Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

Features

- **Eight General-purpose Floating-point Data Registers, Each Supporting a Full 80-bit Extended Precision Real Data Format (a 64-bit Mantissa Plus a Sign Bit, and a 15-bit Signed Exponent)**
- **A 67-bit Arithmetic Unit to Allow Very Fast Calculations with Intermediate are Precision Greater than the Extended Precision Format**
- **A 67-bit Barrel Shifter for High-speed Shifting Operations (for Normalizing etc.)**
- **Special-purpose Hardware for High-speed Conversion Between Single, Double, and Extended Formats and the Internal Extended Format**
- **An Independent State Machine to Control Main Processor Communication for Pipelined Instruction Processing**
- **Forty-six Instructions, Including 35 Arithmetic Operations**
- **Full Conformation to the IEEE 754 Standard, Including All Requirements and Suggestions**
- **Support of Functions Not Defined by the IEEE Standard, Including a Full Set of Trigonometric and Transcendental Functions**
- **Seven Data Type Types: Byte, Word and Long Integers; Single, Double, and Extended Precision Real Numbers; and Packed Binary Coded Decimal String Real Numbers**
- **Twenty-two Constants Available In The On-chip ROM, Including π, e, and Powers of 10**
- **Virtual Memory/Machine Operations**
- **Efficient Mechanisms for Procedure Calls, Context Switches, and Interrupt Handling**
- **Fully Concurrent Instruction Execution with the Main Processor**
- **Fully Concurrent Instruction Execution of Multiple Floating-point Instructions**
- **Use with any Host Processor, on an 8-, 16- or 32-bit Data Bus**
- **Available in 16.67, 20, 25 and 33 MHz for T_c from -55°C to +125°C**
- **V_{CC} = 5V ± 10%**

Description

The TS68882 enhanced floating-point co-processor is a full implementation of the IEEE Standard for Binary Floating-Point Arithmetic (754) for use with the THOMSON TS68000 Family of microprocessors. It is a pin and software compatible upgrade of the TS68881 with optimized MPU interface that provides over 1.5 times the performance of the TS68881. It is implemented using VLSI technology to give systems designers the highest possible functionality in a physically small device.

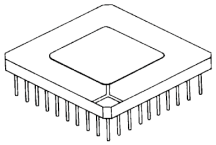
Intended primarily for use as a co-processor to the TS68020/68030 32-bit microprocessor units (MPUs), the TS68882 provides a logical extension to the main MPU integer data processing capabilities. It does this by providing a very high performance floating-point arithmetic unit and a set of floating-point data registers that are utilized in a manner that is analogous to the use of the integer data registers. The TS68882 instruction set is a natural extension of all earlier members of the TS68000 Family, and supports all of the addressing modes of the host MPU. Due to the flexible bus interface of the TS68000 Family, the TS68882 can be used with any of the MPU devices of the TS68000 Family, and it may also be used as a peripheral to non-TS68000 processors.

Screening/Quality

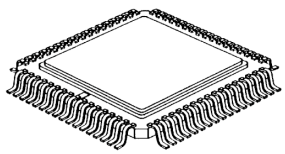
This product could be manufactured in full compliance with either:

- MIL-STD-883 Class B
- DESC 5962-89436
- or According to ATMEL-Grenoble Standards

R suffix
PGA 68
Ceramic Pin Grid Array



F suffix
CQFP 68
Ceramic Quad Flat Pack



CMOS Enhanced Floating-point Co-processor

TS68882

Rev. 2119A–HIREL–04/02

Introduction

The TS68882 is a high-performance floating-point device designed to interface with the TS68020 or TS68030 as a co-processor. This device fully supports the TS68000 virtual machine architecture, and is implemented in HCMOS, Atmel's low power, small geometry process. This process allows CMOS and HMOS (high-density NMOS) gates to be combined on the same device. CMOS structures are used where speed and low power is required, and HMOS structures are used where minimum silicon area is desired. The HCMOS technology enables the TS68882 to be very fast while consuming less power than comparable HMOS, and still have a reasonably small die size.

