Integrated Device Technology, Inc.

CMOS DUAL-PORT RAM 16K (2K x 8-BIT) WITH SEMAPHORE

IDT 71322S
IDT 71322L

FEATURES:

- High-speed access
  - Military: 45/55/70ns (max.)
  - Commercial: 35/45/55/70ns (max.)
- Low-power operation
  - IDT71322S
    Active: 500mW (typ.)
    Standby: 5mW (typ.)
  - IDT71322L
    Active: 500mW (typ.)
    Standby: 1mW (typ.)
- Fully asynchronous operation from either port
- Full on-chip hardware support of semaphore signalling between ports
- Battery backup operation—2V data retention
- TTL-compatible, single 5V(±10%) power supply
- Available in a variety of plastic and hermetic packages for both through hole and surface mount applications
- Military product compliant to MIL-STD-883, Class B

DESCRIPTION:

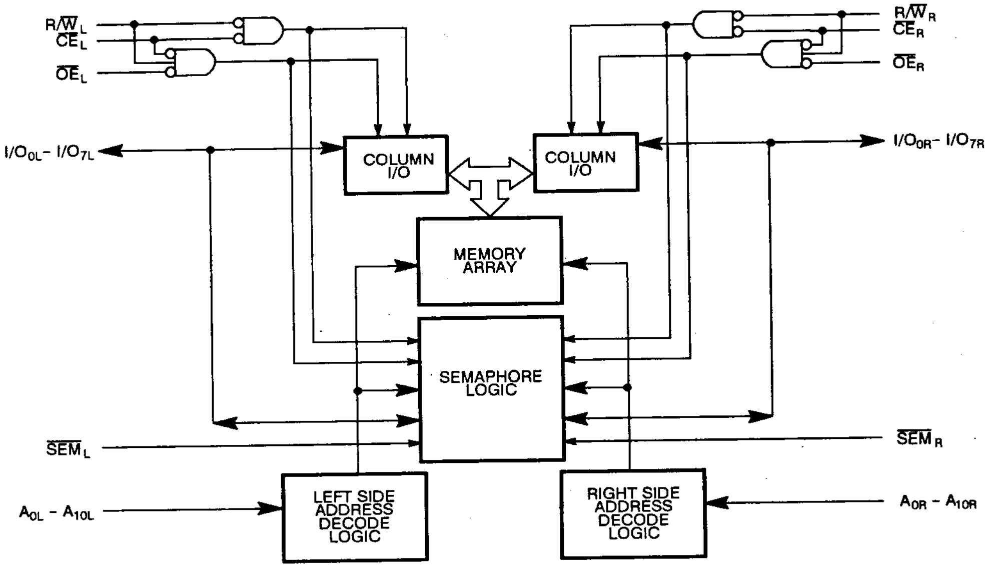
The IDT71322 is an extremely high-speed 2K x 8 dual-port static RAM with full on-chip hardware support of semaphore signalling between the two ports.

The IDT71322 provides two independent ports with separate control, address and I/O pins that permit independent, asynchronous access for reads and writes to any location in memory. To assist in arbitrating between ports, a fully independent semaphore logic block is provided. This block contains unassigned flags which can be accessed by either side; however, only one side can control the flag at any time. An automatic power down feature, controlled by $\overline{CE}$ and $\overline{SEM}$, permits the on-chip circuitry of each port to enter a very low standby power mode.

Fabricated using IDT's CEMOS ™ high-performance technology, this device typically operates on only 500mW of power at maximum access times as fast as 35ns. Low-power (L) versions offer battery backup data retention capability, with each port typically consuming 200µW from a 2V battery.

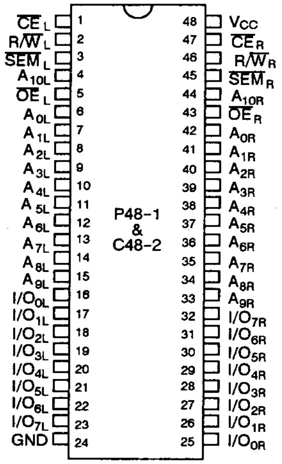
The IDT71322 is packaged in a 48-pin sidebraze or plastic DIP or 52-pin LCC and PLCC. Military grade product is manufactured in compliance with the latest revision of MIL-STD-883, Class B.

FUNCTIONAL BLOCK DIAGRAM

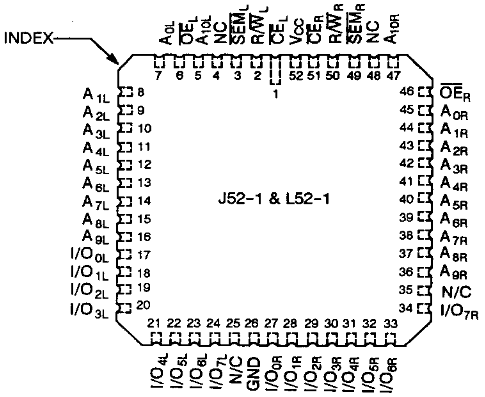


CEMOS is a trademark of Integrated Device Technology, Inc.

