPRGEN):
   a. nbits = Num_Words*Word_Width;
   b. `IF (   0 <= nbits <   32) (trial_ncols =   2)` ncols = number of RAM columns
      `IF (  32 <= nbits <   64) (trial_ncols =   8)`
      `IF (  64 <= nbits <  192) (trial_ncols =  16)`
      `IF ( 192 <= nbits < 1024) (trial_ncols =  32)`
      `IF (1024 <= nbits < 8192) (trial_ncols =  64)`
      `IF (8192 <= nbits < inf.) (trial_ncols = 128)`
   c. coldec = trial_ncols/Word_Width
   d. Round coldec down to nearest valid value (1,2,4,8, or 16)
   e. Word limits on different values of column decode:
      Num_words must be >= 384 for coldec = 16
      Num_words must be >= 80 for coldec = 8
3. Num_Words is internally rounded up to the nearest multiple of 2*coldec.
4. TIMING EQUATIONS & CELL SIZE: (use nrows = Num_Words/coldec, ncols = Word_Width*coldec)
   - ^{t}prea is for port A, ^{t}preb is for port B, etc. No "a" or "b" suffix means either port.
   - All times in nanoseconds and are NOMINAL (Vdd=5.0 volts, T=25°C, Nom. process)
   -See NCR ASIC Data Book for process, voltage, and temperature derating.

		Example: Num_Words=16, Word_Width=16, portArw=1, CL=0 (coldec = 2)
^{t}asu = 1.38 + .0656*nrows	MIN	1.9 ns
^{t}ah = 0	MIN	0.0 ns
^{t}prea = 3.81 + .0419*nrows + .0497*ncols + 0.64*CL	MIN	5.7 ns
^{t}preb = 3.77 + .0419*nrows + .0509*ncols + 0.63*CL	MIN	5.7 ns
^{t}rda = 7.18 + .1513*nrows + .1134*ncols + 0.99*CL	MAX	12.0 ns
^{t}rdta = 7.30 + .1513*nrows + .1134*ncols + 1.70*CL	MAX	12.1 ns
^{t}rdb = 7.10 + .1538*nrows + .1197*ncols + 0.97*CL	MAX	12.2 ns
^{t}rdtb = 6.90 + .1538*nrows + .1197*ncols + 1.66*CL	MAX	12.0 ns
^{t}rdha = ^{t}prea	MAX	5.7 ns
^{t}rdhb = ^{t}preb	MAX	5.7 ns
^{t}tr = 2.0 + 1.70*CL	MIN	2.0 ns
^{t}wcs min. = −4.0, max. = 4.0		+4/−4 ns
^{t}wh min. = 0, max. = 4.0		0 to 4.0 ns
^{t}wdsa = 5.78 + .0538*nrows + .0291*ncols (See Note 1)	MIN	7.1 ns
^{t}wdsb = 6.06 + .0500*nrows + .0250*ncols (See Note 1)	MIN	7.3 ns
^{t}wdh = 1.17 + .0075*nrows + .0138*ncols	MIN	1.7ns
^{t}rwx = ^{t}rdx	MIN	a:12.0ns, b:12.2ns

Cell Width (mils) = 12.683 + 1.924*nrows + 1.745*portArw + 0.448*coldec (coldec > 2)
Cell Width (mils) = 16.876 + 1.924*nrows + 3.109*portArw (coldec = 1)
Cell Height (mils) = 11.237 + 1.143*ncols + 0.963*$\log_2$(nrows) + 0.211*coldec − 0.039*Word_Width
Note 1: See Applications Note titled *Port Contention* for a possible exception to this parameter value.

TIMING EXAMPLES FOR 16X16 AND 128X8 DUAL PORT RAMS

SYMBOL	16X16 DUAL PORT RAM (PORT A READ/WRITE)	NOMINAL VDD=5V		WORST CASE VDD=4.5V						UNITS
		TA=25C		TA=70C		TA=85C		TA=125C		
	PARAMETER	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t_{asu}	Address Setup Time Before ENABLE	2		4		4		5		ns
t_{ah}	Address Hold Time after ENABLE	0		0		0		0		ns
t_{prea}	Precharge Time	6		12		13		15		ns
t_{preb}	Precharge Time	6		12		13		15		ns
t_{rwa}	Read or Write Enable High	13		26		27		30		ns
t_{rwb}	Read or Write Enable High	13		26		27		30		ns
t_{rda}	Read Access Time		13	26		27		30		ns
t_{rdb}	Read Access Time		13	26		27		30		ns
t_{rdta}	Tristate Read Access Time		13	27		28		31		ns
t_{rdtb}	Tristate Read Access Time		13	26		27		31		ns
t_{rdha}	Read Data Hold Time		6		12		13		15	ns
t_{rdhb}	Read Data Hold Time		6		12		13		15	ns
t_{tr}	Tristate and Untristate Time		3	6		6		7		ns
t_{wcs}	Write Control Setup	-4	4	-8	8	-9	9	-10	10	ns
t_{wh}	Write Control Hold	0	4		8		9		10	ns
t_{wdsa}	Write Data Setup	7		15		15		17		ns
t_{wdsb}	Write Data Setup	7		15		16		17		ns
t_{wdh}	Write Data Hold	2		3		4		4		ns

Switching Characteristics (Input tr, tf=1.4ns, CL=0.5pF)

SYMBOL	128X8 DUAL PORT RAM (PORT A READ-ONLY) PARAMETER	NOMINAL VDD=5V TA=25C		WORST CASE VDD=4.5V TA=70C		TA=85C		TA=125C		UNITS
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t_{asu}	Address Setup Time Before ENABLE	2		5		5		6		ns
t_{ah}	Address Hold Time after ENABLE	0		0		0		0		ns
t_{prea}	Precharge Time	8		16		17		19		ns
t_{preb}	Precharge Time	8		16		17		19		ns
t_{rwa}	Read Enable High	17		36		37		42		ns
t_{rwb}	Read or Write Enable High	18		36		38		43		ns
t_{rda}	Read Access Time		17	36		37		42		ns
t_{rdb}	Read Access Time		18	36		38		43		ns
t_{rdta}	Tristate Read Access Time		18	37		38		43		ns
t_{rdtb}	Tristate Read Access Time		18	37		38		43		ns
t_{rdha}	Read Data Hold Time		8		16		17		19	ns
t_{rdhb}	Read Data Hold Time		8		16		17		19	ns
t_{tr}	Tristate and Untristate Time		3	6		6		7		ns
t_{wcs}	Write Control Setup	-4	4	-8	8	-9	9	-10	10	ns
t_{wh}	Write Control Hold	0	4		8		9		10	ns
t_{wdsb}	Write Data Setup	8		17		18		20		ns
t_{wdh}	Write Data Hold	2		4		5		5		ns

Switching Characteristics (Input tr, tf=1.4ns, CL=0.5pF)

APPLICATION NOTES

Using the Tristate Outputs DOAT(i) and DOBT(i)

The tristate outputs of the dual port RAM generator are not implemented in quite the same way as in some other NCR supercells. Figure 1 shows the circuits used for both always-driving outputs (DOA(i) and DOB(i)) and the tristate outputs. Instead of a full CMOS transmission gate for the tristate outputs, a single N-channel transistor is used to reduce circuit area. The effect of this is that the tristate outputs, when driven high, will not reach a full VDD level. Instead, they will reach a level of about 3.3 volts when VDD is 5.0 volts. This is enough to be considered a logic high for subsequent gate inputs, but there will be reduced noise margin and increased DC power dissipation caused by this non-rail level.

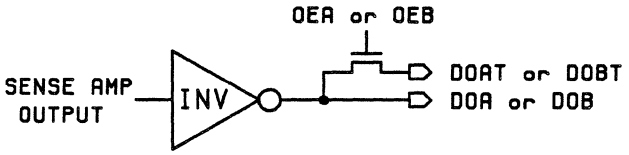


Figure 1 Tristate output circuit

For this reason it is important to use pull-up cells on each tristate output line (see Figure 2). This is good design practice for any tristate line used in semicustom design. The PU30 cell is not meant to provide a valid logic level on a bus line, but only to provide a "default" level for those times when it is not actively driven by anything else.

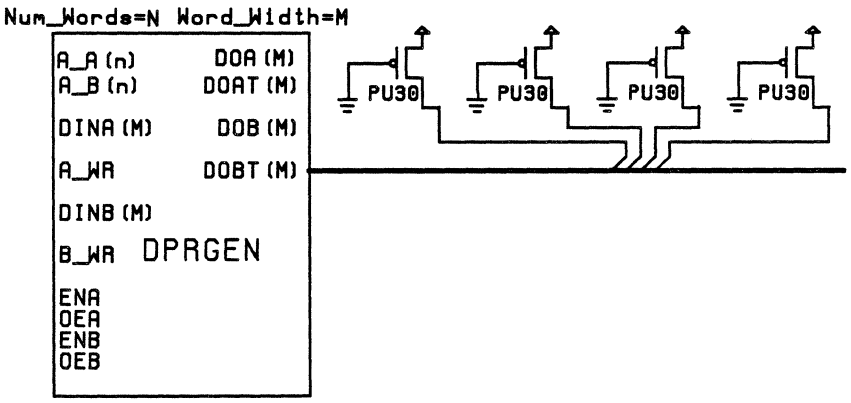


Figure 2 Tristate bus pullups

The pullups will cause no problem with "late 1's" on the tristate outputs. During the precharge phase for each port, the internal bit lines and sense amp inputs are pulled high. This causes the always-driving outputs to also go high during precharge. If the output enable line (OEA or OEB) is asserted at the beginning of a read phase, then the tristate outputs will also stay high if they were not immediately previously driven low by another gate. There is no possibility of a glitch on this line (and therefore losing the good high

level) since the read access time for a "1" is actually zero. Only 0's in the data word being read have non-zero access time.

Reducing AC Power Dissipation

For most designers, the AC power dissipation of CMOS semicustom chips is not a problem needing consideration since it is very low compared to bipolar or NMOS implementations. Some applications, though, need low AC operating power as well.

If the address lines are allowed to switch several times before becoming stable at the address setup time, then excess AC power dissipation will occur due to the large capacitances on the internal address lines and their complements. This kind of power dissipation is calculated as:

$P=CV^2f,$

P = power
C = total driven capacitance
V = voltage swing (= VDD)
f = switching frequency

Since C is fixed inside the RAM, reducing this power can only be done by reducing the number of times that the address lines switch. This can be done by latching the address lines externally to the dual-port RAM.

Port Contention

The A and B ports of the dual-port RAM are meant to be independent in operation. The possibility of contention between the two ports does exist, however, and can happen in the following combinations:

1. One port's write cycle overlaps the other port's read cycle to the same address (read/write contention).
2. Both ports' write cycles overlap to the same address (write/write contention).

Note that port contention doesn't exist if both ports are reading from the same address. Discrete, dual-port RAM chips contain contention logic that detects when both ports are attempting to access the same address (whether reading or writing). The port which drives this address second is given a busy signal by the contention logic, which inhibits it from actually reading or writing to that address. When the first port has switched to a different address, the busy signal to the second port is deactivated.

This generator does not contain such contention logic, so care must be taken to assure correct operation. The benefit to not including this logic is more flexible operation. For instance, two-port reads can occur from the same address without one port being locked out. This is important for register files, for an example see Figure 4.

Read/write contention can happen in three different ways (see Figure 3), and only one of them causes real contention. Of the three read/write overlaps shown, only the first, where the read on one port isn't complete before the other port starts to write, is a case of real port contention and must be avoided. In the second case, where both enables overlap within +/– 10ns, there will be no contention *if the data to be written is stable from the beginning of the write enable.* This overrides the write data setup time parameter in the AC characteristics table. In the third case, the data written to the addressed word will be stable in time for the reading port to access it, so no contention occurs.

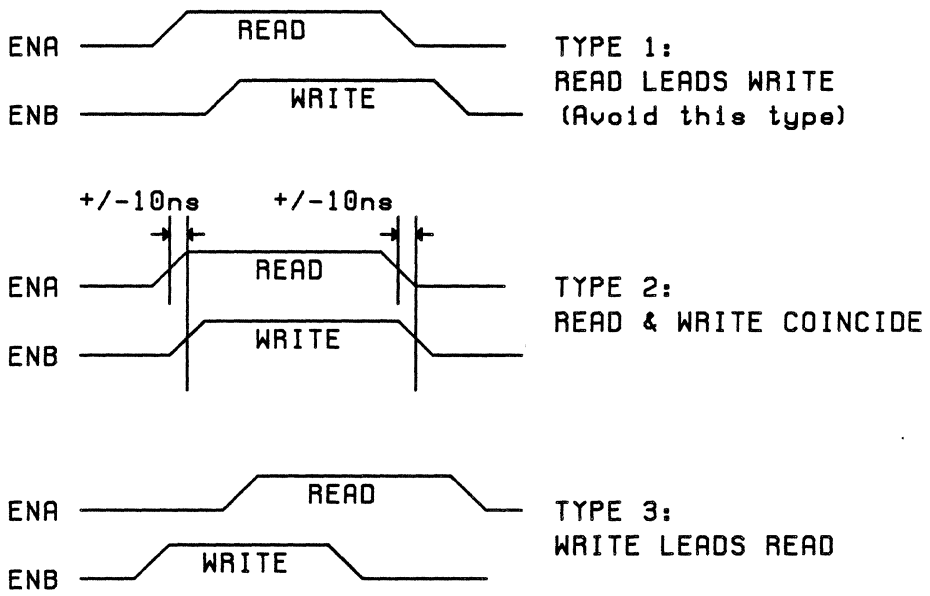


Figure 3 Read/Write port contention

Write/write combinations will always cause contention, since the data in the addressed word will contain unknown data except when both ports write identical data. If the enable signals ENA and ENB don't overlap their high times at all, then there is no contention.

Designing for Testability

A test program for a semicustom chip must be able to verify that all circuit functions operate correctly. This implies both functional correctness (correct design) and fault detection (the circuit was manufactured without defects). Several methods exist for a achieving both of these goals, all of which involve some trade-off between degree of checking and the amount of extra logic required. These include:

1. Multiplex part pins to the address and data pins in a test mode.
2. Use scan registers to serially shift in address and input data, and shift out output data.

See the NCR VS2000 Standard Cell Library Databook, Section 6, "Designing for Testability" for more information.

Example of Register File Used With ALU

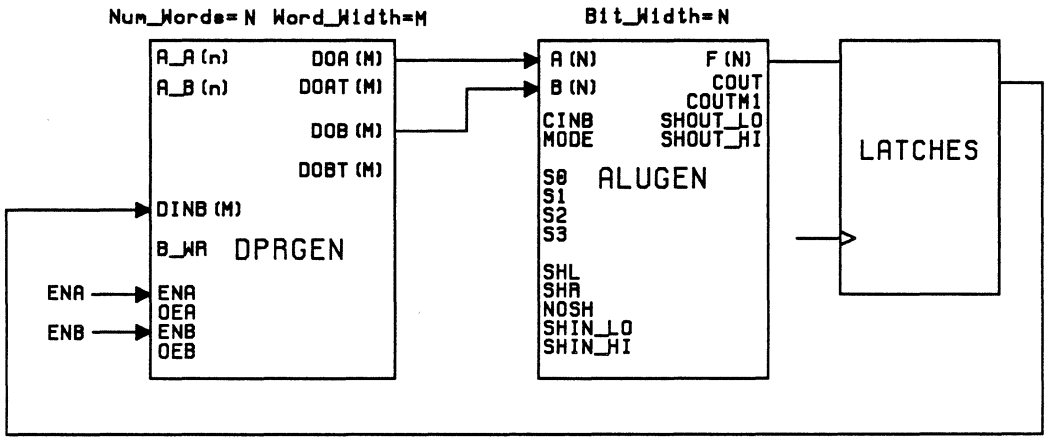


Figure 4 DPRGEN as a register file source for ALU operands

In Figure 4, the two operands for the ALU are read simultaneously from the A and B ports of the RAM, and then the ALU result is written back to the RAM through port B on the next cycle. The "PortArw" parameter for this RAM is set to "0" because the A port never has data written to it. There will never be any possibility of port contention because of this. The enable signals for each port can be tied together, which causes a dummy read of port A when the result is being written back through port B, or they can be separated as shown. Tying them together requires less logic to generate the ENA signal, but increases the operating power dissipation.

Example of FIFO

Figure 5 shows an implementation of a FIFO using the dual-port RAM supercell. Less sophisticated shift register type FIFOs can be built using the "SHFTGEN" supercell.

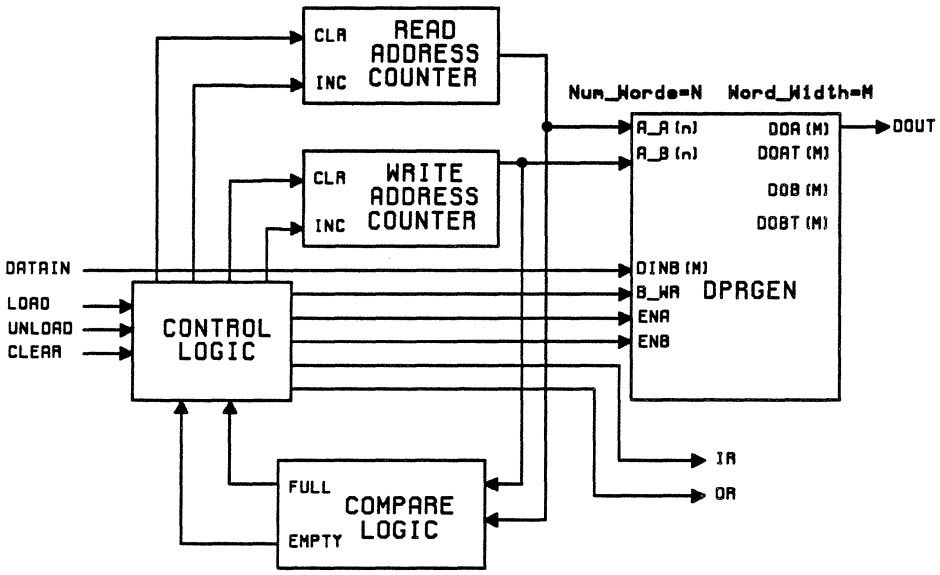


Figure 5 DPRGEN used in a FIFO buffer

- Data is loaded with the LOAD input from DATAIN, and data is led out to the DOUT bus using UNLOAD. Address comparison determines whether the buffer is full (read address equals write address plus 1) or empty (write address equals read address plus 1). The IR flag (Input Ready) indicates that the FIFO is not full, and the OR flag (Output Ready) indicates the not empty condition.

Naming Conventions

ViGen will automatically create a default cell name for each unique DPRGEN configuration with the configuration information encoded in the following manner:

DPR *num_words* X *num_bits* NW (for portArw = 0)

or

DPR *num_words* X *num_bits* W (for portArw = 1)

Therefore, the default names for the two example configurations would be:

DPR16X16W

DPR128X16NW

Limitations

QUICKSIM Save and Restore functions will not currently operate with generated cells.

SIMULATION ERROR MESSAGES

Simulation using the Mentor Quicksim simulator is done using a behavioral language model (BLM). This model checks for timing and usage errors during simulation. A listing of the possible error message types and their descriptions is included here.

MESSAGES REPORTED BY DPRGEN CONTAIN TWO MAJOR TYPES:

1. OPERATIONAL ERRORS – caused by invalid operation: Unknown data on control lines.
   Invalid states/modes.
   Improper sequence of states.

   OPERATIONAL ERROR SYNTAX

   $inst_name: ERROR *time_of_error error_description*
   $inst_name: *action taken by model because of error*
   $inst_name:

   OPERATIONAL ERRORS INCLUDE

   UNKNOWNS ON ADDRESSES DURING READ.
   UNKNOWNS ON ADDRESSES DURING WRITE.
   WRITE TO SAME ADDRESS FROM A&B PORTS SIMULTANEOUSLY.
   READ/WRITE TO SAME ADDRESS FROM A&B PORTS OUT OF SPEC (see *Port Contention).*
   NO PRECHARGE BETWEEN SUCCESSIVE READ/WRITE OPERATIONS.

   example:

   $i21(dprgen): ERROR Time=2013 UNKNOWN ADDRESS ON A PORT DURING WRITE
   $i21(dprgen): UNKNOWN DATA WILL BE WRITTEN TO EVERY BLOCK OF RAM ARRAY
   $i21(dprgen): WHICH IS ENABLED.

2. TIMING ERRORS/WARNINGS – caused by violation of timing specifications in documentation.
   Warning messages are produced when transitions occur within a margin delta of specification value.

   TIMING ERROR/WARNING SYNTAX

   $inst_name: ERROR *msg_type* OF TYPE *type_check spec_name current_time*
   $inst_name: *control_pin_name direction_of_edge* at *transition_time* ns
   $inst_name: *data_pin_name* changed at *transition_time* ns
   $inst_name: spec is *spec_value* ns (+*margin_value* ns margin)

   examples:

   $i21(dprgen): ERROR TIMING VIOLATION OF TYPE SETUP t_WDS Time=2103 ns
   $i21(dprgen): ENB falling at 1987 ns
   $i21(dprgen): DINB changed at 1980 ns
   $i21(dprgen): spec is 17 ns (+5 ns margin)

   $i21(dprgen): ERROR MARGIN WARNING OF TYPE SETUP t_WDS Time=2103 ns
   $i21(dprgen): ENB falling at 1987 ns
   $i21(dprgen): DINB changed at 1967 ns
   $i21(dprgen): spec is 17 ns (+5 ns margin)

FIFOGEN

ViGen CONFIGURABLE FUNCTION

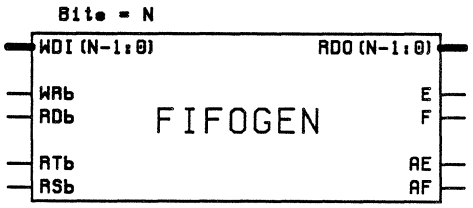
VS1500 FIFO Generator

FEATURES

- Compiled FIFO cell allows flexible configurations of up to 6K bits
- Maximum word width of 36 bits
- Maximum number of words is 512
- Fully asynchronous Read/Write operation
- User configurable Almost Full flag
- User configurable Almost Empty flag
- Retransmit ability

SYMBOL

The symbol for FIFOGEN will be unique for each configuration. An example is given here only for reference.