With some performance degradation, the TS68882 can also be used as a peripheral processor in systems where the TS68020 or TS68030 is not the main processor (i.e., TS68000, TS68010). The configuration of the TS68882 as a peripheral processor or co-processor may be completely transparent to user software (i.e., the same object code may be executed in either configuration).

The architecture of the TS68882 appears to the user as a logical extension of the TS68000 Family architecture. Coupling of the co-processor interface allows the TS68020/TS68030 programmer to view the TS68882 registers as though the registers are resident in the TS68020/TS68030. Thus, a TS68020 or TS68030/TS68882 device pair appears to be one processor that supports seven floating-point and integer data types, and has eight integer data registers, eight address registers, and eight floating-point data registers.

As shown in Figure 1, the TS68882 is internally divided into four processing elements; the Bus Interface Unit (BIU), the Conversion Control Unit (CCU), the Execution Control Unit (ECU), and the Microcode Control Unit (MCU). The BIU communicates with the main processor, the CCU controls the main processor communications dialog and performs some data conversions, and the ECU and MCU execute most floating-point calculations.

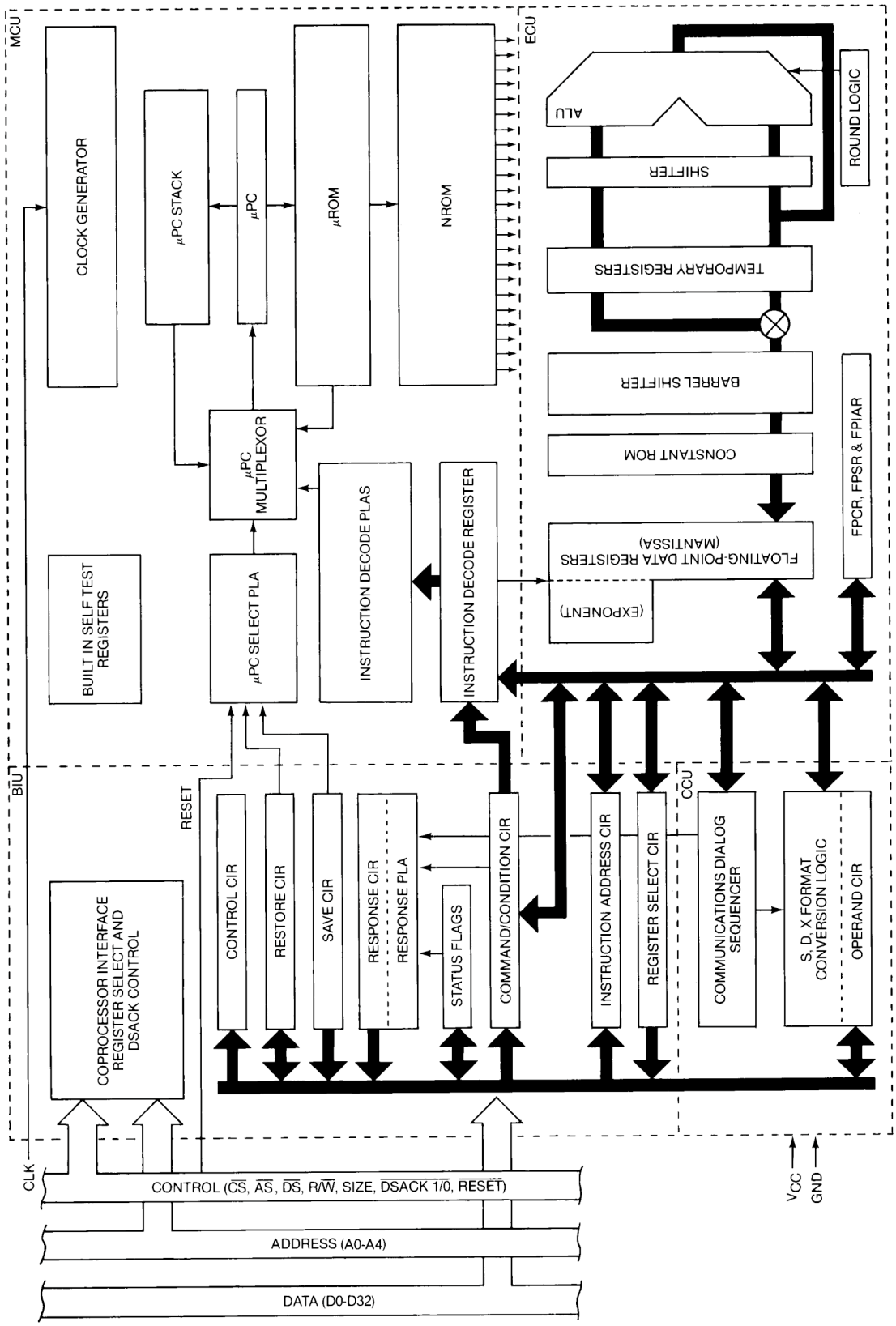
The BIU contains the co-processor interface registers, and the 32-bit control, and instruction address registers. In addition to these registers, the register select and DSACK timing control logic is contained in the BIU. Finally, the status flags used to monitor the status of communications with the main processor are contained in the BIU.