PIN CONFIGURATIONS



DIP TOP VIEW



LCC/PLCC TOP VIEW

CAPACITANCE (T_A = +25°C, f = 1.0MHz)

SYMBOL	PARAMETER[1]	CONDITIONS	MAX.	UNIT
C_{IN}	Input Capacitance	V_{IN} = 0V	11	pF
C_{OUT}	Output Capacitance	V_{OUT} = 0V	11	pF

NOTE:

1. This parameter is determined by device characteristics, but is not production tested.

ABSOLUTE MAXIMUM RATINGS [1]

SYMBOL	RATING	COMMERCIAL	MILITARY	UNIT
V_{TERM}	Terminal Voltage with Respect to GND	-0.5 to +7.0	-0.5 to +7.0	V
T_A	Operating Temperature	0 to +70	-55 to +125	°C
T_{BIAS}	Temperature Under Bias	-55 to +125	-65 to +135	°C
T_{STG}	Storage Temperature	-55 to +125	-65 to +150	°C
P_T	Power Dissipation	1.5	1.5	W
I_{OUT}	DC Output Current	50	50	mA

NOTE:

1. Stresses greater than those listed under ABSOLUTE MAXIMUM RATINGS may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

RECOMMENDED OPERATING TEMPERATURE AND SUPPLY VOLTAGE

GRADE	AMBIENT TEMPERATURE	GND	V_{CC}
Military	-55°C to +125°C	0V	5.0V ± 10%
Commercial	0°C to +70°C	0V	5.0V ± 10%

RECOMMENDED DC OPERATING CONDITIONS

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
V_{CC}	Supply Voltage	4.5	5.0	5.5	V
GND	Supply Voltage	0	0	0	V
V_{IH}	Input High Voltage	2.2	–	6.0	V
V_{IL}	Input Low Voltage	-0.5[1]	–	0.8	V

NOTE:

1. V_{IL} (min.) = -3.0V for pulse width less than 20ns.

DC ELECTRICAL CHARACTERISTICS OVER THE OPERATING TEMPERATURE AND SUPPLY VOLTAGE RANGE (V_{CC} = 5.0V ±10%)

SYMBOL	PARAMETER	TEST CONDITIONS	IDT71322S MIN.	IDT71322S MAX.	IDT71322L MIN.	IDT71322L MAX.	UNIT
$\|I_{LI}\|$	Input Leakage Current	V_{CC} = 5.5V, V_{IN} = 0V to V_{CC}	–	10	–	5	µA
$\|I_{LO}\|$	Output Leakage Current	$\overline{CE}$ = V_{IH}, V_{OUT} = 0V to V_{CC}	–	10	–	5	µA
V_{OL}	Output Low Voltage	I_{OL} = 6mA	–	0.4	–	0.4	V
		I_{OL} = 8mA	–	0.5	–	0.5	
V_{OH}	Output High Voltage	I_{OH} = -4mA	2.4	–	2.4	–	V

DC ELECTRICAL CHARACTERISTICS OVER THE OPERATING TEMPERATURE AND SUPPLY VOLTAGE RANGE[1] (V_{CC} = 5.0V ±10%)

SYMBOL	PARAMETER	TEST CONDITION	VERSION		IDT71322x35 COM'L ONLY TYP.[2]	MAX.	IDT71322x45 TYP.[2]	MAX.	IDT71322x55 TYP.[2]	MAX.	IDT71322x70 TYP.[2]	MAX.	UNIT
I_{CC}	Dynamic Operating Current (Both Ports Active)	$\overline{CE}$ = V_{IL} Outputs Open $\overline{SEM}$ = Don't Care f = f_{MAX}[3]	MIL.	S	–	–	100	240	100	230	100	230	mA
				L	–	–	100	200	100	180	100	180	
			COM'L.	S	–	220	100	200	100	200	100	200	
				L	–	180	100	160	100	160	100	160	
I_{CC1}	Dynamic Operating Current (Semaphores Both Sides)	$\overline{CE}$ = V_{IH} $\overline{SEM}$ = V_{IL} Outputs Open f = f_{MAX}[3]	MIL.	S	–	–	85	130	85	130	85	130	mA
				L	–	–	85	110	85	110	85	110	
			COM'L.	S	–	145	85	130	85	130	85	130	
				L	–	115	85	100	85	100	85	100	
I_{SB1}	Standby Current (Both Ports – TTL Level Inputs)	$\overline{CE}_L$ or $\overline{CE}_R$ ≥ V_{IH} $\overline{SEM}_R$ = $\overline{SEM}_L$ ≥ V_{IH} f = f_{MAX}[3]	MIL.	S	–	–	25	70	25	70	25	70	mA
				L	–	–	25	50	25	50	25	50	
			COM'L.	S	–	75	25	70	25	70	25	70	
				L	–	45	25	40	25	40	25	40	
I_{SB2}	Standby Current (One Port – TTL Level Inputs)	$\overline{CE}_L$ or $\overline{CE}_R$ ≥ V_{IH} Active Port Outputs Open, f = f_{MAX}[3] $\overline{SEM}_R$ = $\overline{SEM}_L$ = V_{IH}	MIL.	S	–	–	50	160	50	150	50	150	mA
				L	–	–	50	130	50	120	50	120	
			COM'L.	S	–	140	50	130	50	130	50	130	
				L	–	110	50	100	50	100	50	100	
I_{SB3}	Full Standby Current (Both Ports – All CMOS Level Inputs)	Both Ports $\overline{CE}_L$ and $\overline{CE}_R$ ≥ V_{CC} -0.2V V_{IN} ≥ V_{CC} -0.2V or V_{IN} ≤ 0.2V, $\overline{SEM}_R$ = $\overline{SEM}_L$ = V_{CC} -0.2V, f = 0[3]	MIL.	S	–	–	1	30	1	30	1	30	mA
				L	–	–	0.2	10	0.2	10	0.2	10	
			COM'L.	S	–	15	1	15	1	15	1	15	
				L	–	4	0.2	4	0.2	4	0.2	4	
I_{SB4}	Full Standby Current (One Port – All CMOS Level Inputs)	One Port $\overline{CE}_L$ or $\overline{CE}_R$ ≥ V_{CC} -0.2V V_{IN} ≥ V_{CC} -0.2V or V_{IN} ≤ 0.2V, Active Port Outputs Open, f = f_{MAX}[3]	MIL.	S	–	–	50	130	50	120	50	120	mA
				L	–	–	45	100	45	90	45	90	
			COM'L.	S	–	120	45	110	50	110	50	110	
				L	–	100	45	90	45	90	45	90	

NOTES:

1. x in part numbers indicates power rating (S or L).
2. V_{CC} = 5V, T_A = +25°C
3. f_{MAX} = 1/t_{RC} = All inputs cycling at f = 1/t_{RC} (except Output Enable). f = 0 means no address or control lines change. Applies only to inputs at CMOS level standby, I_{SB3}.