DESCRIPTION

FIFOGEN produces a compiled first in first out memory block. FIFOGEN is based on a dual port latch memory cell which allows for completely asynchronous read/write operation. Read and write address values are stored in ring counters which may be cleared with the RSb (Reset) input. The read counter may be independently cleared using the RTb (retransmit) input, thus allowing data to be re-read multiple times. FIFOGEN also has status flags for Empty, Full, Almost Empty, and Almost Full. The user may configure the Almost Full and Almost Empty flags to activate any chosen offset from full and empty respectively.

INPUT PARAMETER RANGES

Input Parameter	Allowed Range	Explanation
words	4 – 512	Number of words in the FIFO[1], [2]
bits	1 – 36	Word size in **bits**[1]
ae_offset	1 – words–1	Almost Empty offset from Empty for flag activation
af_offset	1 – words–1	Almost Full offset from Full for flag activation

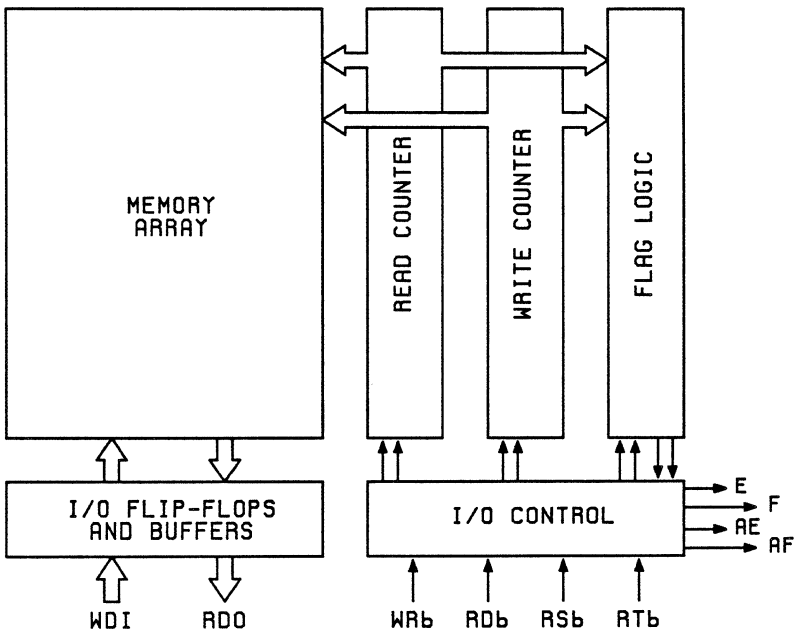
[1] Total number of bits (**bits x words**) must be ≤ 6K.

[2] **words** is internally rounded up to a multiple of the column decode. The column decode is internally selected to be 1–8 (by 1) by FIFOGEN to optimize area and performance. Any changes in **words** will be reported to the user at configuration time.

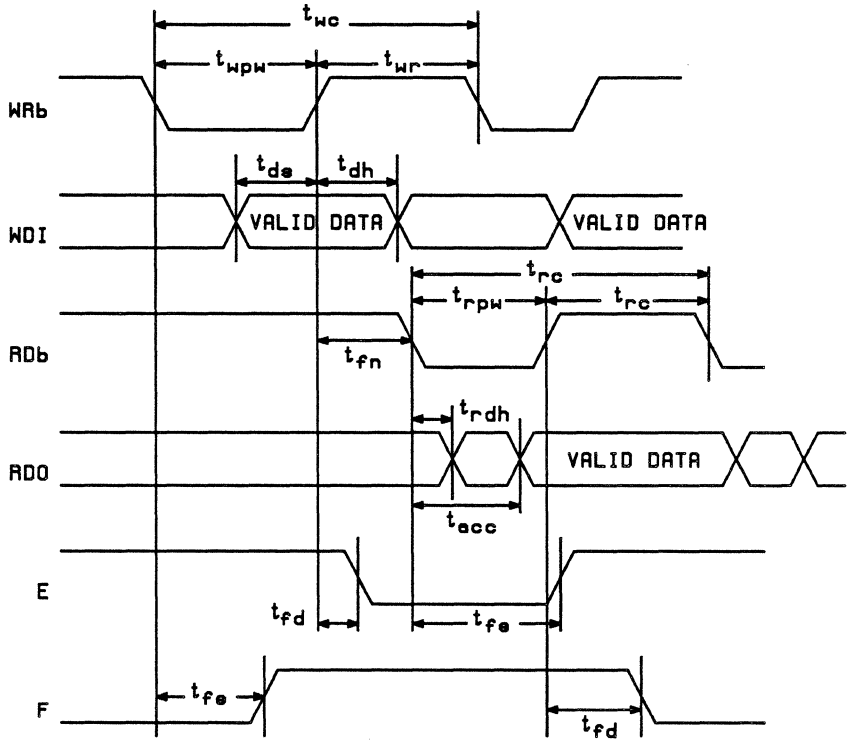
INPUTS/OUTPUTS

Pin/Bus Name (in TDL order)	Function	Req/ Opt	Cap (pF)
INPUTS:			
WRb	Write control pin, active LOW	Req	0.10
RDb	Read control pin, active LOW	Req	0.10
RTb	Retransmit control pin, active LOW. Resets the read counter to first physical memory location	Req	0.10
RSb	Reset control pin, active LOW. Resets the read and write counters to first physical memory location.	Req	0.15
WDI[0:bits−1]	Write data input bus	Req	0.08
OUTPUTS:			
RDO[0:bits−1]	Read data output bus	Req	
E	Empty Flag, set HIGH on falling edge of RDb when reading last data word, reset on next rising edge of WRb	Req	
F	Full Flag, set HIGH on falling edge of WRb when writing last data word, reset on next rising edge of RDb	Req	
AE	Almost Empty Flag, set HIGH on falling edge of RDb when reading last data word, reset on rising edge of WRb	Req	
AF	Almost Full Flag, set HIGH on falling edge of WRb when writing last data word, reset on rising edge of RDb	Req	

FUNCTIONAL BLOCK DIAGRAM



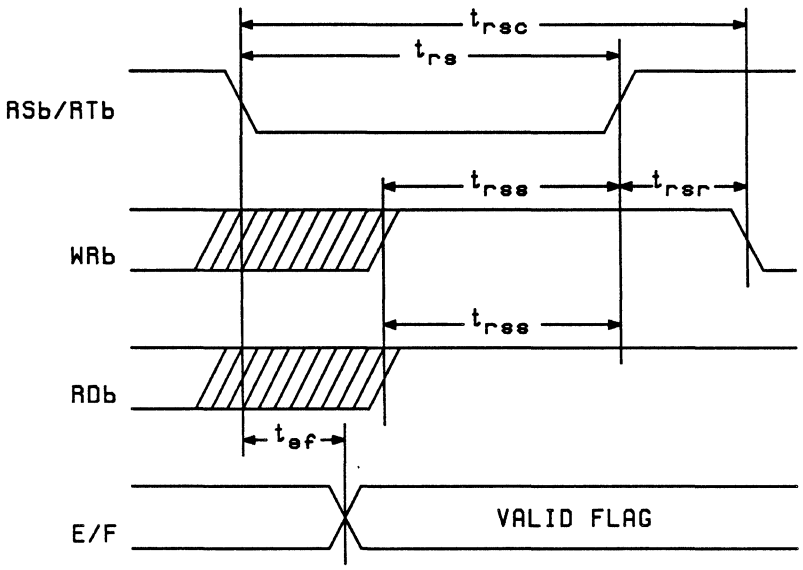
AC WAVEFORMS – READ/WRITE



TIMING PARAMETERS – READ/WRITE

Name	Description
t_{wc}	Minimum Write cycle time
t_{wpw}	Minimum WRb pulse low
t_{wr}	Minimum Write recovery time (WRb high)
t_{ds}	Minimum data setup time from rising edge of WRb
t_{dh}	Minimum data hold time after falling edge of WRb
t_{rc}	Minimum Read cycle time
t_{rpw}	Minimum RDb pulse low
t_{rr}	Minimum Read recovery time (RDb high)
t_{acc}	Maximum delay RDb falling to valid data
t_{rdh}	Minimum data hold time after falling edge of RDb
t_{fe}	Flag Enable Delay
t_{fd}	Flag Disable Delay
t_{fn}	First Write to first valid Read

AC WAVEFORMS – RESET/RETRANSMIT



TIMING PARAMETERS – RESET/RETRANSMIT

Name	Description
t_{rsc}	Minimum Reset/Retransmit cycle time
t_{rs}	Minimum Reset/Retransmit Pulse low
t_{rss}	Minimum WRb/RDb setup before RSb/RTb rising
t_{rsr}	Minimum Reset/Retransmit recovery time
t_{ef}	Minimum Reset/Retransmit to flag output

TIMING, POWER, AND AREA EQUATIONS

FIFOGEN will select its internal configuration based on the requested parameters. A simple algorithm is used which determines an internal architecture which will optimize area/performance tradeoffs. This architecture defines the number of rows and columns of the memory array. Because the user has no control over the internal column decode, **the resulting configuration may have more words than requested** (but never less). The algorithm is geared to minimize this discrepancy. The user will be notified at configuration time of the actual size of the array. Equations are given in terms of input parameters and internal architecture parameters. These can be found by following these steps:

1. CL is the output capacitance in pF.
2. Generator input parameters which affect delays are **words**, **bits**. Derived parameters are **ncols** and **nrows**. **ncols** is the number of cell columns per I/O bit.
3. Calculate first-pass values for derived parameters, for area and performance a square array is best. Use (i).
4. In order to assure reasonable wordline performance, check array width; if it is greater than optimum length (72), decrease column decode (**ncols**) by one. (ii).
5. Otherwise to optimize number of rows, check if increasing column decode would not cause too long a word line (iii).
6. Finally recalculate **nrows** if **ncols** changed (iv).

   i. if (**words** ≤ 32) **ncols** = 1
   else {
   nrows = SQRT(**words***bits**), rounded up to an integer.
   ncols = **words/nrows**, rounded up to an integer.
   if (**ncols** >8) **ncols** = 8
   ii. if (**ncols***bits** > 72) **ncols** = **ncols**−1;
   iii. else if (**ncols***bits** < (72 − **bits**) & **ncols** < 8) **ncols** = **ncols** +1;
   }
   iv. **nrows** = **words/ncols**, rounded up to an integer

All times in nanoseconds and are nominal (V_{DD}=5.0 volts, T=25°C, Nominal process). See the *NCR ASIC Data Book* for process, voltage, and temperature derating factors.

PARAM	DESCRIPTION	TYPICAL VALUE (ns)
t_{wc}	Write Cycle	= 16.26 + 0.212***nrows** + 0.025***bits** + 0.048*(**ncols*bits**)
t_{wpw}	Write Pulse low	= 7.10 + 0.036*(**ncols−1**) + 0.137**nrows** + 0.015*(**ncols*bits**)
t_{wr}	Write Pulse high	= 5.20 + 0.010***nrows** + 0.014***bits** + 0.013*(**ncols*bits**)
t_{ds}	Write Data Setup	= 1.00
t_{dh}	Write Data Hold	= 2.11 + 0.017***bits** + 0.003*(**ncols*bits**)
t_{rc}	Read Cycle Time	= 7.63 + 0.086*(**ncols−1**) + 0.084***nrows** + 0.030*(**ncols*bits**)
t_{rpw}	Read Pulse low	= 7.10 + 0.036*(**ncols−1**) + 0.137**nrows** + 0.015*(**ncols*bits**)
t_{rh}	Read Pulse high	= 6.07 + 0.149*(**ncols−1**) + 0.044*(**ncols*bits**)
t_{acc}	Read Access	= 3.99 + 0.018*(**ncols−1**) + 0.030***bits** + 0.003*(**ncols*bits**) + 0.626***CL**
t_{rdh}	Read Data Hold	= 5.70 + 0.34***CL**
t_{fe}	Flag Enable Delay	= 4.37 + 0.626***CL**
t_{fd}	Flag Disable Delay	= 3.94 + 0.626***CL**
t_{fn}	1st Write to Read	= 8.71 + 0.089*(**ncols−1**) + 0.055***nrows** + 0.032*(**ncols*bits**)
t_{rsc}	RSb/RTb Cycle Time	= 9.56 + 0.118*(**ncols−1**) + 0.209***nrows** + 0.012***bits** + 0.009*(**ncols*bits**)
t_{rs}	RSb/RTb Pulse low	= 2.61 + 0.082***nrows** + 0.005*(**ncols*bits**)
t_{rss}	WRb/RD/b Setup to RSb/RTb	= 0.75 + 0.059***nrows** + 0.007*(**ncols*bits**)
t_{rsr}	RSb/RTb Recovery Time	= 2.88 + 0.068*(**ncols−1**) + 0.038***nrows** + 0.007***bits**
t_{ef}	RSb/RTb to Flag delay	= 4.10 + 0.019*(**ncols−1**) + 0.076***nrows** + 0.007*(**ncols*bits**) + 0.626***CL**

Output Rise and Fall Times

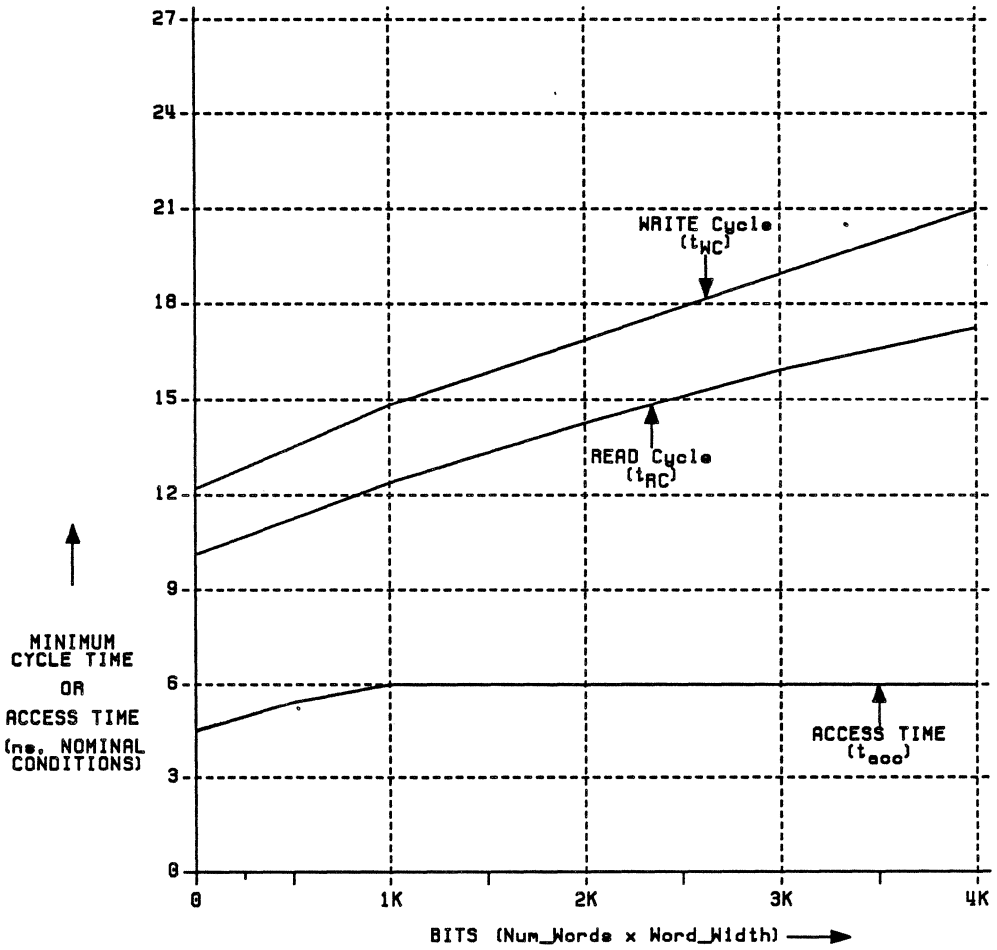
t_r = 0.632 + 2.26*CL
t_f = 1.07 + 1.39*CL

Current Requirements: **I** = 0.148 + 0.004***nrows** + 0.02*(**bits**) mA/MHz
Worst case current process, V_{DD} = 5.0V, T = 25°C

Cell Width (mils) = 30.33 + 1.18*(**ncols*bits**)

Cell Height (mils) = 23.06 + 1.45*(**nrows**) + 0.46*(**cols−1**)

ACCESS AND CYCLE TIMES



(See NCR VS1500 ASIC Data Book for process, voltage, and temperature derating factors.)

TIMING EXAMPLES

Timing for a 64x4 FIFO

Symbol	64x4 FIFO	Nominal V_{DD} = 5V		Worst Case V_{DD} = 4.5V						Units
		TA = 25°C		TA = 70°C		TA = 85°C		TA = 125°C		
	Parameter	Min	Max	Min	Max	Min	Max	Min	Max	
t_{acc}	Read access time		4.8		8.9		9.3		10.4	ns
t_{rdh}	Read data hold		3.6		6.8		7.1		7.9	ns
t_{wc}	Write cycle	12.7		23.7		24.7		27.7		ns
t_{rc}	Read cycle	10.7		19.9		20.8		23.3		ns
t_{fn}	First write till first read	11.5		21.4		22.3		25.1		ns
t_{rsc}	Reset cycle time	8.2		15.3		15.9		17.9		ns
t_{ds}	Write data setup time	0.7		1.2		1.2		1.4		ns
t_{dh}	Write data hold	2.5		4.6		4.8		5.4		ns

Switching characteristics (Input t_r, t_f = 1.4ns, CL = 0.15pF)

Current Requirements: µA/MHz = 0.364
Cell Width (mils) = 53.93
Cell Height (mils) = 43.75
Cell Area ($mils^2$) = 2360

Timing for a 256x9 FIFO

Symbol	256X9 FIFO	Nominal V_{DD} = 5V		Worst Case V_{DD} = 4.5V						Units
		TA = 25°C		TA = 70°C		TA = 85°C		TA = 125°C		
	Parameter	Min	Max	Min	Max	Min	Max	Min	Max	
t_{acc}	Read access time		5.2		9.6		10.0		11.2	ns
t_{rdh}	Read data hold		3.7		6.5		7.1		8.0	ns
t_{wc}	Write cycle	17.2		32.0		33.4		37.5		ns
t_{rc}	Read cycle	14.5		25.9		28.1		31.6		ns
t_{fn}	First write till first read	14.7		26.2		28.5		32.0		ns
t_{rsc}	Reset cycle time	12.4		22.0		24.2		27.1		ns
t_{ds}	Write data setup time	0.7		1.1		1.2		1.4		ns
t_{dh}	Write data hold	2.7		4.9		5.2		6.0		ns

Switching characteristics (Input t_r, t_f = 1.4ns, CL = 0.15pF)

Current Requirements: µA/MHz = 0.476
Cell Width (mils) = 104.67
Cell Height (mils) = 79.47
Cell Area ($mils^2$) = 8320

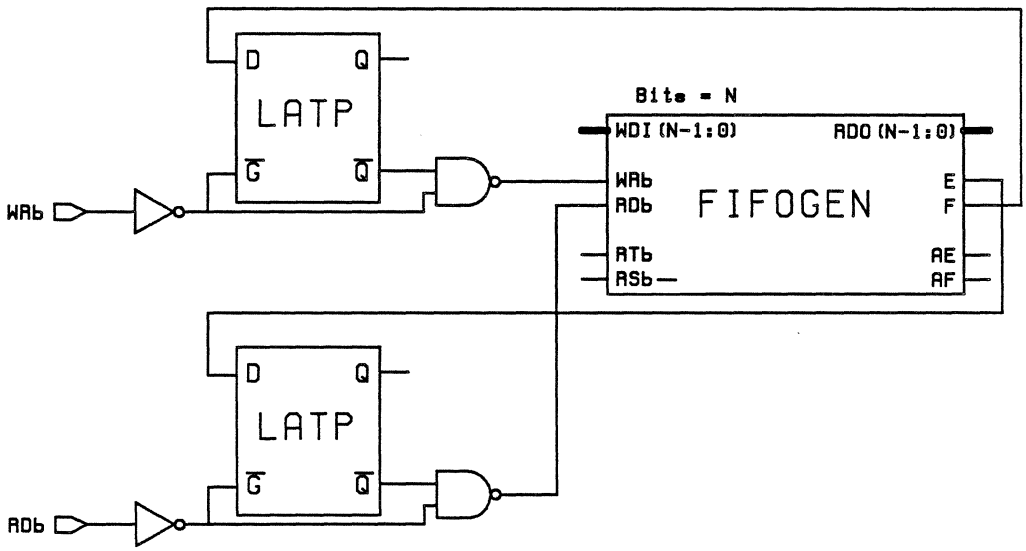
APPLICATION NOTES

On power up a Reset must be performed before any other operations are performed. This assures that the FIFO is starting from a known state.