The CCU contains special-purpose hardware that performs conversions between the single, double, and extended precision memory data formula and the internal data format used by the ECU. It also contains a state machine that controls communications with the main processor during co-processor interface dialogs.

The eight 80-bit floating-point data registers (FP0-FP7) are located in the ECU. In addition to these registers, the ECU contains a high-speed 67-bit arithmetic unit used for both mantissa and exponent calculations, a barrel shifter that can shift from 1-bit to 67-bits in one machine cycle, and ROM constants (for use by the internal algorithms or user programs).

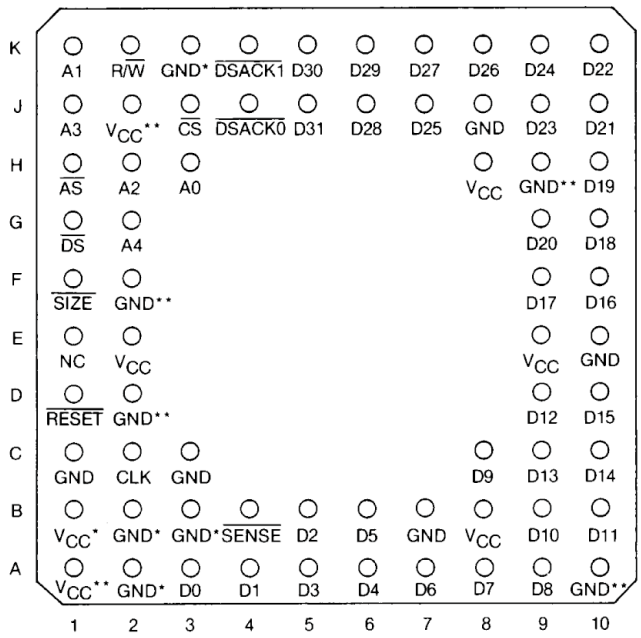
The MCU contains the clock generator, a two-level microcoded sequencer that controls the ECU, the microcode ROM, and self-test circuitry. The built-in self-test capabilities of the TS68882 enhance reliability and ease manufacturing requirements; however, these diagnostic functions are not available to the user.