DATA RETENTION CHARACTERISTICS OVER ALL TEMPERATURE RANGES

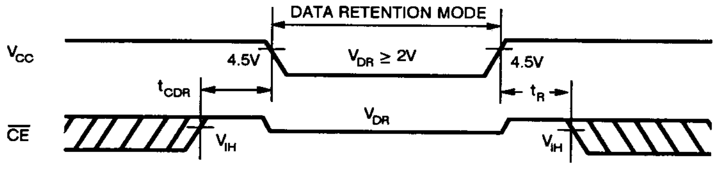
(L Version Only) V_{LC} = 0.2V, V_{HC} = V_{CC} - 0.2V

SYMBOL	PARAMETER	TEST CONDITION		MIN.	TYP.[1]	MAX.	UNIT
					2.0V	2.0V	
V_{DR}	V_{CC} for Data Retention	–		2.0	–	–	V
I_{CCDR}	Data Retention Current	V_{CC} = 2 V $\overline{CS} \geq V_{HC}$ $V_{IN} \geq V_{HC}$ or $\leq V_{LC}$	MIL.	–	100	4000	µA
			COM'L.	–	100	1500	
t_{CDR}[3]	Chip Deselect to Data Retention Time			0	–	–	ns
t_R[3]	Operation Recovery Time			t_{RC}[2]	–	–	ns

NOTES:

1. T_A = +25°C, V_{CC} = 2 V
2. t_{RC} = Read Cycle Time
3. This parameter is guaranteed but not tested.

LOW V_{CC} DATA RETENTION WAVEFORM



AC TEST CONDITIONS

Input Pulse Levels	GND to 3.0V
Input Rise/Fall Times	5ns
Input Timing Reference Levels	1.5V
Output Reference Levels	1.5V
Output Load	See Figures 1 & 2

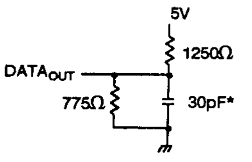


Figure 1. Output Load

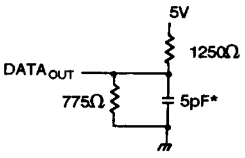


Figure 2. Output Load
(for t_{LZ}, t_{HZ}, t_{WZ}, t_{OW})

* Including scope and jig.

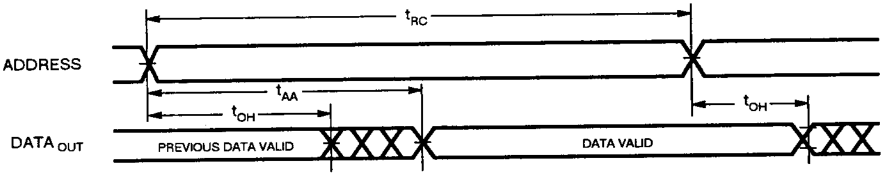
AC ELECTRICAL CHARACTERISTICS OVER THE OPERATING TEMPERATURE AND SUPPLY VOLTAGE RANGE

SYMBOL	PARAMETER	IDT71322S35 IDT71322L35 COM'L ONLY		IDT71322S45 IDT71322L45		IDT71322S55 IDT71322L55		IDT71322S70 IDT71322L70		UNIT
		MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
READ CYCLE										
t_{RC}	Read Cycle Time	35	–	45	–	55	–	70	–	ns
t_{AA}	Address Access Time	–	35	–	45	–	55	–	70	ns
t_{ACE}	Chip Enable Access Time (3)	–	35	–	45	–	55	–	70	ns
t_{AOE}	Output Enable Access Time	–	20	–	25	–	30	–	40	ns
t_{OH}	Output Hold From Address Change	5	–	5	–	5	–	5	–	ns
t_{LZ}	Output Low Z Time (1, 2)	5	–	5	–	5	–	5	–	ns
t_{HZ}	Output High Z Time (1, 2)	–	20	–	25	–	30	–	40	ns
t_{PU}	Chip Enable to Power Up Time (2)	0	–	0	–	0	–	0	–	ns
t_{PD}	Chip Disable to Power Down Time (2)	–	50	–	50	–	50	–	50	ns
t_{SOP}	Sem Flg update Pulse ($\overline{OE}$ or $\overline{SEM}$)	15	–	15	–	20	–	20	–	ns
t_{WDD}	Write Pulse to Data Delay (4)	–	70	–	80	–	90	–	60	ns
t_{DDD}	Write Data Valid to Read Data Delay (4)	–	35	–	45	–	55	–	70	ns

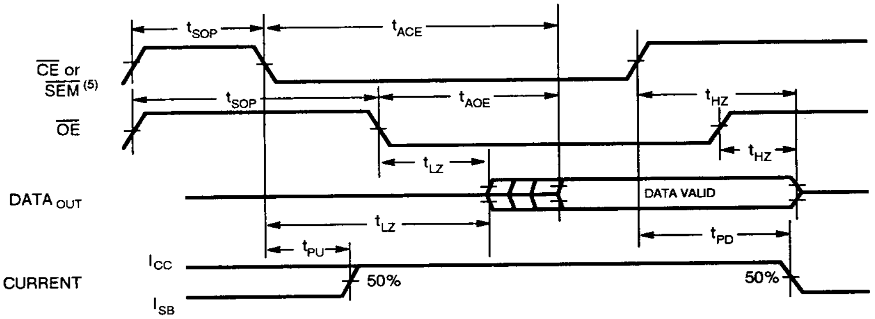
NOTES:

1. Transition is measured ±500mV from low or high impedance voltage with load (Figures 1 and 2).
2. This parameter is guaranteed but not tested.
3. To access RAM, $\overline{CE}$ = V_{IL}, $\overline{SEM}$ = V_{IH}. To access semaphore, $\overline{CE}$ = V_{IH}, $\overline{SEM}$ = V_{IL}.
4. Port to Port delay through RAM cells from writing port to Reading port.