Reading the FIFO when it is empty and writing to a full FIFO will cause the status flags to be set unknown during simulation. While the logical state of the flags is predictable, the meaning of the status flags (full, empty, etc.) becomes ambiguous. A reset will clear the FIFO.

Retransmit allows the read pointer to be reset to the first physical memory address. Care must be taken to assure that the intended operation is achieved. Retransmit also causes ambiguous definition of the status flags. During retransmit the empty flag will be set false and the full flag true, so only reads should be performed until the FIFO is empty. The full flag will remain true until the FIFO is empty.

The following circuit may be used to disable reads when the FIFO is empty and writes when it is full.



Naming Conventions

ViGen will automatically create a default name for each FIFOGEN configuration, with the configuration information encoded in the following manner:

Fwwwbbeeefff

where:
www = Number of words
bb = Number of bits
eee = Almost–Empty offset
fff = Almost–Full offset

All of the values have the given number of digits. Values that have less than the given number of digits will be left–padded with zeroes. For instance, a 31x8 FIFO with AE=4 and AF=5 will have a name of:

F03108004005

MACCGEN

ViGen CONFIGURABLE FUNCTION

VS1500 MAC Generator

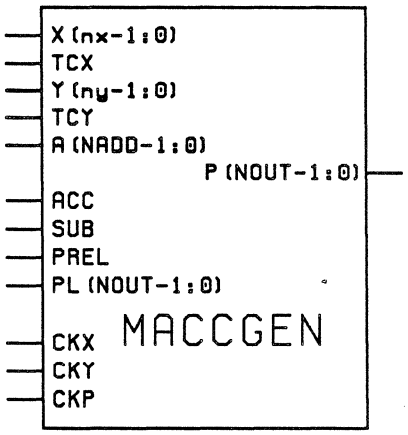
FEATURES

- Variable size compiled multiplier accumulator function
- X and Y inputs independently selectable from 6 to 32 bits wide
- High performance architecture uses Booth encoding and Wallace tree partial product summation
- Two's complement, unsigned magnitude, and mixed mode multiplication
- Input number format can be pin-selectable
- Extra adder input allows implementation of P(i) = X x Y + A ± P(i−1). The A input is usually used for rounding
- X and Y inputs have separate registers and clocks
- Output is registered, with its own clock
- Accumulator functions include ACC, SUB, and preload
- Output may be accumulated up to eight extra bits

SYMBOL

The symbol for MACCGEN will be unique for each configuration. An example is given here only for reference.

where NOUT = nx + ny + acc_bits
NADD = nx + 4



INPUT PARAMETER RANGES

Input Parameter	Allowed Range	Explanation
nx	6 – 32 (even only)	Number of bits in the X input word
ny	6 – 32 (even, ny≤nx)	Number of bits in the Y input word
tcx	0, 1, or 2	=0: X in unsigned magnitude format =1: X in two's complement format =2: X input format selectable with an input pin (TCX)
tcy	0, 1, or 2	=0: Y in unsigned magnitude format =1: Y in two's complement format =2: Y input format selectable with an input pin (TCY)
add_in	0 or 1	Selects whether extra word A is to be added (1 = yes)
acc_bits	2, 4, 6, or 8	Number of extra bits to accumulate beyond nx+ny

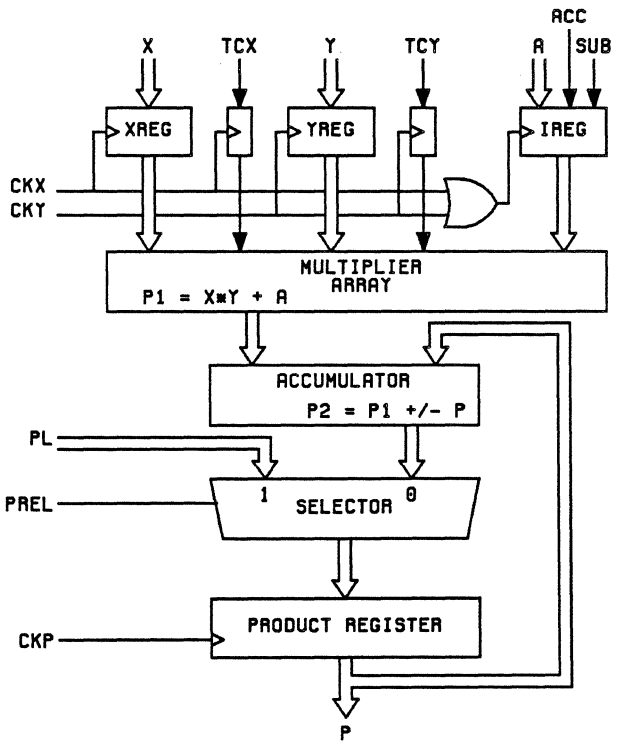
INPUTS/OUTPUTS

Pin/Bis Name (in TDL order)	Function	Req/ Opt	Cap (pF)
INPUTS:			
X[0:nx−1]	X input bus. Data on X is loaded into the X input register on the rising edge of CKX.	Req	0.164
Y[0:ny−1]	Y input bus. Data on Y is loaded into the Y input register on the rising edge of CKY.	Req	0.160
A[0:nx+acc_bits+3]	Optional adder input, usually used for rounding. Data on A is loaded into the A register on the rising edge of the logical OR of CKX and CKY. A is treated as a two's complement number, regardless of tcx and tcy selections. Exists if add_in=1.	Opt	0.060
CKX	X input clock, active on rising edge	Req	0.162
CKY	Y input clock, active on rising edge	Req	0.162
CKP	Clock input for the product register. Active on rising edge.	Req	0.110
TCX	X input two's complement control. When HIGH, the X input is treated as a two's complement number. When LOW, X is treated as an unsigned magnitude number. Exists if tcx=2.	Opt	0.091
TCY	Y input two's complement control, similar to TCX. Exists if tcy=2.	Opt	0.100
ACC	Accumulate control. When HIGH, the contents of the product register are added to or subtracted from the current product XxY(+A). Sampled on the rising edge of the logical OR of CKX and CKY.	Req	0.060
SUB	Subtract. When ACC and SUB are both HIGH, the contents of the product register are subtracted from the current product XxY(+A). Sampled on the rising edge of the logical OR of CKX and CKY.	Req	0.060
PREL	Preload control. When HIGH, data on the PL input bus is loaded into the product register on the rising edge of CKP.	Req	0.375
PL[0:nx+ny+acc_bits−1]	Preload bus. Data from PL is loaded into the product register on the rising edge of CKP.	Req	0.153
OUTPUTS:			
P[0:nx+ny+acc_bits−1]	Product register output bus, indicating the current contents of the product register. Updated on the rising edge of CKP.	Req	

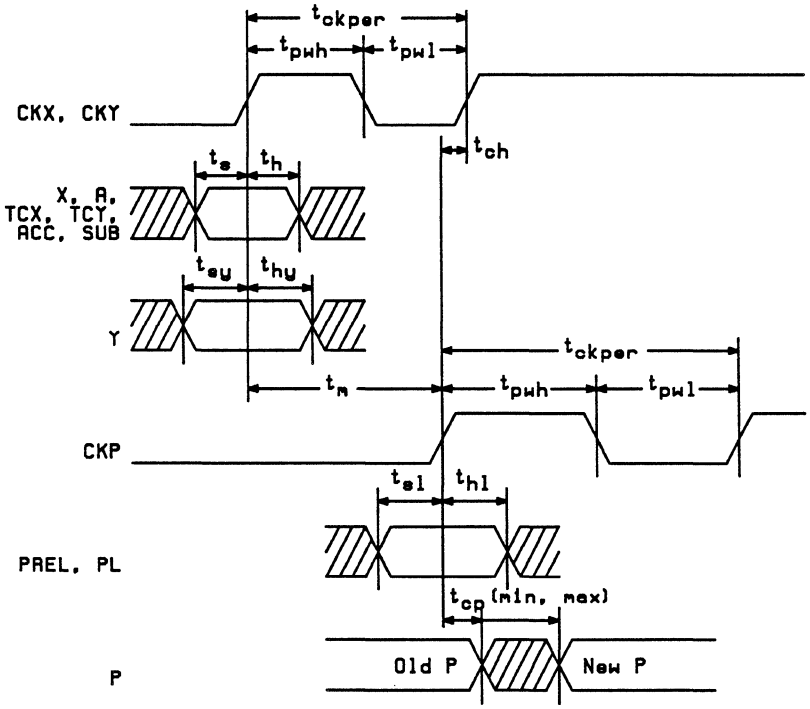
TIMING PARAMETERS

Name	Description
t_{ckper}	Minimum clock period for CKX, CKY, or CKP
t_m	Clocked multiply time
t_{cp}	Output delay time for P outputs after a rising CKP edge. The minimum time is the hold time for the previous product, and the time between minimum and maximum is the bus settling time.
t_s	Setup time for X, A, TCX, and TCY inputs before a rising CKX or CKY edge
t_h	Hold time for X, A, TCX, and TCY inputs after a rising CKX or CKY edge
t_{sy}	Setup time for Y inputs before a rising CKY edge
t_{hy}	Hold time for Y inputs after a rising CKY edge
t_{sl}	Setup time for PREL and PL inputs before a rising CKP edge.
t_{hl}	Hold time for PREL and PL inputs after a rising CKP edge.
t_{pwh}	Minimum high clock pulse width
t_{pwl}	Minimum low clock pulse width
t_{ch}	Relative hold time. CKP must rise at or before CKX or CKY to guarantee correct product output.

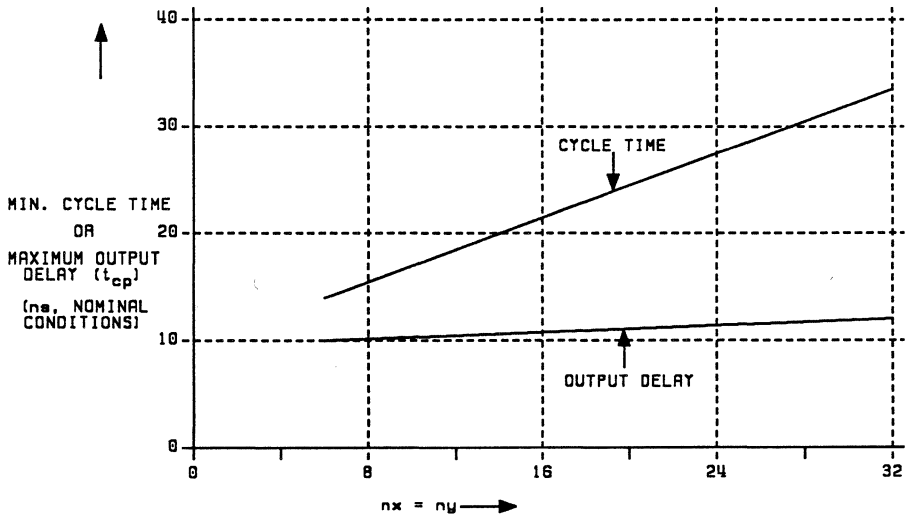
FUNCTIONAL BLOCK DIAGRAM



MACCGEN AC WAVEFORMS



CYCLE TIME FOR SQUARE MACS (nx = ny, acc_bits = 4)



(See NCR ASIC Data Book for process, voltage, and temperature derating factors.)

TIMING, POWER, AND AREA EQUATIONS

1. Input parameters for the following equations are nx, ny and acc_bits. CL is the output capacitance in pF.
2. Timing parameters are specified for nominal process, V_{DD}=5.0 volts, and temperature = 25°C. See the NCR ASIC Data Book for information on derating factors for process, voltage and temperature.
3. All delays for rising and falling outputs are equal, while rise and fall times are individually specified.

PARAM	DESCRIPTION	TYPICAL VALUE (ns)
t_{ckper}	Min. clock period	t_{ckper} = 9.70 + 0.08*nx + 0.67*ny
t_m	Max. clocked multiply/acc or multiply/sub	t_m = 9.70 + 0.08*nx + 0.67*ny
t_{cp}	Output delay NOTE: See App. note #1	$t_{cp(max)}$ = 8.70 + 0.027*nx + 0.060*ny + 0.179*acc_bits + 1.23*CL $t_{cp(min)}$ = 3.57 + 0.020*ny + 1.23*CL
t_s	X,A,TCX, and TCY setup	t_s = 1.56
t_h	X,A,TCX, and TCY hold	t_h = 1.57 + 0.034*nx
t_{sy}	Y setup	t_{sy} = 1.36
t_{hy}	Y hold	t_{hy} = 1.85 + 0.011*ny
t_{sl}	PREL and PL setup	t_{sl} = 1.75 + 0.100*nx + 0.081*ny + 0.097*acc_bits
t_{hl}	PREL and PL hold	t_{hl} = 2.27 + 0.019*nx + 0.028*ny + 0.016*acc_bits
t_{pwh}	Min. high clock pulse	t_{pwh} = 45% of t_{ckper}
t_{pwl}	Min. low clock pulse	t_{pwl} = 45% of t_{ckper}
t_{ch}	Relative hold time	t_{ch} = 0

Rise and Fall Times

t_r = 0.325 + 2.55*CL
t_f = 0.279 + 1.99*CL

Current Requirements: I = 0.0174*ny^2 + 0.115*(nx−ny) mA/MHz
Worst case current process, V_{DD} = 5.5V, T = 25°C
Toggling 0x0 and FF..FFxFF..FF and accumulating

Cell Width (mils) = 9.68 + 2.47*nx + 1.91*ny + 2.33*acc_bits

Cell Height (mils) = 30.72 + 5.86*ny + 2.60*add_in + offset

where: offset = 0 if tcy=1
offset = 11.72 if tcy=0 or 2

TIMING EXAMPLES

8 x 8 two's complement MAC with rounding, 20–bit product register

(MAC8X8TTA4: nx = 8, ny = 8, tcx = 1, tcy = 1, add_in = 1, acc_bits = 4)

Symbol	8 x 8 MAC	Nominal V_{DD} = 5V		Worst Case V_{DD} = 4.5V						Units
		TA = 25°C		TA = 70°C		TA = 85°C		TA = 125°C		
	Parameter	Min	Max	Min	Max	Min	Max	Min	Max	
t_{ckper}	Minimum clock period	15.70		29.4		30.6		34.2		ns
t_m	Maximum multiply time		15.70		29.4		30.6		34.2	ns
t_{cp}	Output delay from CKP	3.91	10.29	7.3	19.3	7.6	20.1	8.5	22.4	ns
t_s	Minimum setup time for X, A inputs	1.56		2.9		3.0		3.4		ns
t_h	Minimum hold time for X, A inputs	1.84		3.4		3.6		4.0		ns
t_{sy}	Minimum setup time for Y inputs	1.36		2.5		2.6		2.9		ns
t_{hy}	Minimum hold time for Y inputs	1.94		3.6		3.7		4.2		ns
t_{sl}	Minimum setup time for PREL, PL	3.59		6.7		7.0		7.8		ns
t_{hl}	Minimum hold time for PREL, PL	2.71		5.1		5.3		5.9		ns
t_{pwh}	Minimum high clock pulse	7.07		13.2		13.8		15.4		ns
t_{pwl}	Minimum low clock pulse	7.07		13.2		13.8		15.4		ns
t_{ch}	Relative hold time	0		0		0		0		ns

Switching characteristics (Input t_r, t_f = 1.4ns, CL = 0.15pF)

16 x 16 unsigned MAC with no rounding, 36–bit product register

(MAC16X16UUN4: nx = 16, ny = 16, tcx = 0, tcy = 0, add_in = 0, acc_bits = 4)

Symbol	16 x 16 MAC	Nominal V_{DD} = 5V		Worst Case V_{DD} = 4.5V						Units
		TA = 25°C		TA = 70°C		TA = 85°C		TA = 125°C		
	Parameter	Min	Max	Min	Max	Min	Max	Min	Max	
t_{ckper}	Minimum clock period	21.70		40.6		42.3		47.3		ns
t_m	Maximum multiply time		21.70		40.6		42.3		47.3	ns
t_{cp}	Output delay from CKP	4.07	10.99	7.6	20.6	7.9	21.4	8.9	24.0	ns
t_s	Minimum setup time for X inputs	1.56		2.9		3.0		3.4		ns
t_h	Minimum hold time for X inputs	2.11		4.0		4.1		4.6		ns
t_{sy}	Minimum setup time for Y inputs	1.36		2.5		2.6		2.9		ns
t_{hy}	Minimum hold time for Y inputs	2.03		3.7		3.9		4.4		ns
t_{sl}	Minimum setup time for PREL, PL	5.03		9.4		9.8		11.0		ns
t_{hl}	Minimum hold time for Y PREL, PL	3.09		5.8		6.0		6.7		ns
t_{pwh}	Minimum high clock pulse	9.77		18.3		19.0		21.3		ns
t_{pwl}	Minimum low clock pulse	9.77		18.3		19.0		21.3		ns
t_{ch}	Relative hold time	0		0		0		0		ns

Switching characteristics (Input t_r, t_f = 1.4ns, CL = 0.15pF)

APPLICATION NOTES

Output delay calculations

Due to the way that the product register is implemented within MACCGEN, the delay from CKP to P is actually dependent on the multiply cycle time. The delay, t_{cp}, is calculated with both a minimum and a maximum value. The minimum value of t_{cp} is the output hold time for the previous product P. Data on the P output bus will not change before t_{cp}(min). The maximum value of t_{cp}, as calculated, only applies when the MAC is being run at maximum frequency (minimum rising CKX/CKY to rising CKP). If the MAC is being run slower, then t_{cp}(max) actually gets less, as shown in Figure 1.

Above a certain multiplication period, referred to in Figure 1 as t_{sync}, t_{cp}(max) will have the same value as t_{cp}(min). In other words, all of the bits in the P bus will switch simultaneously. For multiplication periods between t_{ckper}(min) and t_{sync}, t_{cp}(max) derates linearly as shown.

The period t_{sync} is defined as:

$$t_{sync} = t_{ckper}\ (min) + 1.2^{*}\ (\ t_{cp}(max) - t_{cp}(min)\ )$$

where t_{ckper}, $t_{cp}(max)$, and $t_{cp}(min)$ are all as defined in the parametric equations.

Although the $t_{cp}(min)$ and $t_{cp}(max)$ parameters for each output pin will be slightly different due to differing output capacitive loads, for simplicity the simulation model will assign the shortest $t_{cp}(min)$ and the longest $t_{cp}(max)$ values to the entire P bus. This effect will be minor in most situations.

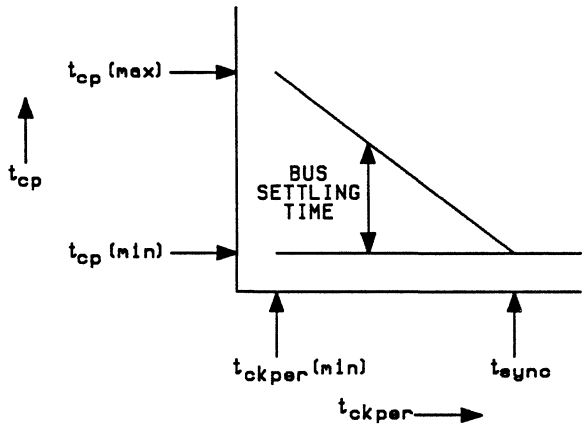


Figure 1 t_{cp} variation with t_{ckper}

Number formats

It is only a matter of the user's convention to decide where to place the "binary point" for input and output numbers. For instance, MACCGEN treats the products

01.11 * 001.1 = 00010.101 (1.75 * 1.5 = 2.625)

and 0111 * 0011 = 00010101 (7 * 3 = 21)

exactly the same. The most common number formats used are integer and fractional notation.

Integer notation places the binary point after the LSB of both the inputs and the output, whether unsigned or two's complement. Fractional notation, on the other hand, is different for unsigned and two's complement. On inputs, unsigned fractional places the binary point before the MSB, and two's complement fractional places it after the MSB. If acc_bits is 0, unsigned fractional notation will place the binary point before the MSB of the output, and two's complement fractional places it after the output's second most significant bit. If acc_bits is greater than 0, then "acc_bits" extra significant digits are added to the left. These formats are shown in Figure 2.