Figure 1. TS68882 Simplified Block



Pin Assignments

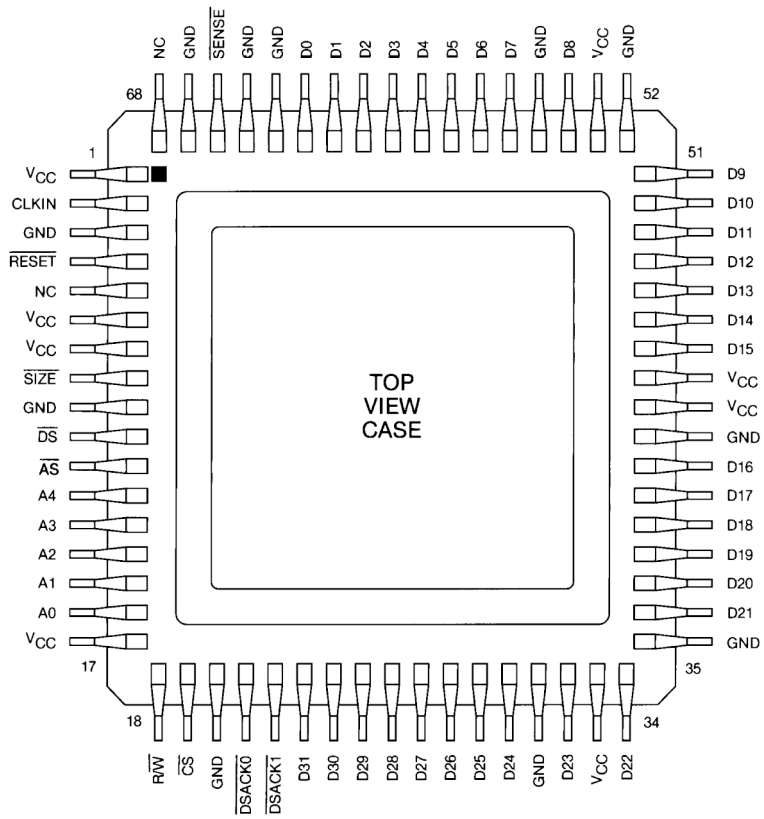
Figure 2. PGA Terminal Designation



Pin group	V_{CC}	GND
D31-D16	H8	J8
D15-D00	B8	B7
Internal logic $\overline{DSACK1}$, $\overline{DSACK0}$	E2, E9	A2, B2, B3, B4 (note), C3, E10, K3
Separate	—	C1
Extra	A1, B1, J2	A10, D2, F2, H9
Note : $\overline{SENSE}$ pin, may be used as an additional GND pin.		

* Reserved for future ATMEL-Grenoble use

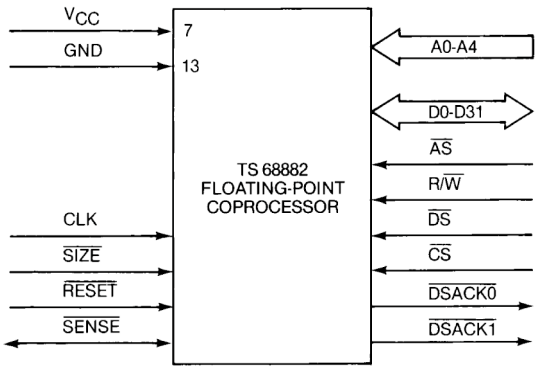
Figure 2b. CQFP Terminal Designation



Functional Signal Descriptions

This section contains a brief description of the input and output signals for the TS68882 floating-point co-processor. The signals are functionally organized into groups as shown in Figure 3.

Figure 3. TS68882 Input/output Signals



Note: The terms assertion and negation are used extensively. This is done to avoid confusion when describing “active-low” and “active-high” signals. The term assert or assertion is used to indicate that a signal is active or true, independent of whether that level is represented by a high or low voltage. The term negate or negation is used to indicate that a signal is inactive or false.

Signal Summary

Table 1 provides a summary of all the TS68882 signals described in this section.

Table 1. Signal Summary

Signal Name	Mnemonic	Input/Output	Active State	Three State
Address Bus	A0 - A4	Input	High	
Data Bus	D0 - D31	Input/Output	High	Yes
Size	$\overline{SIZE}$	Input	Low	
Address Strobe	$\overline{AS}$	Input	Low	
Chip Select	$\overline{CS}$	Input	Low	
Read/Write	R/$\overline{W}$	Input	High/Low	
Data Strobe	$\overline{DS}$	Input	Low	
Data Transfer and Size Acknowledge	$\overline{DSACK0}$, $\overline{DSACK1}$	Output	Low	Yes
Reset	$\overline{RESET}$	Input	Low	
Clock	CLK	Input		
Sense Device	$\overline{SENSE}$	Input/Output	Low	No
Power Input	V_{CC}	Input		
Ground	GND	Input		

Detailed Specifications

Scope

This drawing describes the specific requirements for the microprocessor 68882, 16.67, 20 MHz and 25 MHz, in compliance with MIL-STD-883 class B.