TIMING WAVEFORM OF READ CYCLE NO. 1, EITHER SIDE (1, 2, 4)



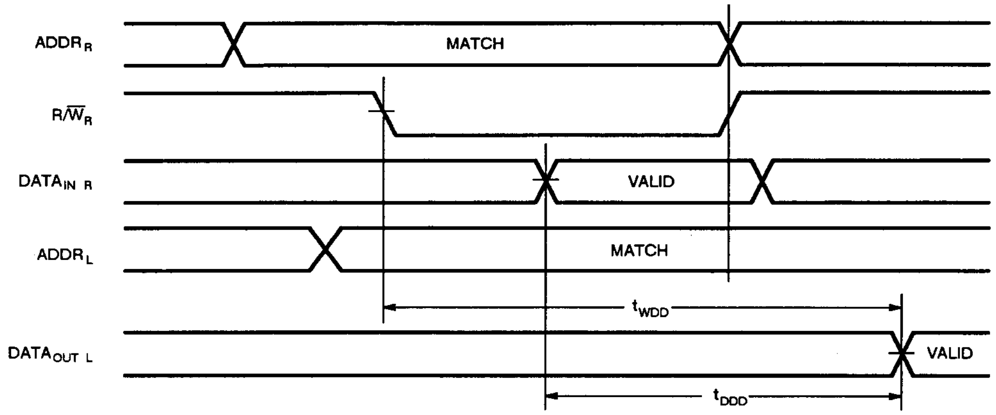
TIMING WAVEFORM OF READ CYCLE NO. 2, EITHER SIDE (1, 3)



NOTES:

1. R/$\overline{W}$ is high for Read Cycles.
2. Device is continuously enabled, $\overline{CE}$ = V_{IL}. This waveform cannot be used for semaphore reads.
3. Addresses valid prior to or coincident with $\overline{CE}$ transition low.
4. $\overline{OE}$ = V_{IL}
5. To access RAM, $\overline{CE}$ = V_{IL}, $\overline{SEM}$ = V_{IH}. To access semaphore, $\overline{CE}$ = V_{IH}, $\overline{SEM}$ = V_{IL}.

TIMING WAVEFORM OF READ WITH DELAY



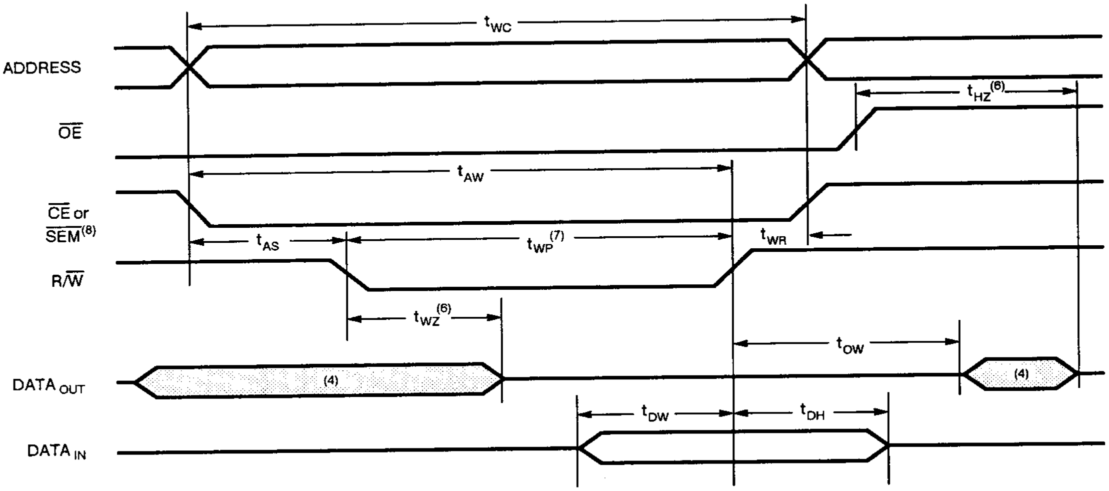
AC ELECTRICAL CHARACTERISTICS OVER THE OPERATING TEMPERATURE AND SUPPLY VOLTAGE RANGE

SYMBOL	PARAMETER	IDT71322S35 IDT71322L35 COM'L ONLY		IDT71322S45 IDT71322L45		IDT71322S55 IDT71322L55		IDT71322S70 IDT71322L70		UNIT
		MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
WRITE CYCLE										
t_{WC}	Write Cycle Time	35	–	45	–	55	–	70	–	ns
t_{EW}	Chip Enable to End of Write [3]	30	–	40	–	50	–	60	–	ns
t_{AW}	Address Valid to End of Write	30	–	40	–	50	–	60	–	ns
t_{AS}	Address Set-up Time	0	–	0	–	0	–	0	–	ns
t_{WP}	Write Pulse Width	30	–	40	–	50	–	60	–	ns
t_{WR}	Write Recovery Time	0	–	0	–	0	–	0	–	ns
t_{DW}	Data Valid to End of Write	20	–	20	–	25	–	30	–	ns
t_{HZ}	Output High Z Time [1,2]	–	20	–	20	–	25	–	30	ns
t_{DH}	Data Hold Time [4]	3	–	3	–	3	–	3	–	ns
t_{WZ}	Write Enabled to Output in High Z [1,2]	–	20	–	20	–	25	–	30	ns
t_{OW}	Output Active From End of Write [1,2,4]	3	–	3	–	3	–	3	–	ns
t_{SWRD}	$\overline{SEM}$ Flag Write to Read Time	10	–	10	–	10	–	10	–	ns
t_{SPS}	$\overline{SEM}$ Flag Contention Window	10	–	10	–	10	–	10	–	ns