Integer Unsigned Inputs

X[nx−1]	X[nx−2]	...	X[1]	X[0]
2^{nx-1}	2^{nx-2}		2^{1}	2^{0}

Y[ny−1]	Y[ny−2]	...	Y[1]	Y[0]
2^{ny-1}	2^{ny-2}		2^{1}	2^{0}

Fractional Unsigned Inputs

X[nx−1]	X[nx−2]	...	X[1]	X[0]
2^{-1}	2^{-2}		2^{-nx+1}	2^{-nx}

Y[ny−1]	Y[ny−2]	...	Y[1]	Y[0]
2^{-1}	2^{-2}		2^{-ny+1}	2^{-ny}

Integer Two's Complement Inputs

X[nx−1]	X[nx−2]	...	X[1]	X[0]
-2^{nx-1}	2^{nx-2}		2^{1}	2^{0}

Y[ny−1]	Y[ny−2]	...	Y[1]	Y[0]
-2^{ny-1}	2^{ny-2}		2^{1}	2^{0}

Fractional Two's Complement Inputs

X[nx−1]	X[nx−2]	...	X[1]	X[0]
-2^{0}	2^{-1}		2^{-nx+2}	2^{-nx+1}

Y[ny−1]	Y[ny−2]	...	Y[1]	Y[0]
-2^{0}	2^{-1}		2^{-ny+2}	2^{-ny+1}

Integer Unsigned Output (nout = nx+ny+acc_bits)

P[nout−1]	P[nout−2]	P[nout−3]	...	P[2]	P[1]	P[0]
2^{nout-1}	2^{nout-2}	2^{nout-3}		2^{2}	2^{1}	2^{0}

Fractional Unsigned Output (nout = nx+ny+acc_bits)

P[nout−1]	P[nout−2]	P[nout−3]	...	P[2]	P[1]	P[0]
2^{acc_bits-1}	2^{acc_bits-2}	2^{acc_bits-3}		$2^{-nx-ny+2}$	$2^{-nx-ny+1}$	2^{-nx-ny}

Integer Two's Complement Output (nout = nx+ny+acc_bits)

P[nout−1]	P[nout−2]	P[nout−3]	...	P[2]	P[1]	P[0]
-2^{nout-1}	2^{nout-2}	2^{nout-3}		2^{2}	2^{1}	2^{0}

Fractional Two's Complement Output (nout = nx+ny+acc_bits)

P[nout−1]	P[nout−2]	P[nout−3]	...	P[2]	P[1]	P[0]
-2^{acc_bits+1}	2^{acc_bits}	2^{acc_bits-1}		$2^{-nx-ny+4}$	$2^{-nx-ny+3}$	$2^{-nx-ny+2}$

Figure 2

Product rounding

The extra adder input, A, provides a flexible way to do whatever product rounding you may need. If you don't need rounding, then setting the input parameter "add_in" to 0 will reduce the circuit area slightly.

Many applications do not need the full precision available from MACCGEN. In a 16x16 MAC, for instance, you may only be able to use the 16 most significant bits of the 32-bit product (assuming acc_bits = 0). To avoid a systematic bias due to truncation error, the result can be rounded by always adding a 1 at the 17th most significant bit. This is accomplished by tying pin A[15] to a logic HIGH, and tying the rest of the A bus to logic LOW. Since the rounding position and value are determined by external logic and not fixed by the MACCGEN compiler, rounding can be conditional and variable, allowing implementation of adaptive algorithms.

Two's Complement Rounding:

Rounding can be done slightly differently for two's complement numbers to get a "free" extra bit of precision. If it is noted that for all products except 10..00 x 10..00, the two most significant bits of the product are identical, then the portion of the product extracted may be right-shifted by one bit. (Assume acc_bits = 0 in the following discussion.)

In a 16x16 MAC, for example, the 16-bit product can be taken as bits 30 (MSB) through 15 (LSB), and rounding would then be done by adding a 1 at bit 14. Of course, to allow this extra bit of precision requires eliminating the value 100..00 as a valid input. In fractional two's complement notation, this corresponds to limiting the input values to the range

$$-0.1111.. \leq X,Y \leq 0.11111 \quad (-1 \text{ not allowed})$$

Multiplying −1 x −1 results in the product −1 using this method. Many standard multiplier parts (IDT IDT7216/17, Cypress CY7C516/17, etc.) include a Format Adjust to allow this mode of output.

Naming conventions

ViGen will automatically create a default cell name for each unique MACCGEN configuration. This name is encoded in the following manner:

MAC nx X ny U|T|S U|T|S N|A d

where U means tcx (tcy) = 0 (first x, then y)
T means tcx (tcy) = 1
S means tcx (tcy) = 2
N means add_in = 0
A means add_in = 1
d is the value of acc_bits

The MACs shown in the timing examples charts are therefore named:

MAC8X8TTA4
and MAC16X16UUN4

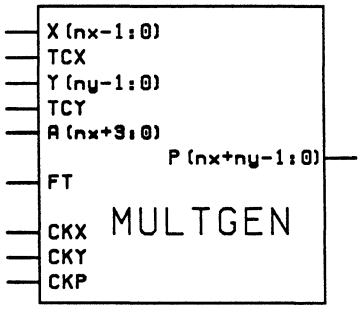
VS1500 Multiplier Generator

FEATURES

- Variable size compiled function, with X and Y inputs independently selectable from 4 to 32 bits wide
- High performance architecture uses Booth encoding and Wallace tree partial product summation
- Two's complement, unsigned magnitude, and mixed mode multiplication
- Input number format can be pin-selectable
- Extra adder input allows implementation of P = X x Y + A. This input is usually used for rounding
- X and Y inputs have separate registers and clocks
- Output is optionally registered with its own clock
- Output register can be pin-selectable

SYMBOL

The symbol for MULTGEN will be unique for each configuration. An example is given here only for reference.



INPUT PARAMETER RANGES

Input Parameter	Allowed Range	Explanation
nx	4 – 32 (even only)	Number of bits in the X input word
ny	4 – 32 (even, ny≤nx)	Number of bits in the Y input word
tcx	0, 1, or 2	=0: X in unsigned magnitude format =1: X in two's complement format =2: X input format selectable with an input pin (TCX)
tcy	0, 1, or 2	=0: Y in unsigned magnitude format =1: Y in two's complement format =2: Y input format selectable with an input pin (TCY)
add_in	0 or 1	Selects whether extra word A is to be added (1 = yes)
ft	0, 1 or 2	flow through: =0: output register included =1: no output register included =2: output register selectable with an input pin

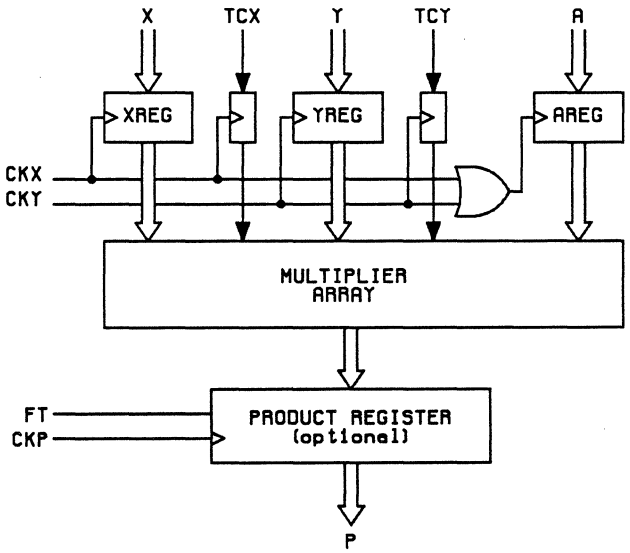
INPUTS/OUTPUTS

Pin/Bus Name (in TDL order)	Function	Req/ Opt	Cap (pF)
INPUTS:			
X[0:nx−1]	X input bus. Data on X is loaded into the X input register on the rising edge of CKX.	Req	0.164
Y[0:ny−1]	Y input bus. Data on Y is loaded into the Y input register on the rising edge of CKY.	Req	0.160
A[0:nx+3]	Optional adder input, usually used for rounding. Data on A is loaded into the A register on the rising edge of the logical OR of CKX and CKY. A is treated as a two's complement number, regardless of tcx and tcy selections. Exists if add_in=1.	Opt	0.060
CKX	X input clock, active on rising edge	Req	0.162
CKY	Y input clock, active on rising edge	Req	0.162
CKP	Clock input for the optional product register. Active on rising edge. Exists if ft=0 or ft=2.	Opt	0.110
FT	Flow−through control. When HIGH, the product register is transparent. When LOW, the product register is clocked by CKP. FT may only change when CKP is LOW. Exists if ft=2.	Opt	0.181
TCX	X input two's complement control. When HIGH, the X input is treated as a two's complement number. When LOW, X is treated as an unsigned magnitude number. Exists if tcx=2.	Opt	0.091
TCY	Y input two's complement control, similar to TCX. Exists if tcy=2.	Opt	0.100
OUTPUTS:			
P[0:nx+ny−1]	Output product bus, indicating the current contents of the product register if in clocked operation, or the product of the current x and y register contents if in flow−through mode. If clocked, P is updated after a rising CKP edge. If flow−through, P is updated after a rising CKX or CKY edge.		

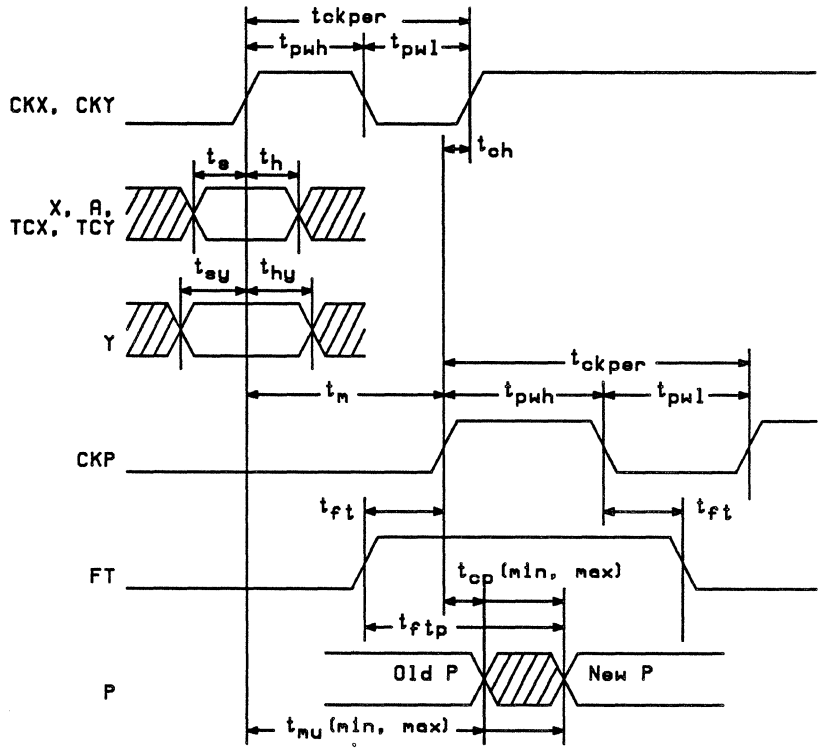
TIMING PARAMETERS

Name	Description
t_{ckper}	Minimum clock period for CKX, CKY, or CKP
t_m	Clocked multiply time (if Product Register is used)
t_{mu}	Unclocked multiply time (if Product Register isn't used). The minimum time is the hold time for the previous product, and the time between minimum and maximum is the bus settling time.
t_{cp}	Output delay time for P outputs after a rising CKP edge. The minimum time is the hold time for the previous product, and the time between minimum and maximum is the bus settling time.
t_s	Setup time for X, A, TCX, and TCY inputs before a rising CKX or CKY edge
t_h	Hold time for X, A, TCX, and TCY inputs after a rising CKX or CKY edge
t_{sy}	Setup time for Y inputs before a rising CKY edge
t_{hy}	Hold time for Y inputs after a rising CKY edge
t_{pwh}	Minimum high clock pulse width
t_{pwl}	Minimum low clock pulse width
t_{ch}	Relative hold time. CKP must rise at or before CKX or CKY to guarantee correct product output.
t_{ft}	Minimum transition margin from either CKP edge to changing FT
t_{ftp}	Maximum delay from FT rising edge to valid P output, assuming that t_{mu} is also satisfied

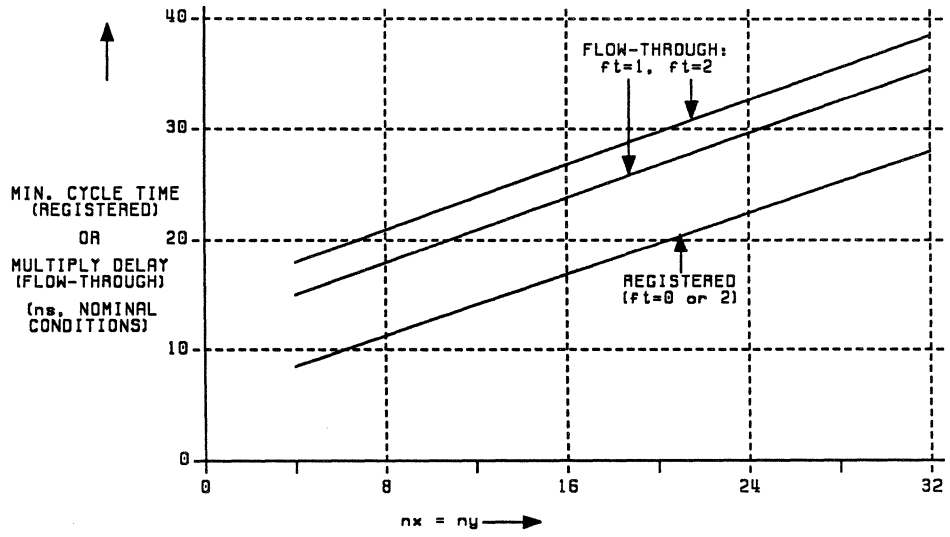
FUNCTIONAL BLOCK DIAGRAM



MULTGEN AC WAVEFORMS



MULTIPLICATION TIME FOR SQUARE MULTIPLIERS (nx = ny)



(See NCR ASIC Data Book for process, voltage, and temperature derating factors.)

TIMING, POWER, AND AREA EQUATIONS

1. Input parameters for the following equations are nx and ny. The validity of some equations depends on the input parameter ft. CL is the output capacitance in pF.
2. Timing parameters are specified for nominal process, V_{DD}=5.0 volts, and temperature = 25°C. See the NCR ASIC Data Book for information on derating factors for process, voltage and temperature.
3. All delays for rising and falling outputs are equal, while rise and fall times are individually specified.

PARAM	DESCRIPTION	TYPICAL VALUE (ns)
t_{ckper}	Min. clock period	t_{ckper} = 5.84 + 0.07*nx + 0.63*ny
t_m	Max. clocked multiply	t_m = 5.84 + 0.07*nx + 0.63*ny
$t_{mu(ft=1)}$	Unclocked multiply delay (flow-thru, ft=1)	$t_{mu(max)}$ = 12.08 + 0.10*nx + 0.63*ny + 0.792*CL $t_{mu(min)}$ = 7.08 + 0.792*CL
$t_{mu(ft=2)}$	Unclocked multiply delay (flow-thru, ft=2)	$t_{mu(max)}$ = 15.06 + 0.10*nx + 0.63*ny + 1.23*CL $t_{mu(min)}$ = 7.08 + 1.23*CL
t_{cp}	Output delay (clocked, ft=0 or 2) NOTE: See App. note #1	$t_{cp(max)}$ = 8.05 + 0.014*nx + 0.094*ny + 1.23*CL $t_{cp(min)}$ = 3.77 + 0.041*ny + 1.23*CL
t_s	X,A,TCX, and TCY setup	t_s = 1.30
t_h	X,A,TCX, and TCY hold	t_h = 1.3 + 0.034*nx
t_{sy}	Y setup	t_{sy} = 1.36
t_{hy}	Y hold	t_{hy} = 1.85 + 0.011*ny
t_{pwh}	Min. high clock pulse	t_{pwh} = 45% of t_{ckper}
t_{pwl}	Min. low clock pulse	t_{pwl} = 45% of t_{ckper}
t_{ch}	Relative hold time (clocked)	t_{ch} = 0
t_{ft}	FT transition margin	t_{ft} = 2.5
t_{ftp}	FT to P delay, assuming $t_{mu(max)}$ is also valid	t_{ftp} = $t_{cp(min)}$

Rise and Fall Times

ft=1:	t_r = 0.545 + 1.33*CL		ft=0 or 2:	t_r = 0.325 + 2.55*CL
	t_f = 0.506 + 1.05*CL			t_f = 0.279 + 1.99*CL

Current Requirements: I = 0.0174*ny^2 + 0.115*(nx−ny) mA/MHz
Worst case current process, V_{DD} = 5.5V, T = 25°C
Toggling 0x0 and FF..FFxFF..FF

Cell Width (mils) = 12.50 + 2.48*nx + 1.58*ny

Cell Height (mils) = 10.67 + 5.33*ny + 2.60*add_in + offset1 + offset2

where: offset1 = 0 if tcy=1
offset1 = 10.67 if tcy=0 or 2
offset2 = 0 if ft=1
offset2 = 6.97 if ft=0 or 2

TIMING EXAMPLES

8 x 8 two's complement multiplier with rounding, product register

(MUL8X8TTAR: nx = 8, ny = 8, tcx = 1, tcy = 1, add_in = 1, ft = 0)

Symbol	8 x 8 Multiplier	Nominal V_{DD} = 5V		Worst Case V_{DD} = 4.5V						Units
		TA = 25°C		TA = 70°C		TA = 85°C		TA = 125°C		
	Parameter	Min	Max	Min	Max	Min	Max	Min	Max	
t_{ckper}	Minimum clock period	11.44		21.1		22.0		24.7		ns
t_m	Maximum clocked multiply		11.44		21.1		22.0		24.7	ns
t_{cp}	Output delay from CKP	5.24	9.10	9.7	16.8	10.1	17.5	11.3	19.7	ns
t_s	Minimum setup time for X, A inputs	1.30		2.4		2.5		2.8		ns
t_h	Minimum hold time for X, A inputs	1.57		2.9		3.0		3.4		ns
t_{sy}	Minimum setup time for Y inputs	1.36		2.5		2.6		2.9		ns
t_{hy}	Minimum hold time for Y inputs	1.94		3.6		3.7		4.2		ns
t_{pwh}	Minimum high clock pulse	5.15		9.5		9.9		11.1		ns
t_{pwl}	Minimum low clock pulse	5.15		9.5		9.9		11.1		ns
t_{ch}	Relative hold time	0		0		0		0		ns

Switching characteristics (Input t_r, t_f = 1.4ns, CL = 0.15pF)

16 x 16 unsigned multiplier with no rounding, flow-through

(MUL16X16UUNF: nx = 16, ny = 16, tcx = 0, tcy = 0, add_in = 0, ft = 1)

Symbol	16 x 16 Multiplier	Nominal V_{DD} = 5V		Worst Case V_{DD} = 4.5V						Units
		TA = 25°C		TA = 70°C		TA = 85°C		TA = 125°C		
	Parameter	Min	Max	Min	Max	Min	Max	Min	Max	
t_{ckper}	Minimum clock period	17.04		31.4		32.8		36.8		ns
t_{mu}	Maximum unclocked multiply	7.20	23.88	13.3	44.1	13.9	46.0	15.5	51.6	ns
t_s	Minimum setup time for X inputs	1.30		2.4		2.5		2.8		ns
t_h	Minimum hold time for X inputs	1.84		3.4		3.6		4.0		ns
t_{sy}	Minimum setup time for Y inputs	1.36		2.5		2.6		2.9		ns
t_{hy}	Minimum hold time for Y inputs	2.03		3.7		3.9		4.4		ns
t_{pwh}	Minimum high clock pulse	7.67		14.1		14.8		16.6		ns
t_{pwl}	Minimum low clock pulse	7.67		14.1		14.8		16.6		ns

Switching characteristics (Input t_r, t_f = 1.4ns, CL = 0.15pF)

APPLICATION NOTES

Output delay calculations for product registers

This applications note applies only to those multipliers that have product registers, i.e., ft=0 or ft=2. If the flow-through parameter, ft, is equal to 2, then this note only applies when in registered mode (pin FT is LOW).

Due to the way that the product register is implemented within MULTGEN, the delay from CKP to P is actually dependent on the multiply cycle time. The delay, t_{cp}, is calculated with both a minimum and a maximum value. The minimum value of t_{cp} is the output hold time for the previous product P. Data on the P output bus will not change before t_{cp}(min). The maximum value of t_{cp}, as calculated, only applies when the multiplier is being run at maximum frequency (minimum rising CKX/CKY to rising CKP). If the multiplier is being run slower, then t_{cp}(max) actually gets less, as shown in Figure 1.

Above a certain multiplication period, referred to in Figure 1 as t_{sync}, t_{cp}(max) will have the same value as t_{cp}(min). In other words, all of the bits in the P bus will switch simultaneously. For multiplication periods between t_{ckper}(min) and t_{sync}, t_{cp}(max) derates linearly as shown.

The period t_{sync} is defined as:

$$t_{sync} = t_{ckper}\ (\text{min}) + 1.2^{*}\ (\ t_{cp}(\text{max}) - t_{cp}(\text{min})\ )$$

where t_{ckper}, t_{cp}(max), and t_{cp}(min) are all as defined in the parametric equations.

Although the t_{cp}(min) and t_{cp}(max) parameters for each output pin will be slightly different due to differing output capacitive loads, for simplicity the simulation model will assign the shortest t_{cp}(min) and the longest t_{cp}(max) values to the entire P bus. This effect will be minor in most situations.

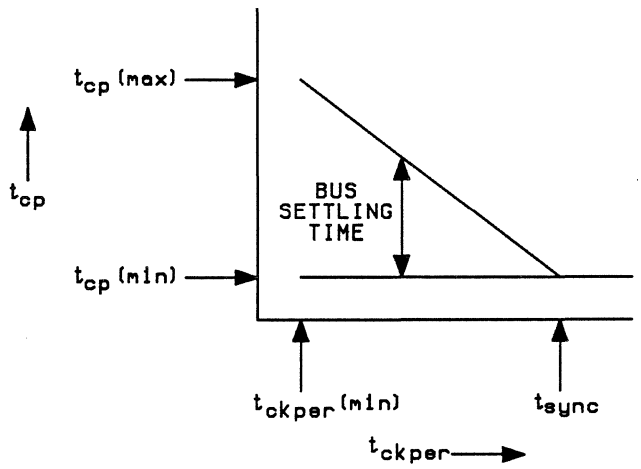


Figure 1 t_{cp} variation with t_{ckper}

Number formats

It is only a matter of the user's convention to decide where to place the "binary point" for input and output numbers. For instance, MULTGEN treats the products

01.11 * 001.1 = 00010.101 (1.75 * 1.5 = 2.625)

and 0111 * 0011 = 00010101 (7 * 3 = 21)

exactly the same. The most common number formats used are integer and fractional notation.