Applicable Documents

MIL-STD-883

1. MIL-STD-883: Test Methods And Procedures For Electronics
2. MIL-PRF-38535 Appendix A: General Specifications For Microcircuits
3. Desc Drawing 5962 - 89436xxx

Requirements

General

The microcircuits are in accordance with the applicable document and as specified herein.

Design and Construction

Terminal Connections

Depending on the package, the terminal connections shall be as shown in Figure 2 and Figure 2b.

Lead Material and Finish

Lead material and finish shall be any option of MIL-STD-1835.

Package

The macrocircuits are packaged in hermetically sealed ceramic packages which are conform to case outlines of MIL-STD-1835 (when defined):

- 68-PIN SQ.PGA UP PAE Outline
- 68-PIN Ceramic Quad Flat Pack CQFP

The precise case outlines are described on Figure 23 and Figure 24.

Electrical Characteristics

Table 2. Absolute Maximum Ratings

Symbol	Parameter	Test Conditions	Min	Max	Unit
V_{CC}	Supply Voltage		-0.3	+7.0	V
V_I	Input Voltage		-0.3	+7.0	V
P_{DMAX}	Max Power Dissipation	T_{CASE} = -55°C to +125°C		0.75	W
T_{CASE}	Operating Temperature	M Suffix	-55	+125	°C
		V Suffix	-40	+85	°C
T_{STG}	Storage Temperature		-55	+150	°C
T_{LEADS}	Lead Temperature	Max 5 sec. Soldering		+270	°C

Recommended Condition of Use

Unless otherwise stated, all voltages are referenced to the reference terminal (see Table 1).

Table 3. DC Electrical Characteristics

V_{CC} = 5.0 V_{DC} ± 10%; GND = 0 V_{DC}; Tc = -55°C to +125°C

Symbol	Parameter	Min	Max	Unit
V_{CC}	Supply Voltage	4.5	5.5	V
T_{CASE}	Operating Temperature	-55	+125	°C
V_{IH}	Input High Voltage	2.0	V_{CC}	V
V_{IL}	Input Low Voltage	GND - 0.3	0.8	V
I_{IN}	Input Leakage Current at 5.5V $\overline{CLK}$, $\overline{RESET}$, R/$\overline{W}$, A0-A4, $\overline{CS}$, $\overline{DS}$, $\overline{AS}$, $\overline{SIZE}$		10	µA
I_{TSI}	HI-Z (Off state) Input Current at 2.4V/0.4V $\overline{DSACK0}$, $\overline{DSACK1}$, D0-D31		20	µA
V_{OH}	Output High Voltage (IOH = -400 µA)(1) $\overline{DSACK0}$, $\overline{DSACK1}$, D0-D31	2.4		V
V_{OL}	Output Low Voltage (IOL = 5.3 mA)(1) $\overline{DSACK0}$, $\overline{DSACK1}$, D0-D31		0.5	V
I_{OL}	Output Low Current (VOL = GND) SENSE		500	µA
P_D	Power Dissipation		0.75	W
C_{IN}	Capacitance (V_{IN} = 0, T_A = 25°C, f = 1 MHz)(2)		20	pF
C_L	Output Load Capacitance		130	pF

Notes: 1. Test load, see Figure 5.
2. Capacitance is periodically sampled rather than 100% tested.

Thermal Characteristics

Table 4.

Package	Symbol	Parameter	Value	Rating
PGA 68	θ_{JA}	Thermal Resistance - Ceramic Junction To Ambient	33	°C/W
	θ_{JC}	Thermal Resistance - Ceramic Junction To Case	4	°C/W
CQFP	θ_{JA}	Thermal Resistance - Ceramic Junction To Ambient	33	°C/W
	θ_{JC}	Thermal Resistance - Ceramic Junction To Case	3	°C/W

Power Considerations

The average chip-junction temperature, T_J, in °C can be obtained from:

$T_J = T_A + (P_D + \theta_{JA})$ (1)

T_A = Ambient Temperature, °C

θ_{JA} = Package Thermal Resistance, Junction-to-Ambient, °C/W

$P_D = P_{INT} + P_{I/O}$

$P_{INT} = I_{CC} \times V_{CC}$, Watts - Chip Internal Power

$P_{I/O}$ = Power Dissipation on Input and Output Pins - User Determined

For most applications $P_{I/O} < P_{INT}$ and can be neglected.