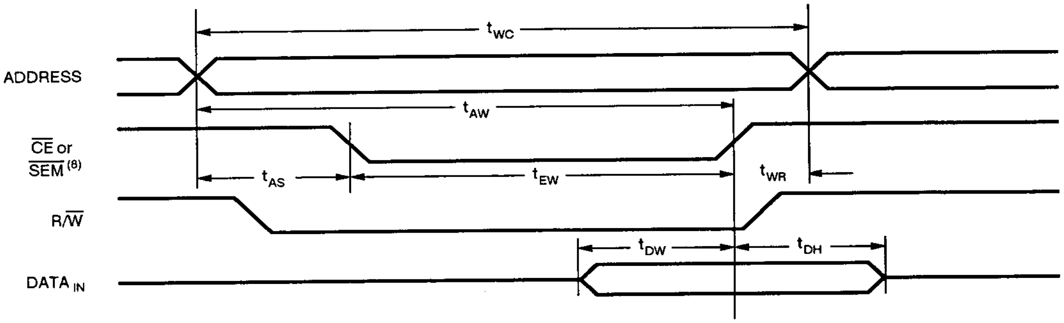
NOTES:

1. Transition is measured ±500mV from low or high impedance voltage with load (Figures 1 and 2).
2. This parameter is guaranteed but not tested.
3. To access RAM, $\overline{CE} = V_{IL}$, $\overline{SEM} = V_{IH}$. To access semaphore, $\overline{CE} = V_{IH}$, $\overline{SEM} = V_{IL}$. Either condition must be valid for the entire t_{EW} time.
4. The specification for t_{DH} must be met by the device supplying write data to the RAM under all operating conditions. Although t_{DH} and t_{OW} values will vary over voltage and temperature, the actual t_{DH} will always be smaller than the actual t_{OW}.

TIMING WAVEFORM OF WRITE CYCLE NO. 1, $R/\overline{W}$ CONTROLLED TIMING (1, 2, 3, 4, 6, 7, 8)



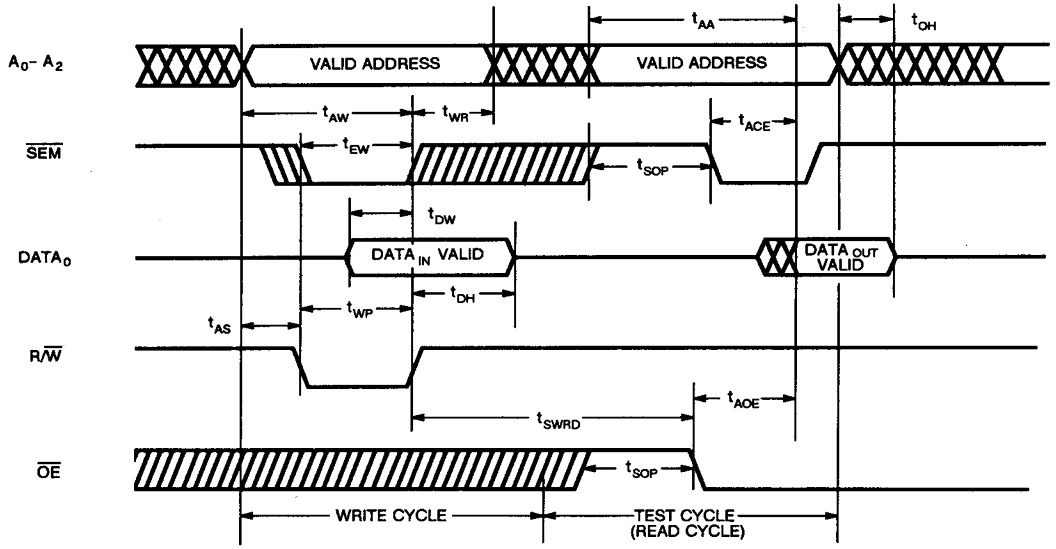
TIMING WAVEFORM OF WRITE CYCLE NO. 2, $\overline{CE}$ CONTROLLED TIMING (1, 2, 3, 5, 8)



NOTES:

1. $R/\overline{W}$ must be high during all address transitions.
2. A write occurs during the overlap (t_{EW} or t_{WP}) of a low $\overline{CE}$ or $\overline{SEM}$ and a low $R/\overline{W}$.
3. t_{WR} is measured from the earlier of $\overline{CE}$ or $R/\overline{W}$ (or $\overline{SEM}$ or $R/\overline{W}$) going high to the end of write cycle.
4. During this period, the I/O pins are in the output state, and input signals must not be applied.
5. If the $\overline{CE}$ or $\overline{SEM}$ low transition occurs simultaneously with or after the $R/\overline{W}$ low transition, the outputs remain in the high impedance state.
6. Transition is measured ±500mV from steady state with a 5pF load (including scope and jig).
7. If $\overline{OE}$ is low during a $R/\overline{W}$ controlled write cycle, the write pulse width must be the larger of t_{WP} or (t_{WZ} + t_{DW}) to allow the I/O drivers to turn off and data to be placed on the bus for the required t_{DW}. If $\overline{OE}$ is high during an $R/\overline{W}$ controlled write cycle, this requirement does not apply and the write pulse can be as short as the specified t_{WP}.
8. To access RAM, $\overline{CE}$ = V_{IL}, $\overline{SEM}$ = V_{IH}. To access semaphore, $\overline{CE}$ = V_{IH}, $\overline{SEM}$ = V_{IL}. Either condition must be valid for the entire t_{EW} time.