Integer notation places the binary point after the LSB of both the inputs and the output, whether unsigned or two's complement. Fractional notation, on the other hand, is different for unsigned and two's complement. Unsigned fractional places the binary point before the MSB, and two's complement fractional places it after the MSB for the inputs and after the second most significant bit for the output. These formats are shown in Figure 2.

Integer Unsigned Inputs

X[nx−1]	X[nx−2]	...	X[1]	X[0]
2^{nx-1}	2^{nx-2}		2^1	2^0

Y[ny−1]	Y[ny−2]	...	Y[1]	Y[0]
2^{ny-1}	2^{ny-2}		2^1	2^0

Fractional Unsigned Inputs

X[nx−1]	X[nx−2]	...	X[1]	X[0]
2^{-1}	2^{-2}		2^{-nx+1}	2^{-nx}

Y[ny−1]	Y[ny−2]	...	Y[1]	Y[0]
2^{-1}	2^{-2}		2^{-ny+1}	2^{-ny}

Integer Two's Complement Inputs

X[nx−1]	X[nx−2]	...	X[1]	X[0]
-2^{nx-1}	2^{nx-2}		2^1	2^0

Y[ny−1]	Y[ny−2]	...	Y[1]	Y[0]
-2^{ny-1}	2^{ny-2}		2^1	2^0

Fractional Two's Complement Inputs

X[nx−1]	X[nx−2]	...	X[1]	X[0]
-2^0	2^{-1}		2^{-nx+2}	2^{-nx+1}

Y[ny−1]	Y[ny−2]	...	Y[1]	Y[0]
-2^0	2^{-1}		2^{-ny+2}	2^{-ny+1}

Integer Unsigned Output (nout = nx+ny)

P[nout−1]	P[nout−2]	P[nout−3]	...	P[2]	P[1]	P[0]
2^{nout-1}	2^{nout-2}	2^{nout-3}		2^2	2^1	2^0

Fractional Unsigned Output (nout = nx+ny)

P[nout−1]	P[nout−2]	P[nout−3]	...	P[2]	P[1]	P[0]
2^{-1}	2^{-2}	2^{-3}		$2^{-nout+2}$	$2^{-nout+1}$	2^{-nout}

Integer Two's Complement Output (nout = nx+ny)

P[nout−1]	P[nout−2]	P[nout−3]	...	P[2]	P[1]	P[0]
-2^{nout-1}	2^{nout-2}	2^{nout-3}		2^2	2^1	2^0

Fractional Two's Complement Output (nout = nx+ny)

P[nout−1]	P[nout−2]	P[nout−3]	...	P[2]	P[1]	P[0]
-2^1	2^0	2^{-1}		$2^{-nout+4}$	$2^{-nout+3}$	$2^{-nout+2}$

Figure 2

Product rounding

The extra adder input, A, provides a flexible way to do whatever product rounding you may need. If you don't need rounding, then setting the input parameter "add_in" to 0 will reduce the circuit area slightly.

Many applications do not need the full precision available from MULTGEN. In a 16x16 multiplier, for instance, you may only be able to use the 16 most significant bits of the 32-bit product. To avoid a systematic bias due to truncation error, the result can be rounded by always adding a 1 at the 17th most significant bit. This is accomplished by tying pin A[15] to a logic HIGH, and tying the rest of the A bus to logic LOW. Since the rounding position and value are determined by external logic and not fixed by the MULTGEN compiler, rounding can be conditional and variable, allowing implementation of adaptive algorithms.

Two's Complement Rounding:

Rounding can be done slightly differently for two's complement numbers to get a "free" extra bit of precision. If it is noted that for all products except 10..00 x 10..00, the two most significant bits of the product are identical, then the portion of the product extracted may be right-shifted by one bit.

In the 16x16 multiplier, for example, the 16-bit product can be taken as bits 30 (MSB) through 15 (LSB), and rounding would then be done by adding a 1 at bit 14. Of course, to allow this extra bit of precision requires eliminating the value 100..00 as a valid input. In fractional two's complement notation, this corresponds to limiting the input values to the range

$$-0.1111.. \leq X,Y \leq 0.11111 \quad (-1 \text{ not allowed})$$

Multiplying −1 x −1 results in the product −1 using this method. Many standard multiplier parts (IDT IDT7216/17, Cypress CY7C516/17, etc.) include a Format Adjust to allow this mode of output.

Naming conventions

ViGen will automatically create a default cell name for each unique MULTGEN configuration. This name is encoded in the following manner:

MUL nx X ny U|T|S U|T|S N|A R|F|S

where U means tcx (tcy) = 0 (first x, then y)
T means tcx (tcy) = 1
S means tcx (tcy) = 2
N means add_in = 0
A means add_in = 1
R means ft = 0
F means ft = 1
S means ft = 2 (in the last character position)

The multipliers shown in the timing examples charts are therefore named:

MUL8X8TTAR
and MUL16X16UUNF

VS1500 Data Multiplexer

GENERAL DESCRIPTION

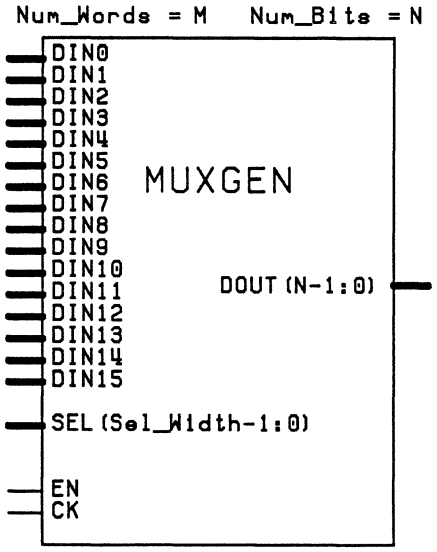
- The data multiplexer is an N-bit by M-word data selector/multiplexer
- Selectable decode: Binary or one-of-M
- Optional positive edge triggered flip-flops for latching the output bus
- Optional tristatable output bus

The multiplexer generator uses several OR-AND-INVERT type logic circuits whose outputs are OR'ed together. When more than one select is high in the one-of-M decode configuration, the data inputs are logically OR'ed together. All inputs and outputs are buffered for low input capacitance and high drive. This multiplexer is intended primarily for use in bus type applications where several multiplexers are controlled by the same set of signals. For NUM_BITS = 1, more efficient implementations can be built with standard cells. The greater the number of output bits, the more appropriate the use of the MUXGEN cell.

SYMBOL

M = number of words (4-16)
N = number of bits per bus (1-16)

The symbol for MUXGEN will be unique for each configuration. An example is given here only for reference.



INPUT PARAMETER RANGES

INPUT PARAMETER	ALLOWED RANGE	EXPLANATION
NUM_WORDS	4-16	Number of buses multiplexed per output bit.
NUM_BITS	1-16	Number of output bits.
TRISTATE_OPT	0 or 1	0 denotes always driving outputs. 1 denotes tristatable outputs.
BINARY_OPT	0 or 1	0 denotes one-of-M decode. 1 denotes binary decode.
CK_OPT	0 or 1	0 denotes unclocked outputs. 1 denotes positive edge triggered D-type flip-flop on output.

MUXGEN

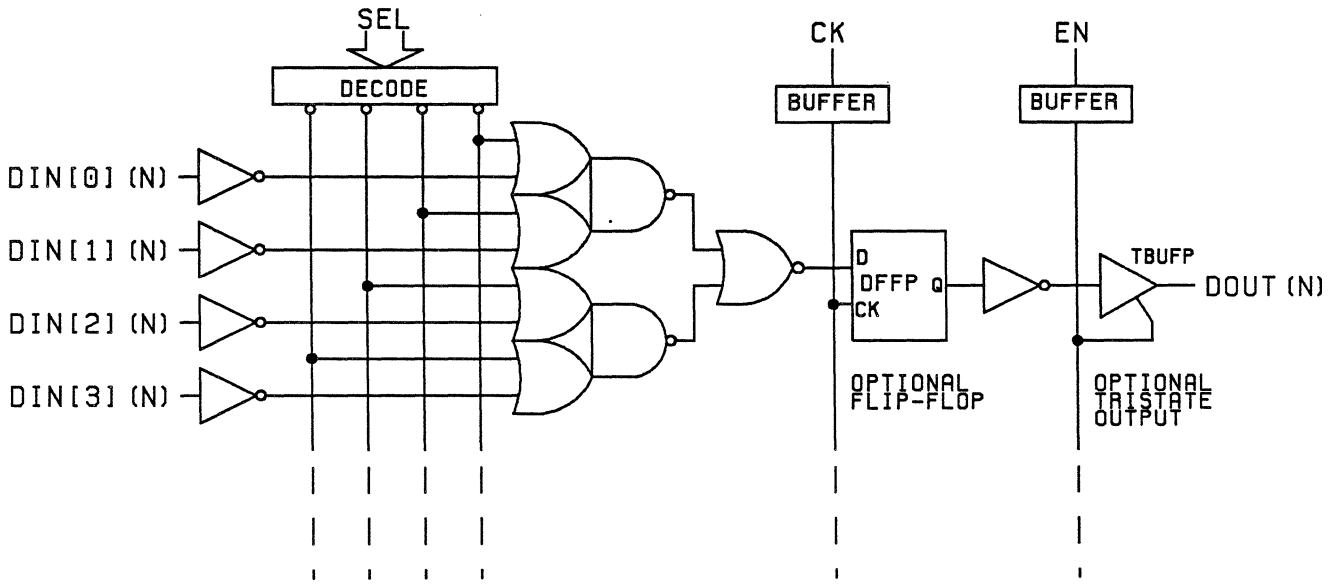
ViGen CONFIGURABLE FUNCTION

INPUTS/OUTPUTS

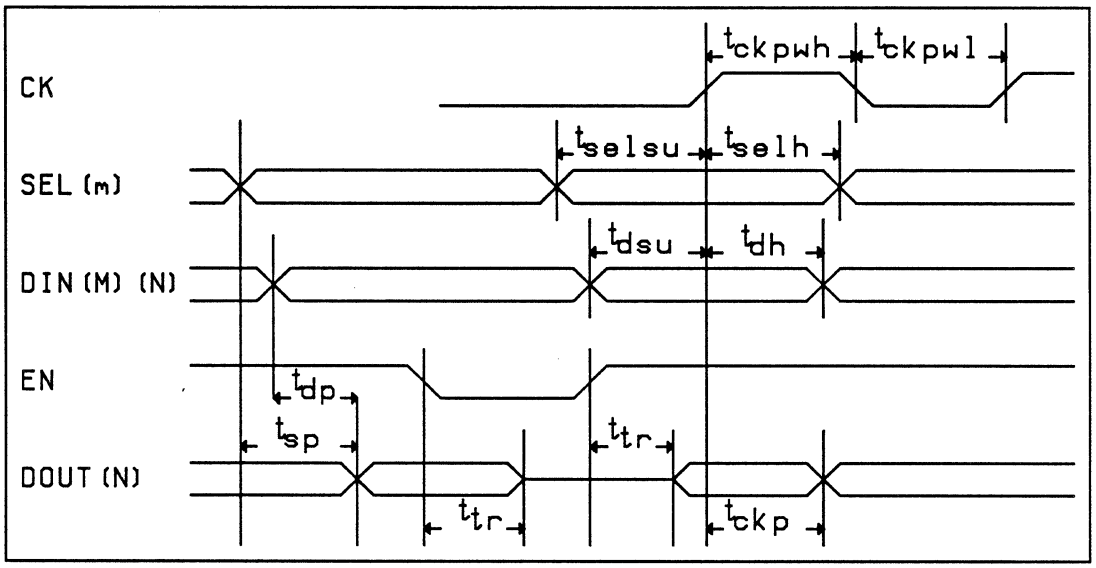
Definitions of the MUXGEN inputs and outputs are given in the following table.

PIN NAME	FUNCTION	REQ/ OPT	CAP (pF)
INPUTS:			
DIN (M) (N)	Data inputs: M words, N bits per word.	REQ	0.066
SEL (m)	Select lines, active high, m = Log_2 (M) for binary decode or m = M for 1–of–M decode.	REQ	0.064 0.077
EN	Tristate enable pin, tristates when low.	OPT	0.079
CK	Clock pin to latch output data.	OPT	0.078
OUTPUTS:			
DOUT (N)	Output bus		0.164

FUNCTIONAL BLOCK DIAGRAM (1–bit slice)



AC WAVEFORMS



MUXGEN

ViGen CONFIGURABLE FUNCTION

TIMING PARAMETERS AND CELL SIZE AS FUNCTIONS OF INPUT PARAMETERS

1. The input parameters are: NUM_WORDS, NUM_BITS, BINARY_OPT, CK_OPT, TRISTATE_OPT. A list of allowed ranges for use in the timing equations is given on the first page of the MUXGEN data sheet. CL is output capacitance in pF.
2. Timing parameters are specified for nominal process, Vdd=5V, and temperature = 25°C. See the NCR ASIC Data Book for information on derating factors for process, voltage and temperature. The t_{plh} notation refers to the output switching from low to high, and t_{phl} from high to low.

PARAM	DESCRIPTION	TYPICAL VALUE (ns)
t_{dp}	Data to output propagation delay (unclocked)	t_{plh} = 4.36 + 0.094*NUM_WORDS + 0.345*TRISTATE_OPT + (0.54 + 0.161*TRISTATE_OPT)*CL t_{phl} = 2.91 + 0.204*NUM_WORDS + 0.345*TRISTATE_OPT + (0.632 + 0.104*TRISTATE_OPT)*CL
t_{sp}	Select to output propagation delay (unclocked)	t_{plh} = 4.82 + (0.168*BINARY_OPT + 0.147)*NUM_WORDS + 0.079*NUM_BITS + 0.345*TRISTATE_OPT + (0.540 + 0.161*TRISTATE_OPT)*CL t_{phl} = 4.80 + (0.150*BINARY_OPT + 0.193)*NUM_WORDS + 0.098*NUM_BITS + 0.345*TRISTATE_OPT + (0.632 + 0.104*TRISTATE_OPT)*CL
t_{tr}	ENOUT to tristate on or off delay	t_{plh} = 1.79 + 0.042*NUM_BITS + 0.678*CL t_{phl} = 1.39 + 0.050*NUM_BITS + 0.828*CL
t_{ckp}	Rising clock edge to output propagation delay	t_{plh} = 5.48 + 0.042*NUM_BITS + 0.345*TRISTATE_OPT + (0.540 + 0.161*TRISTATE_OPT)*CL t_{phl} = 3.52 + 0.027*NUM_BITS + 0.345*TRISTATE_OPT + (0.632 + 0.104*TRISTATE_OPT)*CL
t_{dsu}	Data setup time	3.68 + 0.120*NUM_WORDS - 0.019*NUM_BITS
t_{dh}	Data hold time	0
t_{selsu}	Select setup time	4.58 + (0.150*BINARY_OPT + 0.193)*NUM_WORDS + 0.079*NUM_BITS
t_{selh}	Select hold time	0
t_{ckpwh}	Min clock pulse (high)	3.45 + 0.039*NUM_BITS
t_{ckpwl}	Min clock pulse (low)	3.45 + 0.039*NUM_BITS

Cell Width (mils) = 3.97 + 1.32*NUM_WORDS + 2.85*CK_OPT + 0.78*TRISTATE_OPT

Cell Height (mils) = 5.26 + BINARY_OPT*(decode) + ((tracks + 3)*0.259 + bitslice)*NUM_BITS

where:

decode = 4.61 for NUM_WORDS ≤ 4
6.92 for NUM_WORDS = 5–7
9.21 for NUM_WORDS = 8
10.32 for NUM_WORDS ≥ 9

tracks = integer portion of (NUM_WORDS/2)

bitslice = 1.67 for NUM_WORDS < 10
1.85 for NUM_WORDS ≥ 10

TIMING EXAMPLES

NUM_WORDS=8, NUM_BITS=8,
TRISTATE_OPT=0, BINARY_OPT=1, CK_OPT=0

8 WORD, 8-BIT, BINARY DECODE MULTIPLEXER			NOMINAL VDD=5V		WORST CASE VDD=4.5V						UNITS
			TA=25C		TA=70C		TA=85C		TA=125C		
SYMBOL	PARAMETER		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t_{sp}	Propagation time (SEL to DOUT)	t_{plh}		8.2		16.8		17.6		19.7	ns
		t_{phl}		8.6		17.7		18.4		20.6	ns
t_{dp}	Propagation time (DIN to DOUT)	t_{plh}		5.4		11.1		11.6		13.0	ns
		t_{phl}		4.9		10.1		10.5		11.8	ns

Switching Characteristics (Input tr,tf=1.4ns, CL=0.5pF)

NUM_WORDS=8, NUM_BITS=4,
TRISTATE_OPT=1, BINARY_OPT=0, CK_OPT=1

8 WORD, 4-BIT, CK_OPT, TRISTATE_OPT MULTIPLEXER			NOMINAL VDD=5V		WORST CASE VDD=4.5V						UNITS
			TA=25C		TA=70C		TA=85C		TA=125C		
SYMBOL	PARAMETER		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t_{ckp}	Propagation time (CK to DOUT)	t_{plh}		6.4		13.1		13.7		15.4	ns
		t_{phl}		4.4		9.0		9.4		10.6	ns
t_{tr}	ENOUT to tristate on or off delay	t_{plh}		2.3		4.7		4.9		5.5	ns
		t_{phl}		2.0		4.1		4.3		4.8	ns
t_{ckpwl}	CK low pulse width		3.6		7.4		7.7		8.6		ns
t_{ckpwh}	CK high pulse width		3.6		7.4		7.7		8.6		ns
t_{selsu}	SEL setup time before CK		7.6		15.6		16.3		18.2		ns
t_{selh}	SEL hold time		0		0		0		0		ns
t_{dsu}	DIN setup time before CK		4.6		9.4		9.8		11.0		ns
t_{dh}	DIN hold time		0		0		0		0		ns

Switching Characteristics (Input tr,tf=1.4ns, CL=0.5pF)

MUXGEN

ViGen CONFIGURABLE FUNCTION

APPLICATION NOTES

Area Calculations

Because of a layout approach that uses automatic place and route and compaction techniques, it is impossible to predict cell area exactly without actually creating the cell. Although the cell area equations are usually accurate to within 10%, in some instances the estimates can be off by as much as 20%.

Architecture and Primary Uses

The multiplexer generator is built around an OR–AND–INVERT NOR multiplexer architecture. All input data, select and control lines are buffered so input capacitance is a fixed value for all inputs and will not change with configuration. The select buffers are optimized to drive wide input buses. This makes the multiplexer well suited for bus applications where several multiplexers are all controlled by the same set of control lines. This also means that MUXGEN is not suited for most single bit multiplexer applications due to the large area and delay overhead of this multiple buffering scheme. If the binary decode option is chosen, the select inputs are buffered with an inverter before driving a NAND decoder which in turn drives the select line buffers. The output of the OAI/NOR multiplexer normally is buffered with a high drive buffer, although a tristate buffer will be substituted if the tristate option is chosen. When the tristate option is chosen, the output is driven when EN is high and is high impedance if EN is low. A positive edge triggered flip–flop is placed between the NOR gate and output buffer if the clock option is chosen. The inclusion of the flip–flop will add setup and hold requirements to the input data and select signals relative to the rising clock edge.

Naming Conventions

ViGen will automatically create a default cell name for each unique MUXGEN configuration with the configuration information encoded in the following manner:

MUX *num_words X num_bits X tristate_opt binary_opt ck_opt*

Therefore, the default names for the two example multiplexers would be:

MUX8X8X010
MUX8X4X101

Limitations

QUICKSIM Save and Restore functions will not currently operate with generated cells.

VS1500 RAM Generator

FEATURES

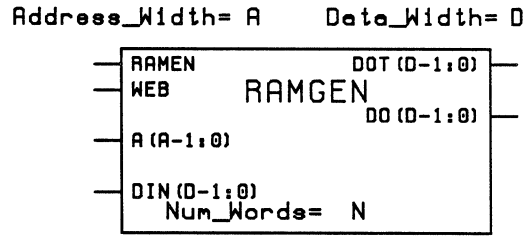
- Modular RAM allows variable configurations up to 32K bits.
- Pseudo static for reduced complexity and operating power (0 dc power)
- Enable input is taken low between each read and write access. This feature is applicable for clocked operation.
- Tristate and always-driving outputs

DESCRIPTION

The RAM generator produces a low power pseudo static RAM. On reads, the RAM enable input (RAMEN) is taken low in order to precharge internal states. When RAMEN input is taken high, the sense amp quickly discharges outputs when reading a logic low. On writes, RAMEN is initially taken low in order to allow address inputs to change. Data from the data input bus is written while RAMEN is high. Many internal states are only precharged to $V_{DD} - V_{threshold}$ in order to reduce power ($=CV^2F$) and decrease read discharge time. To further reduce ac power dissipation when the RAM is disabled, all inputs are internally gated by RAMEN. By externally tying a tristatable output to a data input, a bidirectional I/O line can be obtained.