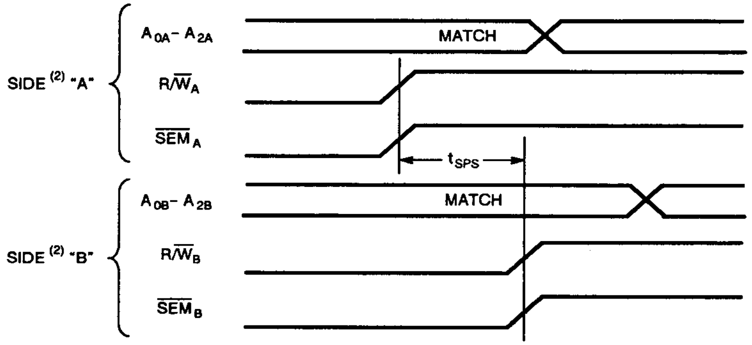
TIMING WAVEFORM OF SEMAPHORE READ AFTER WRITE TIMING, EITHER SIDE [1]



NOTE:

1. $\overline{CE} = V_{IH}$ for the duration of the above timing (both write and read cycle).

TIMING WAVEFORM OF SEMAPHORE CONTENTION [1, 3, 4]



NOTES:

1. $D_{0R} = D_{0L} = V_{IL}$, $\overline{CE}_R = \overline{CE}_L = V_{IH}$. Semaphore Flag is released from both sides (reads as ones from both sides) at cycle start.
2. Either side "A" = left and side "B" = right, or side "A" = right and side "B" = left.
3. This parameter is measured from the point where $R/\overline{W}_A$ or $\overline{SEM}_A$ goes high until $R/\overline{W}_B$ or $\overline{SEM}_B$ goes high.
4. If t_{SPS} is violated, the semaphore will fall positively to one side or the other, but there is no guarantee which side will obtain the flag.

FUNCTIONAL DESCRIPTION

The IDT71322 is an extremely fast dual-port 2K x 8 CMOS static RAM with an additional 8 address locations dedicated to binary semaphore flags. These flags allow either processor on the left or right side of the dual-port RAM to claim a privilege over the other processor for functions defined by the system designer's software. As an example, the semaphore can be used by one processor to inhibit the other from accessing a portion of the dual-port RAM or any other shared resource.

The dual-port RAM features a fast access time, and both ports are completely independent of each other. This means that the activity on the left port in no way slows the access time of the right port. Both ports are identical in function to standard CMOS static RAM and can be read from, or written to, at the same time with the only possible conflict arising from the simultaneous writing of, or a simultaneous READ/$\overline{WRITE}$ of, a non-semaphore location. Semaphores are protected against such ambiguous situations and may be used by the system program to avoid any conflicts in the non-semaphore portion of the dual-port RAM. These devices have an automatic power-down feature controlled by $\overline{CE}$, the dual-port RAM enable, and $\overline{SEM}$, the semaphore enable. The $\overline{CE}$ and $\overline{SEM}$ pins control on-chip power down circuitry that permits the respective port to go into standby mode when not selected. This is the condition which is shown in Table I where $\overline{CE}$ and $\overline{SEM}$ are both high.

Systems which can best use the IDT71322 contain multiple processors or controllers and are typically very high-speed systems which are software controlled or software intensive. These systems can benefit from a performance increase offered by the IDT71322's hardware semaphores, which provide a lockout mechanism without requiring complex programming.

Software handshaking between processors offers the maximum in system flexibility by permitting shared resources to be allocated in varying configurations. The IDT71322 does not use its semaphore flags to control any resources through hardware, thus allowing the system designer total flexibility in system architecture.

An advantage of using semaphores rather than the more common methods of hardware arbitration is that wait states are never incurred in either processor. This can prove to be a major advantage in very high-speed systems.

HOW THE SEMAPHORE FLAGS WORK

The semaphore logic is a set of eight latches which are independent of the dual-port RAM. These latches can be used to pass a flag, or token, from one port to the other to indicate that a shared resource is in use. The semaphores provide a hardware assist for a use assignment method called "Token Passing Allocation." In this method, the state of a semaphore latch is used as a token indicating that a shared resource is in use. If the left processor wants to use this resource, it requests the token by setting the latch. This processor then verifies its success in setting the latch by reading it. If it was successful, it proceeds to assume control over the shared resource. If it was not successful in setting the latch, it determines that the right side processor had set the latch first, has the token and is using the shared resource. The left processor can then either repeatedly request that semaphore's status or remove its request for that semaphore to perform another task and occasionally attempt again to gain control of the token via the set and test sequence. Once the right side has relinquished the token, the left side should succeed in gaining control.

The semaphore flags are active low. A token is requested by writing a zero into a semaphore latch and is released when the same side writes a one to that latch.

The eight semaphore flags reside within the IDT71322 in a separate memory space from the dual-port RAM. This address space is accessed by placing a low input on the $\overline{SEM}$ pin (which acts as a chip select for the semaphore flags) and using the other control pins (Address, $\overline{OE}$, and R/$\overline{W}$) as they would be used in accessing a standard static RAM. Each of the flags has a unique address which can be accessed by either side through address pins A_0 - A_2. When accessing the semaphores, none of the other address pins has any effect.