SYMBOL

The symbol for RAMGEN will be unique for each configuration. An example is given here only for reference. Data_Width = Word_Width and Address_Width = $\lceil \log_2 \text{Num\_Words} \rceil$



INPUT PARAMETER RANGES

Input Parameter	Allowed Range	Explanation
Num_Words	8 – 2048 (even only)	Number of words in RAM array
Word_Width	Any integer ≤32	Number of bits in a word
	8 ≤bits ≤32768	Number of bits = Num_Words x Word_Width

RAMGEN

ViGen CONFIGURABLE FUNCTION

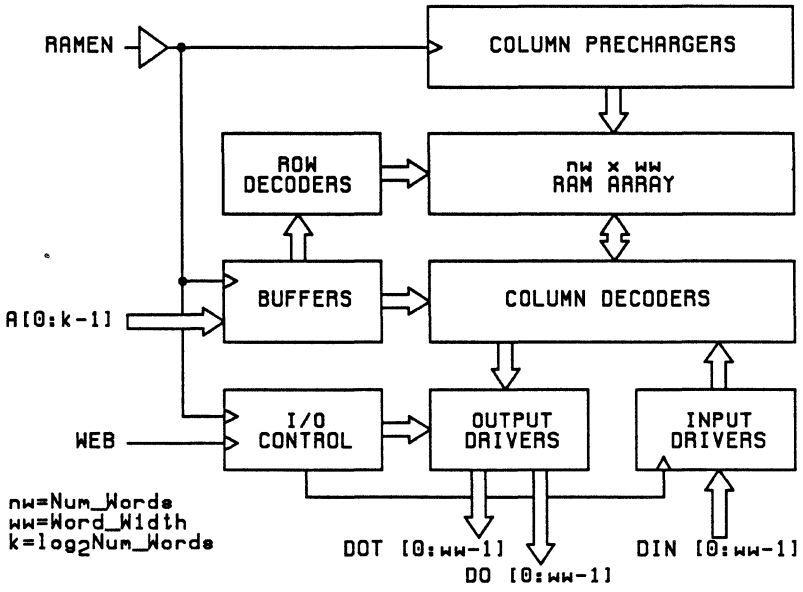
INPUTS/OUTPUTS

Pin/Bus Name (in TDL order)	Function	Cap (pF)
INPUTS:		
RAMEN	RAM enable is active high. Internal states are precharged when RAMEN is low. Reads and writes are enabled when RAMEN is high.	0.377
WEB	Read/Write select: active high for read, active low for write.	0.254
A[0:k−1]	Address input bus of width k ($k = \lceil \log_2 Num\_Words \rceil$). Addresses may only change while RAMEN is low.	0.093
DIN[0:Word_Width−1]	Data input bus.	0.302
OUTPUTS:		
DO[0:Word_Width−1]	Data output bus, driven by RAM (on read) and DIN (on write). Goes high on low RAMEN input.	
DOT[0:Word_Width−1]	Tristate data output bus is driven when WEB and RAMEN inputs are high. Otherwise DOT is at high impedance.	0.190

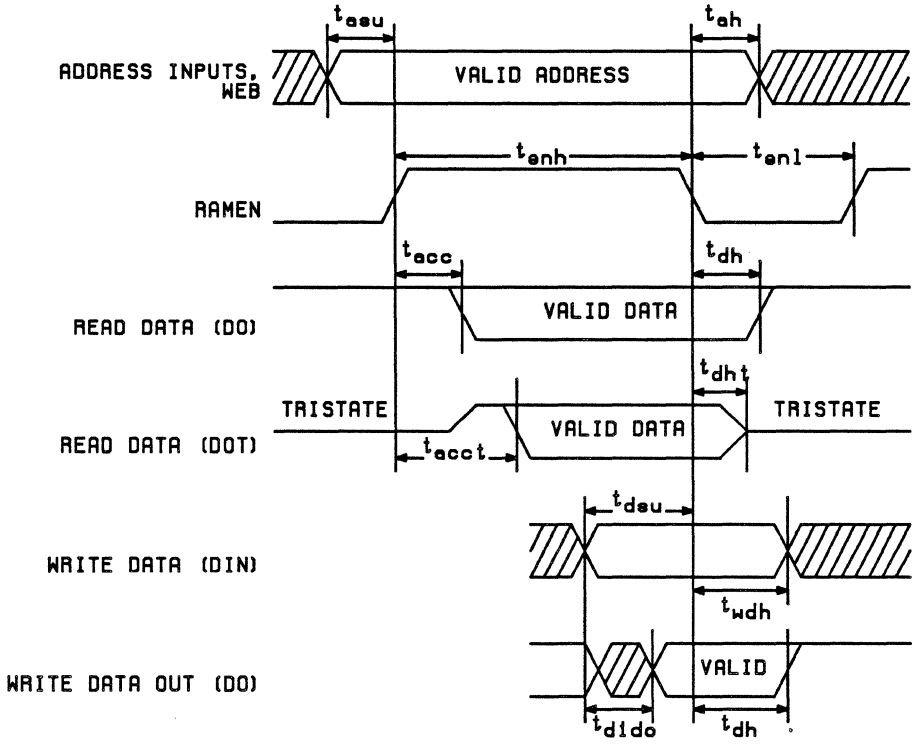
TIMING PARAMETERS

Name	Description
t_{asu}	Minimum address/WEB setup time before RAMEN rises (Read, Write)
t_{ah}	Minimum address/WEB hold time after RAMEN falls (Read, Write)
t_{enl}	Minimum RAMEN low pulse width (Read, Write)
t_{enh}	Minimum RAMEN high pulse width (Read, Write)
t_{acc}	Maximum delay from rising RAMEN to valid output data on DO (Read)
t_{dh}	Minimum delay from falling RAMEN during which DO data remains valid (Read, Write)
t_{acct}	Maximum delay from rising RAMEN to valid output data on DOT (Read)
t_{dht}	Minimum delay from falling RAMEN until DOT outputs at high impedance (Read)
t_{dsu}	Minimum input data (DIN) setup time before RAMEN falls (Write)
t_{wdh}	Minimum input data (DIN) hold time after RAMEN falls (Write)
t_{dido}	Maximum delay from data in to data out while RAMEN high (Write)
t_r	Maximum output rise time on DO
t_f	Maximum output fall time on DO
t_{rt}	Maximum output rise time on DOT (Read)
t_{ft}	Maximum output fall time on DOT (Read)

FUNCTIONAL BLOCK DIAGRAM



RAMGEN AC WAVEFORMS



TIMING, POWER, AND AREA EQUATIONS

Equations are given in terms of several variables which describe a RAM's characteristics. To solve the timing, power, and area equations given below, first determine RAM variables by following these steps:

1. Num_Words and Word_Width are generator input parameters.
2. CL is output capacitance in pF.
3. col_dec = internal column decode. Possible values for col_dec are 4, 8, and 16. RAMGEN chooses smallest col_dec which is $\geq \sqrt{\text{Num\_Words / Word\_Width}}$. If $\sqrt{\text{Num\_Words / Word\_Width}} > 8$, use col_dec = 16.
4. Num_Words is internally rounded up to the nearest multiple of 2*col_dec.
5. rows = internal array rows = Num_Words / col_dec.
6. cols = internal array columns = Word_Width*col_dec.
7. Lr = $\lceil \log_2 \text{rows} \rceil$. (ceiling function: round up to nearest integer)

All times are in nanoseconds and are NOMINAL (V_{DD} = 5.0 volts, T = 25C, Nom. process). See the NCR ASIC Data Book for process, voltage, and temperature derating.

PARAM	DESCRIPTION	TYPICAL VALUE (ns)
t_{asu}	Address/WEB setup	= 0
t_{ah}	Address/WEB hold	= 0.74 + 0.110*Lr + 0.020*Word_Width
t_{enl}**	RAMEN low pulse width	= 4.58 − 0.100*Lr + 0.010*cols + 0.090*rows
t_{enh}*	RAMEN high pulse width	= t_{acc}
t_{acc}*	DO access	= 5.72 − 0.300*Lr + 0.020*cols + 0.076*rows +0.232*CL
t_{dh}	DO hold	= 2.25 + 0.214*Lr + 0.104*Word_Width + 0.390*CL
t_{acct}*	DOT access	= 5.73 − 0.295*Lr + 0.021*cols + 0.076*rows + 0.605*CL
t_{dht}	DOT hold	= 1.25 + 0.119*Word_Width
t_{dsu}*	Write data setup	= 2.68 + 0.026*rows
t_{wdh}	Write data hold	= 0.67 + 0.065*Lr + 0.098*Word_Width
t_{dido}	Data in to data out	= 5.62 + 0.020*cols + 0.067*rows + 0.452*CL
t_r	DO rise	= 0.58 + 0.840*CL
t_f	DO fall	= 0.57 + 0.900*CL
t_{rt}	DOT rise	= 0.77 + 2.230*CL
t_{ft}	DOT fall	= 0.49 + 0.930*CL

* If col_dec equals 4, then 0.5 ns can be subtracted from t_{enh}, t_{acc}, t_{acct}, and t_{dsu}.
** t_{enl} equals maximum of t_{enl} and t_{dh}.

Current Requirements: I = 4.54*rows + 6.32*cols + .0281*rows*cols (μA/MHz)
Worst case current process, V_{DD} = 5.5V, T = 25°C

Cell Height Estimate (mils) = 10.6 + 1.22*rows

Cell Width Estimate (mils) = 6.29 + offset + 0.826*cols

if 04 ≤cols < 40 use offset = 2.50
if 40 ≤cols < 80 use offset = 3.53
if 80 ≤cols ≤256 use offset = 4.87

TIMING EXAMPLES

512 x 8 RAM: col_dec = 8, rows = 64, cols = 64, Lr = 6

Symbol	512 x 8 RAM	Nominal V_{DD} = 5V		Worst Case V_{DD} = 4.5V						Units
		TA = 25°C		TA = 70°C		TA = 85°C		TA = 125°C		
	Parameter	Min	Max	Min	Max	Min	Max	Min	Max	
t_{asu}	Address/WEB set-up	0		0		0		0		ns
t_{ah}	Address/WEB hold	1.56		2.92		3.04		3.40		ns
t_{enl}	RAMEN low pulse width	10.38		19.41		20.24		22.63		ns
t_{enh}	RAMEN high pulse width	10.10		18.88		19.69		22.02		ns
t_{acc}	DO access		10.10		18.88		19.69		22.02	ns
t_{dh}	DO hold	4.42		8.27		8.63		9.65		ns
t_{acct}	DOT access		10.26		19.18		20.00		22.36	ns
t_{dht}	DOT hold	2.20		4.12		4.29		4.80		ns
t_{dsu}	Write data set-up	4.34		8.12		8.47		9.47		ns
t_{wdh}	Write data hold	1.84		3.45		3.60		4.02		ns
t_{dido}	Data in to data out		11.26		21.05		21.95		24.54	ns

Switching characteristics (Input t_r, t_f = 1.4ns, CL = 0.15pF)

64 x 4 RAM: col_dec = 4, rows = 16, cols = 16, Lr = 4

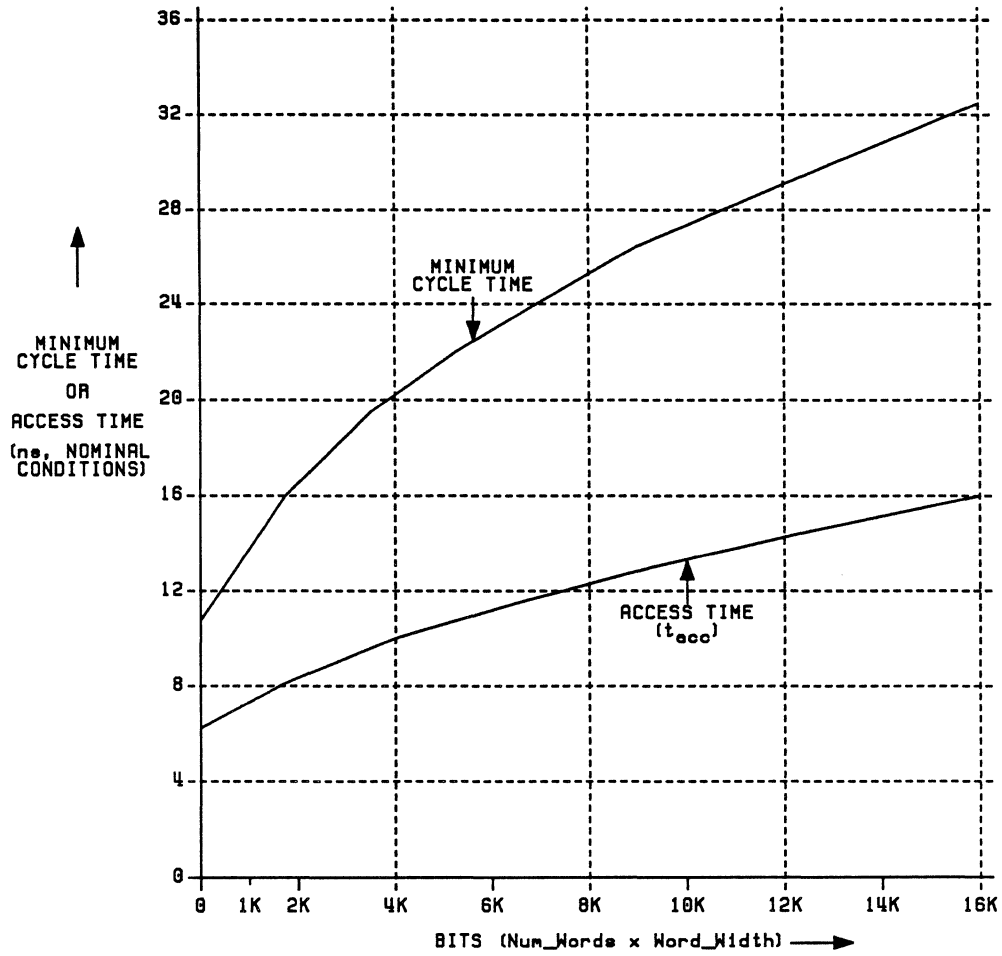
Symbol	64 x 4 RAM	Nominal V_{DD} = 5V		Worst Case V_{DD} = 4.5V						Units
		TA = 25°C		TA = 70°C		TA = 85°C		TA = 125°C		
	Parameter	Min	Max	Min	Max	Min	Max	Min	Max	
t_{asu}	Address/WEB set-up	0		0		0		0		ns
t_{ah}	Address/WEB hold	1.26		2.36		2.46		2.75		ns
t_{enl}	RAMEN low pulse width	5.78		10.81		11.27		12.60		ns
t_{enh}*	RAMEN high pulse width	5.59		10.45		10.90		12.19		ns
t_{acc}*	DO access		5.59		10.45		10.90		12.19	ns
t_{dh}	DO hold	3.58		6.70		6.98		7.81		ns
t_{acct}*	DOT access		5.69		10.65		11.10		12.41	ns
t_{dht}	DOT hold	1.73		3.23		3.37		3.76		ns
t_{dsu}*	Write data set-up	2.60		4.85		5.06		5.66		ns
t_{wdh}	Write data hold	1.32		2.47		2.58		2.88		ns
t_{dido}	Data in to data out		7.08		13.24		13.81		15.43	ns

Switching characteristics (Input t_r, t_f = 1.4ns, CL = 0.15pF)

*Since col_dec equals 4, 0.5 ns (Nominal) was subtracted from t_{enh}, t_{acc}, t_{acct}, and t_{dsu}.

ACCESS AND CYCLE TIME for square arrays (rows = columns)

Graph slightly overestimates for array columns > rows.
Cycle time = $t_{enl} + t_{enh}$



(See NCR ASIC Data Book for process, voltage, and temperature derating factors.)

APPLICATION NOTES

Designing for testability

Test programs for ASIC components must be able to verify that all circuit elements in the design are functioning properly. This includes verifying the functional correctness of the design and detecting faults caused by manufacturing defects. There are many methods to improve the testability of a component, all of which involve trade-offs in the amount of extra logic required, the resulting test time, and the degree of test coverage. A highly recommended method for the RAM is to multiplex the address, control, and data lines to the external part pins during a test mode. This allows direct control and observability of the RAM during test. In pin limited situations, the use of scan registers to shift in the address and data and to shift out the output may be desired.

To adequately test the RAM, the following conditions and patterns should be considered:

- Tying address and data input lines together while writing provides an excellent test of every data value. Verify by comparing each address with address content. This pattern will catch faults in the address decode. Repeat test with address complements written through data inputs. This ensures that logic one and zero are written to every bit and verified. Preferably, all bits should be written first and then read.
- The data output should change frequently between consecutive reads. The above pattern or a "checkerboard" pattern is a good example of this.
- Operate RAM at near maximum frequency to verify that sense amps and other critical timing circuitry are working properly.
- During reads to verify bit contents, set data inputs to a value other than the value being read (such as all 0/1). This verifies that outputs are not being driven by inputs during reads.
- Verify tristate control by reading through tristate outputs.

Column address inputs are the least significant bits of the address. If the RAM is operated as specified, then the critical race conditions which can cause write or read disturb problems will not exist in this design.

Output drive

The data outputs (DO) have high output drives (approximately equal to the drive of an HBUF). Through the tristate outputs (DOT), the drive is about half the data output drive (slightly less than the drive of a TBUFP).

Naming conventions

ViGen will automatically create a default cell name for each unique RAMGEN configuration. This name is encoded in the following manner:

RAMGEN <WORDS> X <WORDBITS>

Therefore, the default names for the two example RAMs would be:

RAMGEN512X8 and RAMGEN64X4.

ROM m x n

ViGen CONFIGURABLE FUNCTION

VS1500 ROM Generator

GENERAL DESCRIPTION

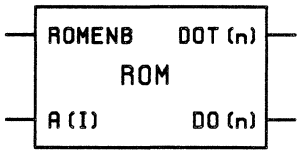
- Modular ROM allows flexible organization
- Clocked operation suited to microcomputer applications
- Access time is less than 57 ns

Inputs: ROMENB, A(I)
Outputs: DO(n), DOT(n)
Input Cap.: Address Inputs = 0.12pF
ROMENB = 0.18pF
Output Cap.: 0.095pF
Cell Size: See 1.5μ ROM Application Notes

SYMBOL

m = number of words (128–8192)
n = word width (1–16)
I = number of address inputs (7–13), where $I = \lceil \log_2(m) \rceil$

The symbol for the ROM will be unique for each workstation. An example is given here only for reference.



The modular ROM supercell is pseudo-static for reduced complexity and operating power. The enable input (ROMENB) must be taken high between each access. The memory array is organized as "m" words, where each word has "n" bits. "m" can be any multiple of 128, while "n" can be any integer. Both tristatable and always driving outputs are provided.

INPUT PARAMETER RANGES

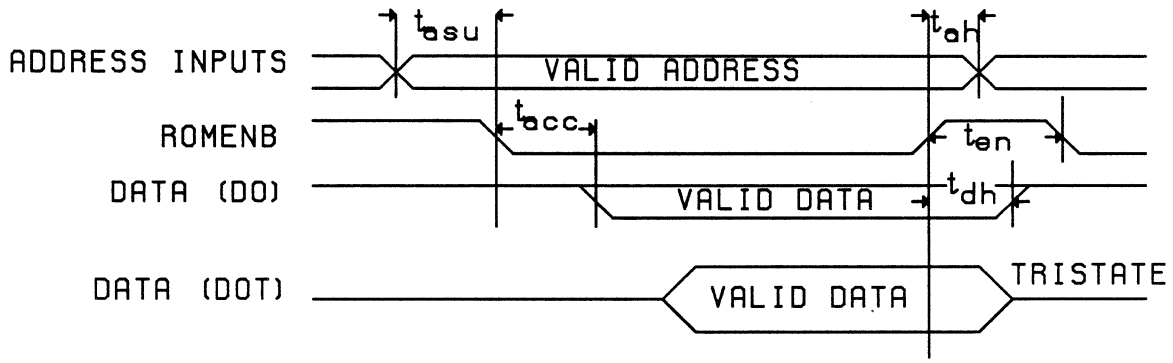
Input Parameter	Allowed Range	Explanation
Number of words (m)	128–8192	Number of words (number of bits = word width x number of words)
Word width (n)	1–16	Word size
Number of address inputs (I)	7–13	Address input bus width

(Input t_r, t_f =1.75 ns, C_L =5.0 pF)

SYMBOL	PARAMETER	NOMINAL V_{DD} = 5V		WORST CASE V_{DD} = 4.5V						UNIT
		T_A = 25C		T_A = 70C		T_A = 85C		T_A = 125C		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t_{ASU}	Address Setup Time before ROMENB (Read)	0	–	0	–	0	–	0	–	ns
t_{AH}	Address Hold Time after ROMENB	3.2	–	5.6	–	6	–	6.4	–	ns
t_{ENL}	ROMENB Low Pulse Width	–	–	*	–	*	–	*	–	ns
t_{ENH}	ROMENB High Pulse Width	8.8	–	17.6	–	18.4	–	20.8	–	ns
t_{ACC}	ROMENB to DO Output (Read)	–	–	–	*	–	*	–	*	ns
t_{DH}	DO Outputs HOld Time (R,W)	4	10.4	8	20	8.8	20.8	9.6	24	ns
t_{ACCT}	ROMENB to DOT Output (Read)	–	–	–	**	–	**	–	**	ns
t_{DHT}	DOT Outputs Hold Time (Read)	2.4	12	4.8	24	5.2	24.8	5.6	28	ns

* See Figure 1
** See Figure 2

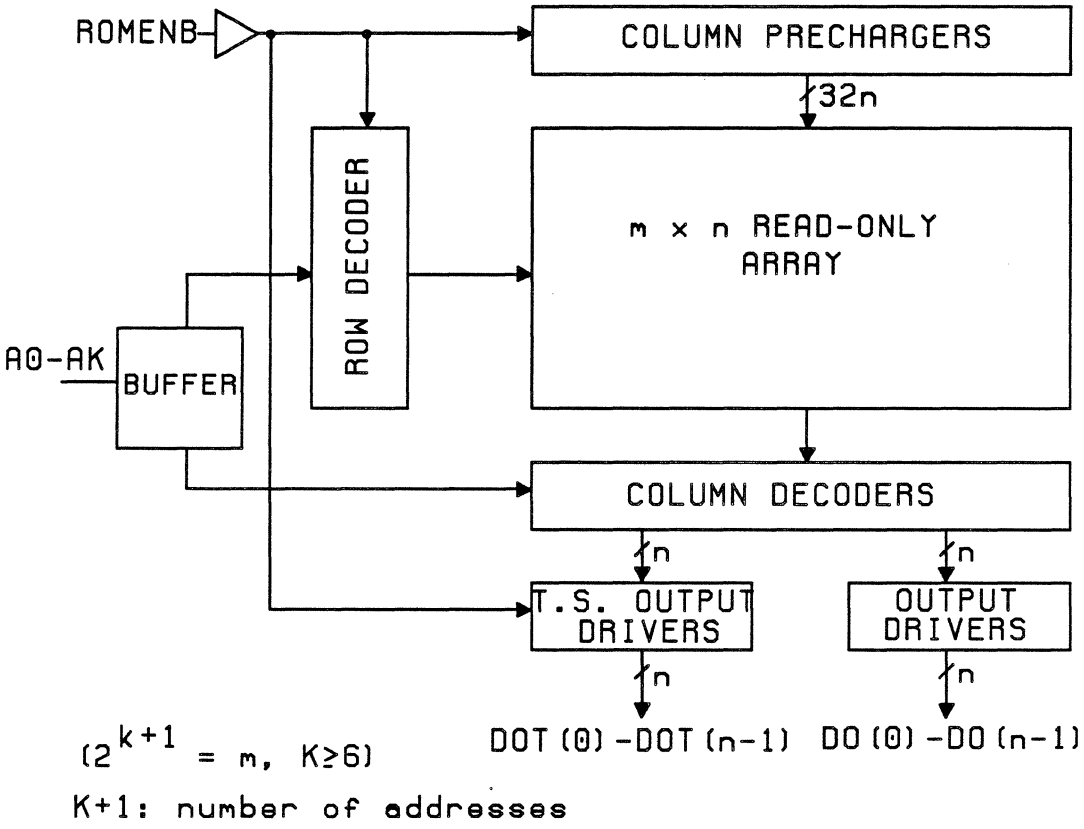
AC WAVEFORMS



ROM m x n

ViGen CONFIGURABLE FUNCTION

FUNCTIONAL BLOCK DIAGRAM



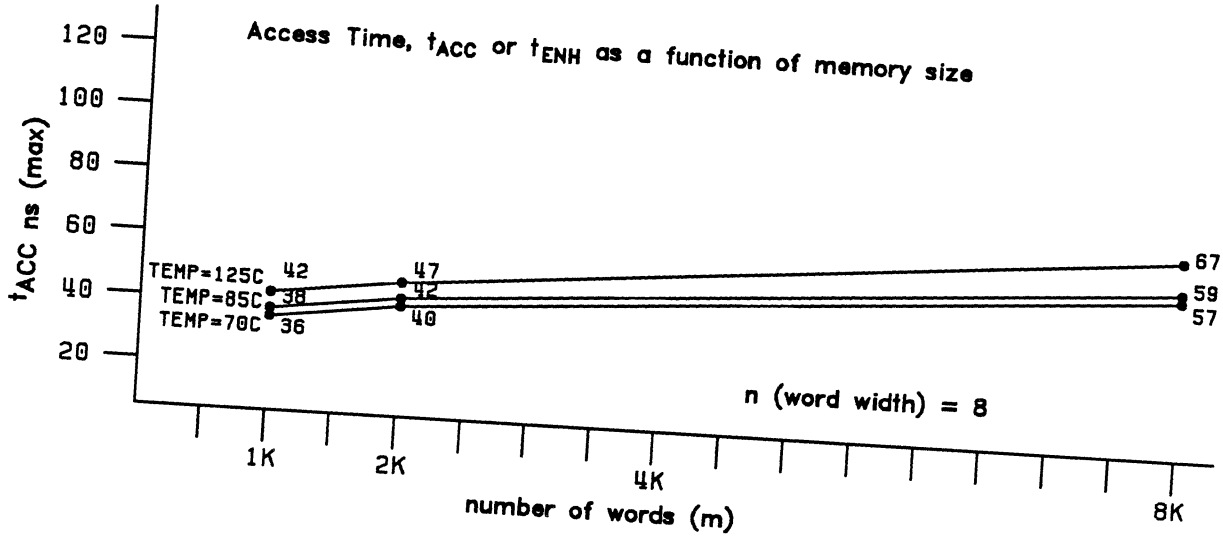


Figure 1 ROM Access time (DO) output

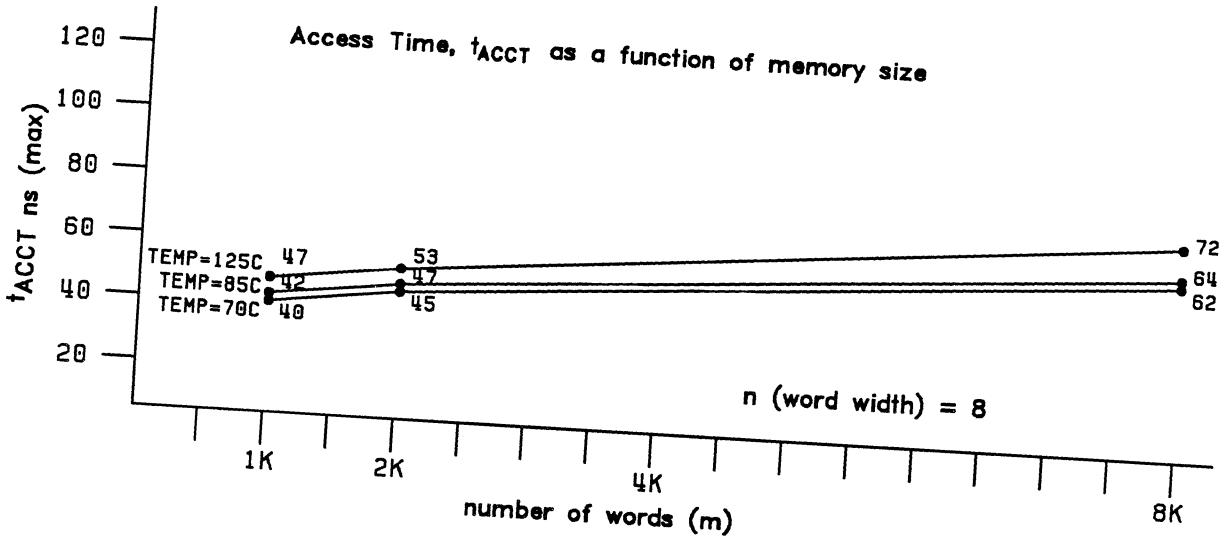


Figure 2 ROM Access time (DOT) output

SHFTGEN

ViGen CONFIGURABLE FUNCTION

VS1500 Shift Register

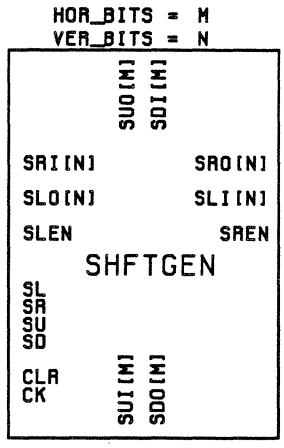
GENERAL DESCRIPTION

- m by n synchronous (positive edge triggered) shift register
- 8 to 32 horizontal bits, 1 to 32 vertical bits
- Rectangular array of static memory elements – a memory element may be conditionally enabled to load data from any of its four adjacent memory elements or from input terminals if the memory element is at the periphery of the array
- Data can always be shifted right. Shift left, up and down options are available
- A synchronous clear function initializes all elements to a zero state
- Right and left data output buses can optionally be configured as tristatable outputs for bus applications

All state changes occur on the rising edge of the clock. The memory elements are static and there is no minimum clock rate. When all control signals are low, the shift register is in the hold state and all memory elements retain their values independent of the clock. The memory element states are changed by the clock when one control signal is high. Multiple assertion of control signals is not allowed.

SYMBOL

The symbol for SHFTGEN will be unique for each configuration. An example is given here for reference only.

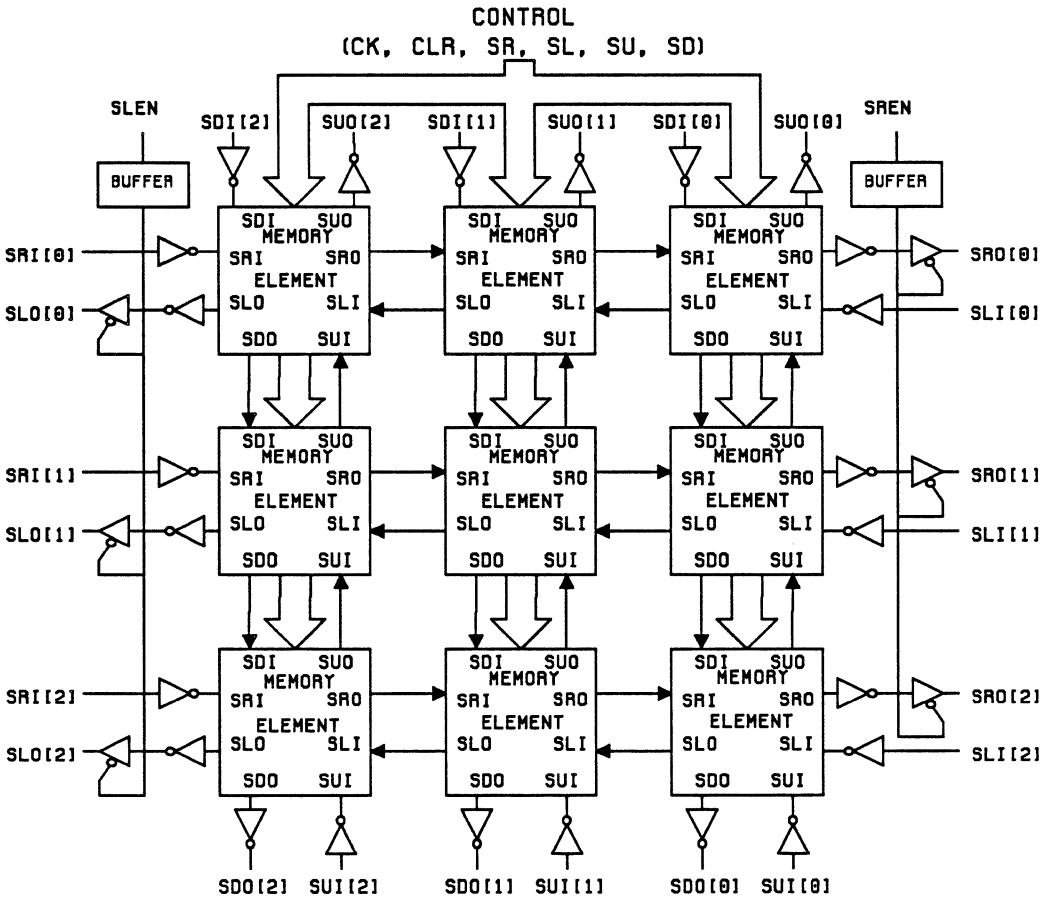


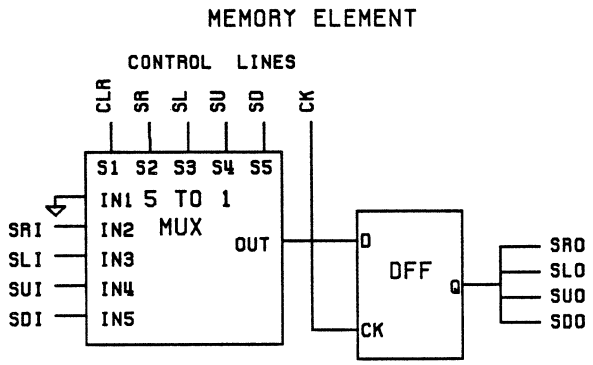
INPUT PARAMETER RANGES

INPUT PARAMETER	ALLOWED RANGE	EXPLANATION
HOR_BITS	8 – 32	Number of horizontal bits
VER_BITS	1 – 32	Number of vertical bits
SL_OPT	0 or 1	0: No shift left option 1: Shift left enabled
SU_OPT	0 or 1	0: No shift up option 1: Shift up enabled
SD_OPT	0 or 1	0: No shift down option 1: Shift down enabled
SREN_OPT	0 or 1	0: Always driving shift right outputs 1: Tristatable shift right outputs
SLEN_OPT	0 or 1	0: Always driving shift left outputs 1: Tristatable shift left outputs (NOTE: Only valid when SL=1)

FUNCTIONAL BLOCK DIAGRAM

The SHFTGEN array is organized with the shift right and shift left least significant bits on the top row. The shift up and shift down least significant bits are on the right most column of the array. If data is input on the shift up or down bus and then shifted right, the least significant bits will be shifted out first. If data is input on the shift right or left bus and then shifted up, the least significant bits will be shifted out first.





SHFTGEN

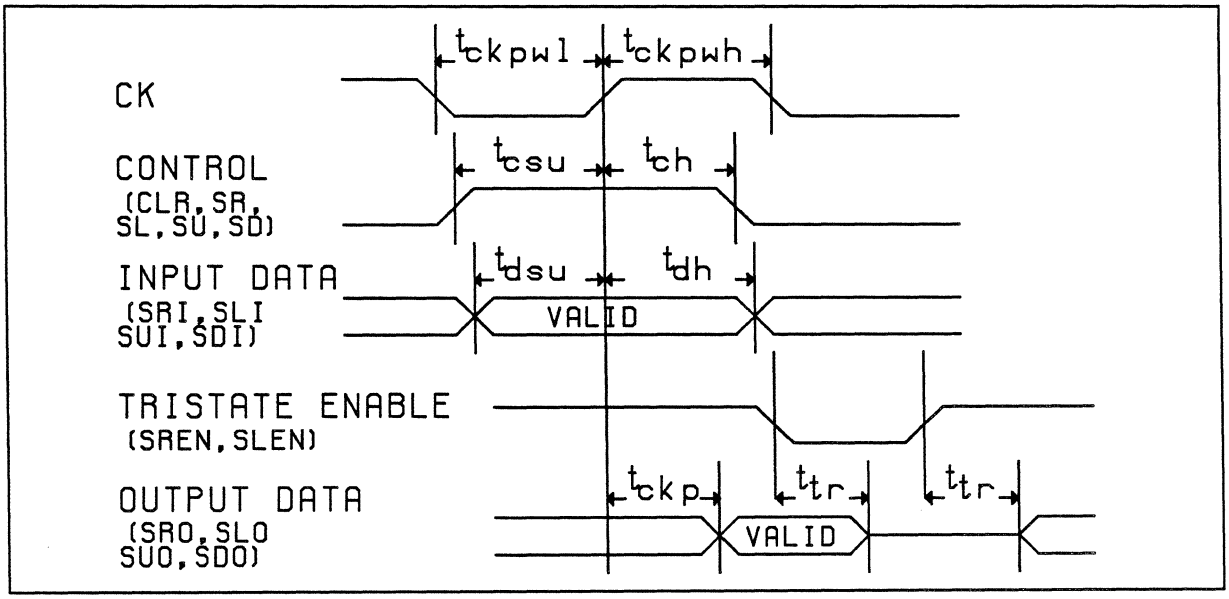
ViGen CONFIGURABLE FUNCTION

INPUTS/OUTPUTS

Definitions of the SHFTGEN inputs and outputs are given in the following table. The netlist order for all buses is least significant bit to most significant bit.

PIN NAME	FUNCTION	REQ/ OPT	CAP (pF)
INPUTS:			
CK	Clock – positive edge triggered	REQ	0.118
CLR	Clear – sets all memory elements to zero on rising clock edge (active high)	REQ	0.118
SR	Shift right control line (active high)	REQ	0.118
SL	Shift left control line (active high)	OPT	0.118
SU	Shift up control line (active high)	OPT	0.118
SD	Shift down control line (active high)	OPT	0.118
SREN	Right output tristate enable, tristates when low	OPT	0.060
SLEN	Left output tristate enable, tristates when low	OPT	0.060
SRI (N)	Shift right data input bus	REQ	0.056
SLI (N)	Shift left data input bus	OPT	0.056
SUI (M)	Shift up data input bus	OPT	0.061
SDI (M)	Shift down data input bus	OPT	0.061
OUTPUTS:			
SRO (N)	Shift right data output bus	REQ	0.181
SLO (N)	Shift left data output bus	OPT	0.181
SUO (M)	Shift up data output bus	OPT	
DSO (M)	Shift down data output bus	OPT	

AC WAVEFORMS



TIMING PARAMETERS AND CELL SIZE AS FUNCTIONS OF INPUT PARAMETERS

1. The input parameters are: HOR_BITS, VER_BITS, SL_OPT, SU_OPT, SD_OPT, SREN_OPT, and SLEN_OPT. A list of allowed ranges for the input parameters is given on page 1. CL is output capacitance in pF.
2. Timing parameters are specified for nominal process, Vdd=5.0 volts, and temperature = 25°C. See the NCR ASIC Data Book for information on derating factors for process, voltage and temperature. The t_{plh} notation refers to the output switching from low to high, and t_{phl} from high to low.

PARAM	DESCRIPTION	VALUE at NOMINAL CONDITIONS (ns)
t_{csu}	Control (CLR,SR,SL,SU,SD) setup time before rising clock edge	5.3
t_{ch}	Control (CLR,SR,SL,SU,SD) hold time after rising clock edge	1.3
t_{dsu}	Data (SRI,SLI,SUI,SDI) setup time before rising clock edge	2.8
t_{dh}	Data (SRI,SLI,SUI,SDI) hold time after rising clock edge	7.2
t_{ckpwl}	Min clock pulse width (low)	3.40 + 0.81*(SL_OPT + SU_OPT + SD_OPT) + 0.072*HOR_BITS + 0.082*VER_BITS
t_{ckpwh}	Min clock pulse width (high)	3.40 + 0.81*(SL_OPT + SU_OPT + SD_OPT) + 0.072*HOR_BITS + 0.082*VER_BITS
t_{ckp} (r,l)	Rising clock edge to valid right or left output data	t_{plh} = 6.30 + 0.81*(SL_OPT+SU_OPT+SD_OPT) + 0.051*HOR_BITS + 0.082*VER_BITS + 1.32*(SR,SL)EN_OPT + 1.00*CL t_{phl} = 5.23 + 0.62*(SL_OPT+SU_OPT+SD_OPT) + 0.048*HOR_BITS + 0.041*VER_BITS + 1.44*(SR,SL)EN_OPT + 1.08*CL

(NOTE: The SREN_OPT delay factor only adds to shift right delays and only when the tristatable shift right option is enabled. The SLEN_OPT delay factor adds only to shift left delays and only when the tristatable shift left option is enabled.)

t_{ckp} (u,d)	Rising clock edge to valid up or down output data	t_{plh} = 6.1 + 0.81*(SL_OPT + SU_OPT + SD_OPT) + 0.072*HOR_BITS + 0.082*VER_BITS + 1.02*CL t_{phl} = 4.1 + 0.62*(SL_OPT + SU_OPT + SD_OPT) + 0.072*HOR_BITS + 0.043*VER_BITS + 0.64*CL
t_{tr}	SREN, SLEN to tristate on or off delay	t_{plh} = 2.3 + 0.065*VER_BITS + 0.98*CL t_{phl} = 2.4 + 0.032*VER_BITS + 1.03*CL

Cell Width (mils) = 3.24 + (4.00 + 0.22*SL_OPT + 0.33*SU_OPT) * HOR_BITS + 1.00*SREN_OPT + 1.94*SL_OPT + 1.00*SLEN_OPT

Cell Height (mils) = 9.97 + 2.33*VER_BITS

TIMING EXAMPLES

HOR_BITS=32, VER_BITS=16, SL_OPT=0, SU_OPT=0, SD_OPT=0, SREN_OPT=0, SLEN_OPT=0

32 HOR_BIT, 16 VER_BIT SHIFTER		NOMINAL VDD=5V		WORST CASE VDD=4.5V						UNITS
SYMBOL	PARAMETER	TA=25C		TA=70C		TA=85C		TA=125C		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t_{csu}	Control (CLR,SR) setup time before rising clock edge	5.3		10.9		11.4		12.8		ns
t_{ch}	Control (CLR,SR) hold time after rising clock edge	1.3		2.7		2.8		3.1		ns
t_{dsu}	Data (SRI) setup time before rising clock edge	2.8		5.8		6.0		6.7		ns
t_{dh}	Data (SRI) hold time after rising clock edge	7.2		14.8		15.4		17.3		ns
t_{ckpr}	Rising clock edge to valid right output data t_{plh}		9.8		20.0		20.9		23.4	ns
	t_{phl}		8.0		16.4		17.1		19.1	
t_{ckpwl}	Minimum clock pulse (low)	7.0		14.4		15.0		16.9		ns
t_{ckpwh}	Minimum clock pulse (high)	7.0		14.4		15.0		16.9		ns

Switching Characteristics (Input tr,tf=1.4ns, CL=0.5pF)

HOR_BITS=16, VER_BITS=4, SL_OPT=0, SU_OPT=0, SD_OPT=1, SREN_OPT=1, SLEN_OPT=0

16 HOR_BIT, 4 VER_BIT SD_OPT, SREN_OPT SHIFTER		NOMINAL VDD=5V		WORST CASE VDD=4.5V						UNITS
SYMBOL	PARAMETER	TA=25C		TA=70C		TA=85C		TA=125C		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
t_{csu}	CLR,SR,SD setup time before rising clock edge	5.3		10.9		11.4		12.8		ns
t_{ch}	CLR,SR,SD hold time after rising clock edge	1.3		2.7		2.8		3.1		ns
t_{dsu}	Data (SRI,SDI) setup time before rising clock edge	2.8		5.8		6.0		6.7		ns
t_{dh}	Data (SRI,SDI) hold time after rising clock edge	7.2		14.8		15.4		17.3		ns
t_{ckpr}	Rising clock edge to valid right output data t_{plh}		10.1		20.7		21.6		24.2	ns
	t_{phl}		8.8		18.0		18.8		21.1	
t_{ckpd}	Rising clock edge to valid down output data t_{plh}		8.9		18.3		19.1		21.4	ns
	t_{phl}		6.4		13.1		13.6		15.3	
t_{tr}	SREN to tristate on or off delay t_{plh}		3.1		6.3		6.6		7.3	ns
	t_{phl}		3.1		6.3		6.5		7.3	
t_{ckpwl}	Minimum clock pulse (low)	5.7		11.7		12.2		13.7		ns
t_{ckpwh}	Minimum clock pulse (high)	5.7		11.7		12.2		13.7		ns

Switching Characteristics (Input tr,tf=1.4ns, CL=0.5pF)

APPLICATION NOTES

Area Calculations

Because of a layout approach that uses automatic place and route and compaction techniques, it is impossible to predict cell area exactly without actually creating the cell. Although the cell area equations are usually accurate to within 10%, in some instances the estimates can be off by as much as 20%.