When writing to a semaphore, only data pin D_0 is used. If a low level is written into an unused semaphore location, that flag will be set to a zero on that side and a one on the other (see Table II). That semaphore can now only be modified by the side showing the zero. When a one is written into the same location from the same side, the flag will be set to a one for both sides (unless a semaphore request from the other side is pending) and then can be written to by both sides. The fact that the side which is able to write a zero into a semaphore subsequently locks out writes from the other side is what makes semaphore flags useful in interprocessor communications. (A thorough discussion on the use of this feature follows shortly.) A zero written into the same location from the other side will be stored in the semaphore request latch for that side until the semaphore is freed by the first side.

When a semaphore flag is read, its value is spread into all data bits so that a flag that is a one reads as a one in all data bits and a flag containing a zero reads as all zeros. The read value is latched into one side's output register when that side's semaphore select ($\overline{SEM}$) and output enable ($\overline{OE}$) signals go active. This serves to disallow the semaphore from changing state in the middle of a read cycle due to a write cycle from the other side. Because of this latch, a repeated read of a semaphore in a test loop must cause either signal ($\overline{SEM}$ or $\overline{OE}$) to go inactive or the output will never change.

A sequence of WRITE/READ must be used by the semaphore in order to guarantee that no system level contention will occur. A processor requests access to shared resources by attempting to write a zero into a semaphore location. If the semaphore is already in use, the semaphore request latch will contain a zero, yet the semaphore flag will appear as a one, a fact which the processor will verify by the subsequent read (see Table II). As an example, assume a processor writes a zero to the left port at a free semaphore location. On a subsequent read, the processor will verify that it has written successfully to that location and will assume control over the resource in question. Meanwhile, if a processor on the right side attempts to write a zero to the same semaphore flag it will fail, as will be verified by the fact that a one will be read from that semaphore on the right side during a subsequent read. Had a sequence of READ/WRITE been used instead, system contention problems could have occurred during the gap between the read and write cycles.

It is important to note that a failed semaphore request must be followed by either repeated reads or by writing a one into the same location. The reason for this is easily understood by looking at the simple logic diagram of the semaphore flag in Figure 3. Two semaphore request latches feed into a semaphore flag. Whichever latch is first to present a zero to the semaphore flag will force its side of the semaphore flag low and the other side high. This condition will continue until a one is written to the same semaphore request latch. Should the other side's semaphore request latch have been written to a zero in the meantime, the semaphore flag will flip over to the other side as soon as a one is written into the first side's request latch. The second side's flag will now stay low until its semaphore request latch is written to a one. From this it is easy to understand that, if a semaphore is requested and the processor which requested it no longer needs the resource, the entire system can hang up until a one is written into that semaphore request latch.

TABLE I – NON-CONTENTION READ/WRITE CONTROL

LEFT OR RIGHT PORT [(1)]					FUNCTION
$R/\overline{W}$	$\overline{CE}$	$\overline{SEM}$	$\overline{OE}$	D_{0-7}	
X	H	H	X	Z	Port Disabled and in Power Down Mode
H	H	L	L	$DATA_{OUT}$	Data in Semaphore Flag Output on Port
X	X	X	H	Z	Output Disabled
↑	H	L	X	$DATA_{IN}$	Port Data Bit D_0 Written Into Semaphore Flag
H	L	H	L	$DATA_{OUT}$	Data In Memory Output on Port
L	L	H	X	$DATA_{IN}$	Data On Port Written Into Memory
X	L	L	X	–	Not Allowed

NOTE:

1. $A_{0L} - A_{10L} \neq A_{0R} - A_{10R}$
   H = HIGH, L = LOW, X = DON'T CARE, Z = HIGH IMPEDANCE
   ↑ = Low-to-High transition

TABLE II – EXAMPLE OF SEMAPHORE PROCUREMENT SEQUENCE [(1)]

FUNCTION	$D_0 - D_7$ LEFT	$D_0 - D_7$ RIGHT	STATUS
No Action	1	1	Semaphore free
Left Port Writes "0" to Semaphore	0	1	Left port has semaphore token
Right Port Writes "0" to Semaphore	0	1	No change. Right side has no write access to semaphore
Left Port Writes "1" to Semaphore	1	0	Right port obtains semaphore token
Left Port Writes "0" to Semaphore	1	0	No change. Left port has no write access to semaphore
Right Port Writes "1" to Semaphore	0	1	Left port obtains semaphore token
Left Port Writes "1" to Semaphore	1	1	Semaphore free
Right Port Writes "0" to Semaphore	1	0	Right port has semaphore token
Right Port Writes "1" to Semaphore	1	1	Semaphore free
Left Port Writes "0" to Semaphore	0	1	Left port has semaphore token
Left Port Writes "1" to Semaphore	1	1	Semaphore free

NOTE:

1. This table denotes a sequence of events for only one of the eight semaphores on the IDT71322.

The critical case of semaphore timing is when both sides request a single token by attempting to write a zero into it at the same time. The semaphore logic is specially designed to resolve this problem. If simultaneous requests are made, the logic guarantees that only one side receives the token. If one side is earlier than the other in making the request, the first side to make the request will receive the token. If both requests arrive at the same time, the assignment will be arbitrarily made to one port or the other.

One caution that should be noted when using semaphores is that semaphores alone do not guarantee that access to a resource is secure. As with any powerful programming technique, if semaphores are misused or misinterpreted, a software error can easily happen. Code integrity is of the utmost importance when semaphores are used instead of slower, more restrictive hardware intensive schemes.

Initialization of the semaphores is not automatic and must be handled via the initialization program at power-up. Since any semaphore request flag which contains a zero must be reset to a one, all semaphores on both sides should have a one written into them at initialization from both sides to assure that they will be free when needed.

USING SEMAPHORES—Some Examples

Perhaps the simplest application of semaphores is their application as resource markers for the IDT71322's dual-port RAM. Say the 2K x 8 RAM was to be divided into two 1K x 8 blocks which were to be dedicated at any one time to servicing either the left or right port. Semaphore 0 could be used to indicate the side which would control the lower section of memory, and Semaphore 1 could be defined as the indicator for the upper section of memory.

To take a resource, in this example the lower 1K of dual-port RAM, the processor on the left port could write and then read a zero into Semaphore 0. If this task were successfully completed (a zero was read back rather than a one), the left processor would assume control of the lower 1K. Meanwhile, the right processor would attempt to perform the same function. Since this processor was attempting to gain control of the resource after the left processor, it would read back a one in response to the zero it had attempted to write into Semaphore 0. At this point, the software could choose to try and gain control of the second 1K section by writing, then reading a zero into Semaphore 1. If it succeeded in gaining control, it would lock out the left side.

Once the left side was finished with its task, it would write a one to Semaphore 0 and may then try to gain access to Semaphore 1. If Semaphore 1 was still occupied by the right side, the left side could undo its semaphore request and perform other tasks until it was able to write, then read a zero into Semaphore 1. If the right processor performs a similar task with Semaphore 0, this protocol would allow the two processors to swap 1K blocks of dual-port RAM with each other.

The blocks do not have to be any particular size and can even be variable, depending upon the complexity of the software using the semaphore flags. All eight semaphores could be used to divide the dual-port RAM or other shared resources into eight parts. Semaphores can even be assigned different meanings on different sides rather than being given a common meaning as was shown in the example above.

Semaphores are a useful form of arbitration in systems like disk interfaces where the CPU must be locked out of a section of memory during a transfer and the I/O device cannot tolerate any wait states. With the use of semaphores, once the two devices had determined which memory area was "off limits" to the CPU, both the CPU and the I/O devices could access their assigned portions of memory continuously without any wait states.

Semaphores are also useful in applications where no memory "WAIT" state is available on one or both sides. Once a semaphore handshake has been performed, both processors can access their assigned RAM segments at full speed.

Another application is in the area of complex data structures. In this case, block arbitration is very important. For this application one processor may be responsible for building and updating a data structure. The other processor then reads and interprets that data structure. If the interpreting processor reads an incomplete data structure, a major error condition may exist. Therefore, some sort of arbitration must be used between the two different processors. The building processor arbitrates for the block, locks it and then is able to go in and update the data structure. When the update is completed, the data structure block is released. This allows the interpreting processor to come back and read the complete data structure, thereby guaranteeing a consistent data structure.

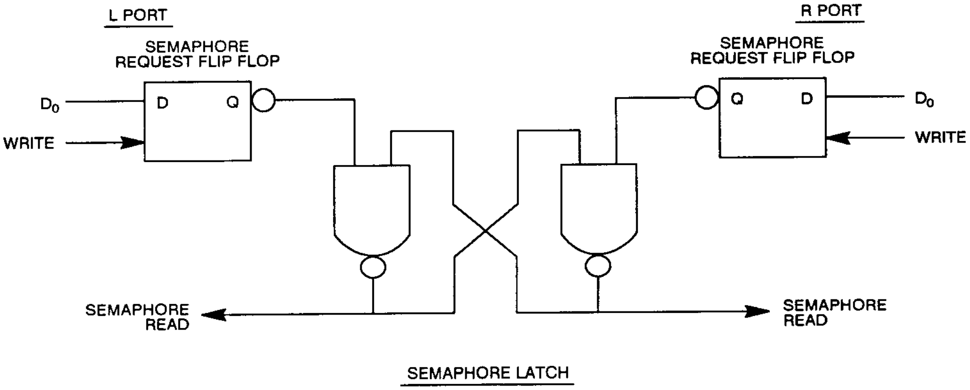
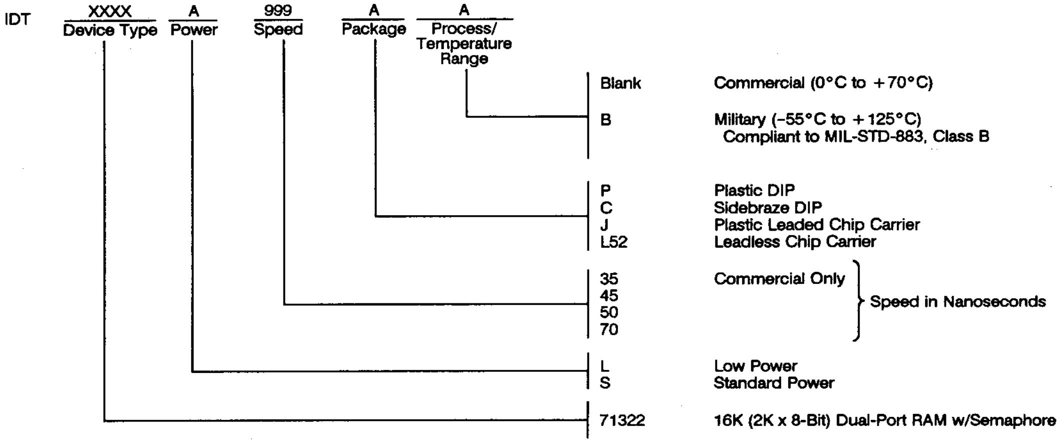


FIGURE 3. IDT71322 Semaphore Logic

ORDERING INFORMATION



IDT XXXX Device Type, A Power, 999 Speed, A Package, A Process/Temperature Range

Blank	Commercial (0°C to +70°C)
B	Military (−55°C to +125°C) Compliant to MIL-STD-883, Class B
P	Plastic DIP
C	Sidebraze DIP
J	Plastic Leaded Chip Carrier
L52	Leadless Chip Carrier
35	Commercial Only
45	
50	
70	
L	Low Power
S	Standard Power
71322	16K (2K x 8-Bit) Dual-Port RAM w/Semaphore

35, 45, 50, 70: Speed in Nanoseconds