Primary Uses

SHFTGEN is intended primarily for use in bus type applications where several bits are shifted in parallel with the same clock and control signals. Although the maximum number of horizontal bits is limited to 32, longer shift registers can be built by increasing the number of bits and using the output from bit 0 (SRO[0]) as the input for bit 1 (SRI[1]), etc. In fact, creating an N X M rectangular array and using the SRO[N] output bit as the input to the SRI[N+1] bit is the preferred way of implementing a 1 bit shift register that is (N X M) bits long. This creates smaller layouts than would be possible when only 1 bit is used. Using this wraparound strategy, a 32 bit shift register can be built in the three different configurations listed below. Note that by decreasing HOR_BITS and increasing VER_BITS, the area of the cell can be significantly reduced. Things to consider with this technique are:

1. The SRO[N] outputs must be connected to the SRI[N+1] inputs on the schematic.
2. This type of configuration cannot be used for parallel to serial or serial to parallel conversion because only the top and bottom horizontal bits can be accessed.
3. Maximum clock frequency is limited by ($t_{ckpr} + t_{dsu}$) or ($t_{ckpwl} + t_{ckpwh}$), whichever is greater.

The area to implement a shift register out of SRBN cells can be approximated as two times the area of the SRBN cell (35.4 $mils^2$/cell) plus the area of any buffering needed. Therefore, the area to implement a 32 bit shift register using SRBN cells would be at least 1134 $mils^2$. Using SHFTGEN with VER_BITS=1 and HOR_BITS=32 would actually create a layout that is bigger than the SRBN implementation. The other two SHFTGEN configurations will produce smaller layouts.

32 BIT SHFTGEN CONFIGURATIONS

VER_BITS	HOR_BITS	AREA
1	32	1615 $mils^2$
2	16	984 $mils^2$
4	8	680 $mils^2$

Parallel to Serial and Serial to Parallel Conversion

All SHFTGEN configurations must include the shift right option. If SHFTGEN is used to do parallel to serial or serial to parallel conversion, the shift right and shift down options should be used instead of the shift right and shift up options. Due to the layout methodology, the addition of the shift down option to the shift right option will result in no increase in cell size because the shift down transfer gate on the input multiplexer fits into an unused area of silicon in the shift right only shift register.

Naming Conventions

ViGen will automatically create a default cell name for each unique SHFTGEN configuration with the configuration information encoded in the following manner:

SG *hor_bits X ver_bits sl_opt su_opt sd_opt sren_opt slen_opt*

Therefore, the default names for the two example shifters would be:

SG32X1600000
SG16X400110

Limitations

QUICKSIM Save and Restore functions will not currently operate with generated cells.

VS1500 High Speed SRAM

GENERAL DESCRIPTION

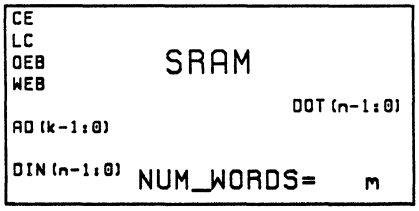
- Modular SRAM allows flexible organization
- Array sizes from 16 to 1024 words of 1 to 16 bits
- Access and Cycle times from 11ns to 22ns, nominal conditions
- Asynchronous operation – requires no clock
- Low power dissipation when unselected

Inputs: CE, LC, OEB, WEB, AD(k), DIN(n)
Outputs: DOT(n)

SYMBOL

The symbol for SRAM will be unique for each configuration. An example is shown here only for reference.

ADDRESS_WIDTH=k DATA_WIDTH=n



This SRAM supercell complements the clocked RAM in the standard cell library by offering higher speed at the expense of greater power dissipation. Independent input control lines provide cell enable, output data latching, and output tristate capability.

INPUT PARAMETER RANGES

INPUT PARAMETER	ALLOWED RANGE	EXPLANATION
Num_Words	16 – 1024 multiple of 16	The memory array is organized as "m" words of "n" bits each
Word_Width	1 – 16	

SRAM m x n

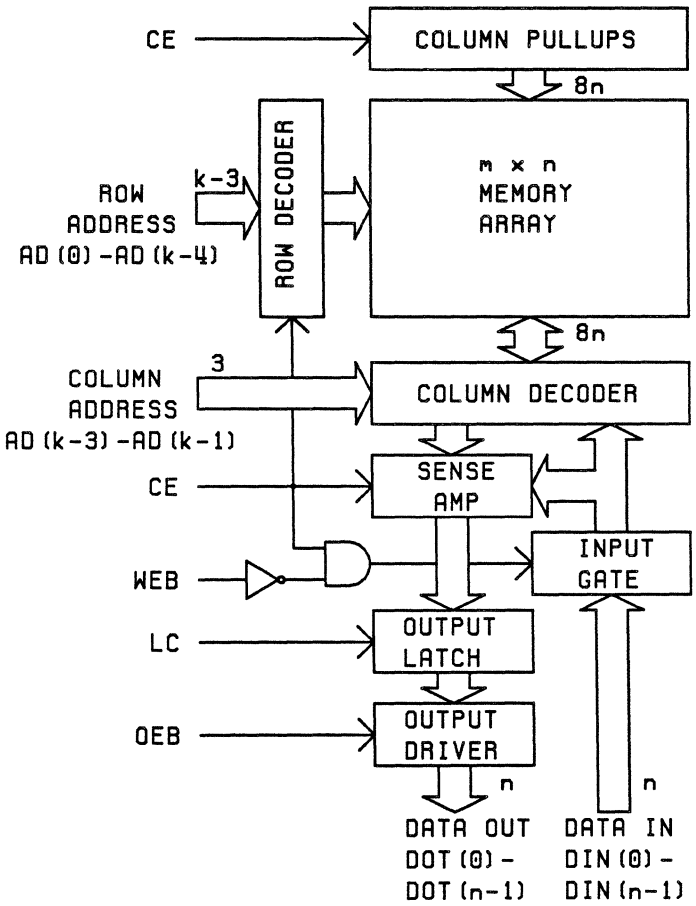
ViGen CONFIGURABLE FUNCTION

INPUTS/OUTPUTS

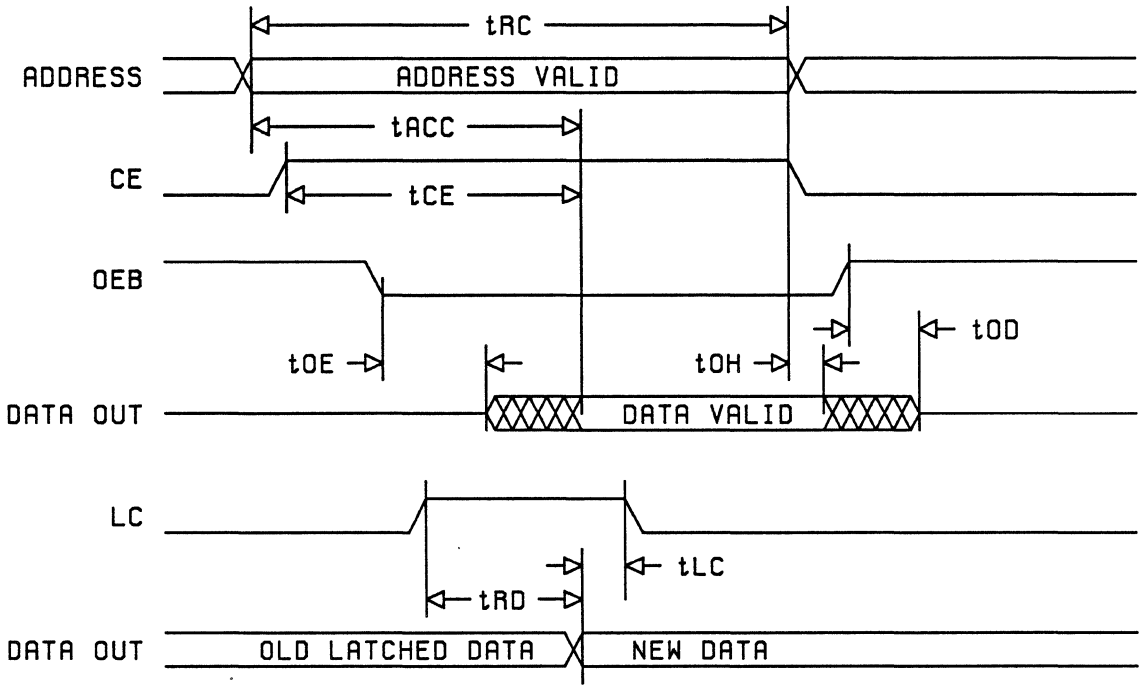
PIN NAME	FUNCTION	CAP (pF)
INPUTS:		
CE	Cell Enable	1.14
LC	Latch Control	1.44*
OEB	Output Enable Bar	0.34
WEB	Write Enable Bar	0.23
AD (k)	Address Bus	0.46
DIN (n)	Data Input Bus	0.24
OUTPUTS:		
DOT (n)	Data Output Bus	0.10

* The capacitance for LC varies with word width.
The value given is for a 16–bit wide word.
C = 0.32 + 0.07*n

FUNCTIONAL BLOCK DIAGRAM



READ CYCLE

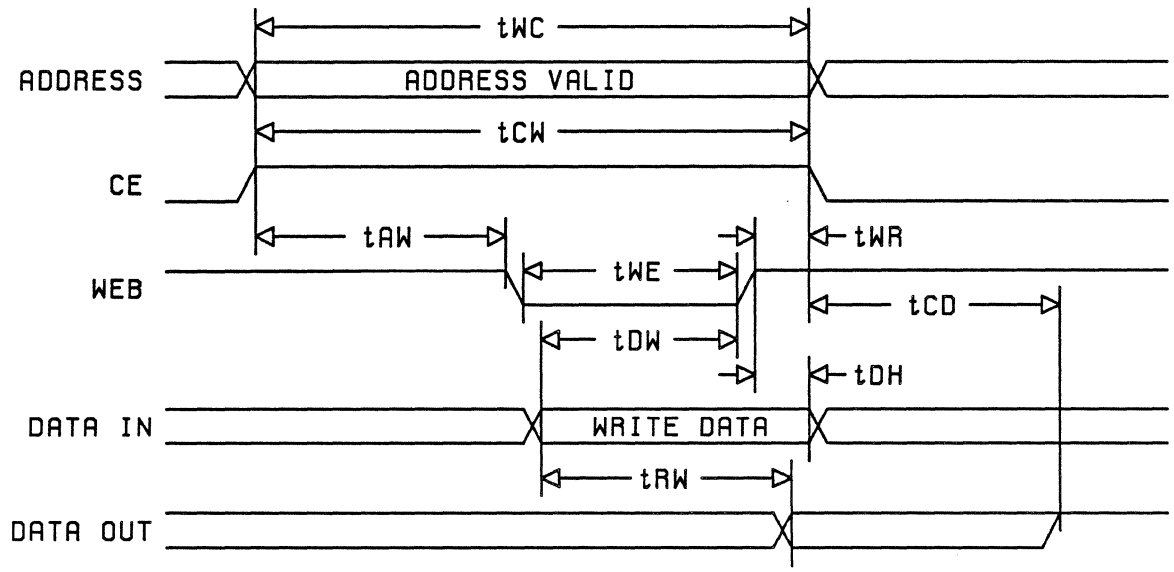


Read Cycle Timing for a 256 x 8 Array

Switching Characteristics (Input tr, tf=1.4ns, CL=0.5pF)

SYMBOL	PARAMETER	NOMINAL VDD=5V		WORST CASE VDD=4.5V						UNITS
		TA=25C		TA=70C		TA=85C		TA=125C		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
tRC	Read Cycle Time	14.0		28.7		29.9		33.6		ns
tACC	Address Access Time		14.0		28.7		29.9		33.6	ns
tCE	Cell Enable Time		14.0		28.7		29.9		33.6	ns
tOH	Data Hold Time	0		0		0		0		ns
tOE	Output Enable Time		5.9		12.2		12.7		14.3	ns
tOD	Output Disable Time		5.9		12.2		12.7		14.3	ns
tRD	LC Read Data Time	0	6.5	0	13.4	0	14.0	0	15.7	ns
tLC	Latch Time	0		0		0		0		ns

WRITE CYCLE



Write Cycle Timing for a 256 x 8 Array

Switching Characteristics (Input tr, tf=1.4ns, CL=0.5pF)

SYMBOL	PARAMETER	NOMINAL VDD=5V		WORST CASE VDD=4.5V						UNITS
		TA=25C		TA=70C		TA=85C		TA=125C		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
tWC	Write Cycle Time	10.4		21.2		22.1		24.8		ns
tCW	Cell Enable Time	10.4		21.2		22.1		24.8		ns
tAW	Address Write Margin	4.3		8.7		9.1		10.2		ns
tWE	Write Enable Time	4.4		9.0		9.4		10.5		ns
tWR	Write Recovery Time	1.7		3.5		3.7		4.1		ns
tDW	Data Write Time	4.1		8.4		8.8		9.8		ns
tDH	Data Hold Time	3.4		7.0		7.3		8.2		ns
tRW	Read after Write Time		8.0		16.5		17.2		19.3	ns
tCD	Data High from CE Low		17.3		35.6		37.1		41.6	ns

TIMING PARAMETERS AND CELL SIZE AS FUNCTIONS OF INPUT PARAMETERS

1. The input parameters are: NUM_WORDS and DATA_WIDTH. CL is the maximum capacitance in pF of the data output bus. ADDRESS_WIDTH is $\log_2$ (NUM_WORDS) rounded up to the nearest integer.
2. Timing parameters are specified for nominal process, Vdd = 5.0 volts, and temperature = 25°C. See the NCR ASIC Data Book for information on derating factors for process, voltage, and temperature.
3. The coefficient of CL is 0.47 for NDLH, and 0.41 for NDHL.

PARAM	DESCRIPTION	TYPICAL VALUE (ns)
t_{rc}	Read cycle time	10.0 + 0.22*DATA_WIDTH + 0.0078*NUM_WORDS + 0.47*CL
t_{acc}	Address access time	10.0 + 0.22*DATA_WIDTH + 0.0078*NUM_WORDS + 0.47*CL
t_{ce}	Cell enable time	10.0 + 0.22*DATA_WIDTH + 0.0078*NUM_WORDS + 0.47*CL
t_{oh}	Data hold time	0
t_{oe}	Output enable time	5.7 + 0.47*CL
t_{od}	Output disable time	5.7 + 0.47*CL
t_{rd}	LC read data time	6.3 + 0.47*CL
t_{lc}	Latch time	0
t_{wc}	Write cycle time	8.22 + 0.127*DATA_WIDTH + 0.0043*NUM_WORDS
t_{cw}	Cell enable time	8.22 + 0.127*DATA_WIDTH + 0.0043*NUM_WORDS
t_{aw}	Address write margin	2.14 + 0.127*DATA_WIDTH + 0.0043*NUM_WORDS
t_{we}	Write enable time	4.38
t_{wr}	Write recovery time	1.70
t_{dw}	Data write time	4.09
t_{dh}	Data hold time	3.41
t_{rw}	Read after write time	7.8 + 0.47*CL
t_{cd}	Data high from CE low	17.1 + 0.47*CL
t_r	Rise time	1.25 + 0.66*CL
t_f	Fall time	1.28 + 0.40*CL

Cell Width (mils)
for DATA_WIDTH ≤ 4:
= 1.6 + 7.3*DATA_WIDTH + 1.04*ADDRESS_WIDTH
for DATA_WIDTH > 4:
= 4.7 + 7.3*DATA_WIDTH + 1.04*ADDRESS_WIDTH

Cell Height (mils) = 14.0 + 0.146*NUM_WORDS

SRAM m x n

ViGen CONFIGURABLE FUNCTION

APPLICATION NOTES

General Description

The modular High Speed 1.5µ SRAM Supercell provides a means of adding high speed random access memory to a standard cell design. The asynchronous operation and familiar control interface makes this supercell similar to a stand alone static RAM device.

Read Mode

To read the High Speed SRAM, enable the cell (CE high) and assert the desired address. The SRAM is asynchronous and requires no precharge cycle between operations. The read data is gated to the output drivers when the LC signal is high and may be latched by bringing LC low. The output drivers are enabled when OEB is low and are in a high impedance state when OEB goes high. This allows easy interfacing and bidirectional I/O if desired.

The Latch Control signal (LC) operates independently form the Cell Enable signal (CE). If LC makes a transition from high to low while CE is low, all 1s will be latched into the output latches.

Write Mode

To write the SRAM, enable the cell (CE high), assert the desired address, and pulse the write enable bar signal (WEB) low. The data from the data inputs will be written to the memory array.

Power Dissipation

This SRAM supercell uses a truly static design approach commonly seen in stand alone devices. This approach allows the advantage of asynchronous operation with the corresponding faster cycle times at the expense of higher static power dissipation. The IDD current draw will depend on the number of bits per word, but will nominally be in the tens of milliamps range. The SRAM may be placed into a very low power mode (less than 100uA IDD) by bringing the CE signal low. This places the cell in a low power mode that is suitable for battery backed memory storage.

SRAM DC IDD CURRENT (Nominal Conditions)

For DATA_WIDTH ≤ 4

IDD READ = 2.7 + 3.3*DATA_WIDTH mA

IDD WRITE = 2.7 + 4.5*DATA_WIDTH mA

For DATA_WIDTH > 4

IDD READ = 5.4 + 3.3*DATA_WIDTH mA

IDD WRITE = 5.4 + 4.5*DATA_WIDTH mA

IDD WITH CE LOW < 100uA

Designing for Testability

Test programs for ASIC components must be able to verify that all circuit elements in the design are functioning properly. This includes verifying the functional correctness of the design and detecting faults caused by manufacturing defects. There are many methods to improve the testability of a component, all of which involve trade–offs in the amount of extra logic required, the resulting test time, and the degree of test coverage. A highly recommended method for the SRAM is to multiplex the address, control, and data lines to the external part pins during a test mode. This allows direct control and observability of the SRAM during test. In pin limited situations, the use of scan registers to shift in the address and data and shift out the output may be desired.

To adequately test the SRAM, the following conditions should be met as a minimum:

1. Every bit must be written to both a one and a zero at least once and verified.
2. The data output should change frequently between consecutive reads (a "checkerboard" pattern is a good example of this).
3. Use at least one pattern that will catch address decode faults. A good example of this is to write the address value to all locations in memory and then read back the results. Another example is to write the parity of the address to all locations and verify the results. The intent here is to catch faults in the address decode which may otherwise go undetected in a highly repetitive pattern. The "checkerboard" pattern is poor at detecting these types of faults.
4. Write the entire memory first and then read back the results. The intent here is to catch any write or read disturb problems that may otherwise go undetected. This is most effective on a highly unrepetitive pattern such as the one used for address decode fault checking.

Naming Conventions

ViGen will automatically create a default cell name for each unique SRAM configuration. The name of each configurations is encoded in the following manner:

SRAM *num_words X data_width*

Two examples of default names are:

SRAM256X16
SRAM32X8

Limitations

The Mentor BLM will store all bits of a word to unknown values (X's) if any bit of a word is unknown during the write cycle.

NOTICE

It is the policy of NCR Corporation to improve products as new technology, components, software, and firmware become available. NCR Corporation, therefore, reserves the right to change specifications without prior notice.

To locate your local NCR sales representative or to obtain more information about NCR products, call the NCR Information Hotline.

1-800-334-5454

February 1990

NCR Microelectronic Products Division

2001 Danfield Court

Fort Collins, Colorado 80525-2998

FAX: (303) 226-9556

Telex: 45-4505 NCR MICRO FTCN

Phone: (303) 226-9500, (303) 223-5100