An Approximate relationship between P_D and T_J (if $P_{I/O}$ is neglected) is:

$P_D = K: (T_J + 273)$ (2)

Solving equations (1) and (2) for K gives

$K = P_D . (T_A + 273) + \theta_{JA} \cdot P_D^2$ (3)

where K is constant pertaining to the particular part K can be determined from the equation (3) by measuring PD (at equilibrium) for a known T_A. Using this value of K, the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

The total thermal resistance of a package (θ_{JA}) can be separated into two components, θ_{JC} and θ_{CA}, representing the barrier to heat flow from the semiconductor junction to the package (case), surface (θ_{JC}) and from the case to the outside ambient (θ_{CA}). These terms are related by the equation:

$\theta_{JA} = \theta_{JC} + \theta_{CA}$ (4)

θ_{JA} is device related and cannot be influenced by the user. However, θ_{CA} is user dependent and can be minimized by such thermal management techniques as heat sinks, ambient air cooling and thermal convection. Thus, good thermal management on the part of the user can significantly reduce θ_{CA} so that θ_{JA} approximately equals θ_{JC}. Substitution of θ_{JC} for θ_{JA} in equation (1) will result in a lower semiconductor junction temperature.

Mechanical and Environmental

The microcircuits shall meet all mechanical environmental requirements of either MIL-STD-883 for class B devices.

Marking

The document defines the markings that are identified in the related reference documents. Each microcircuit is legible and permanently marked with the following information as minimum:

- Atmel-Grenoble Logo
- Manufacturer's Part Number
- Class B Identification
- Date-code of inspection lot
- ESD Identifier if Available
- Country of Manufacturing

Quality Conformance Inspection

DESC/MIL-STD-883

Is in accordance with MIL-M-38510 and method 5005 of MIL-STD-883. Group A and B inspections are performed on each production lot. Group C and D inspection are performed on a periodical basis.

Electrical Characteristics

General Requirements

All static and dynamic electrical characteristics specified and the relevant measurement conditions are given below. For inspection purpose, refer to relevant specification:

Static electrical characteristics for all electrical variants.

Dynamic electrical characteristics for 68882-16 (16.67 MHz), 68882-20 (20 MHz), 68882-25 (25 MHz) and 68882-33 (33 MHz).

For static characteristics, test methods refer to clause "Test Load" on page 13 hereafter of this specification (Table 5).

For dynamic characteristics (Tables 6 and 7), test methods refer to IEC 748-2 method number, where existing.

Table 5. Static Characteristics

V_{CC} = 5.0 V_{DC} ± 10%; GND = 0 V_{DC}; Tc = -55/+125°C or -40/+85°C

Symbol	Parameter	Min	Max	Unit
V_{IH}	Input High Voltage	2.0	V_{CC}	V
V_{IL}	Input Low Voltage	GND - 0.3	0.8	V
I_{IN}	Input Leakage Current at 5.5V CLK, $\overline{RESET}$, R/$\overline{W}$, A0-A4, $\overline{CS}$, $\overline{DS}$, $\overline{AS}$, $\overline{SIZE}$		10	µA
I_{TSI}	HI-Z (off state) Input Current at 2.4V/0.4V $\overline{DSACK0}$, $\overline{DSACK1}$, D0-D31		20	µA
V_{OH}	Output High Voltage (I_{OH} = -400 µA)(1) $\overline{DSACK0}$, $\overline{DSACK1}$, D0-D31	2.4		V
V_{OL}	Output Low Voltage (I_{OL} = 5.3 mA)(1) $\overline{DSACK0}$, $\overline{DSACK1}$, D0-D31		0.5	V
I_{OL}	Output Low Current (V_{OL} = GND) $\overline{SENSE}$		500	µA

Table 5. Static Characteristics

V_{CC} = 5.0 V_{DC} ± 10%; GND = 0 V_{DC}; Tc = -55/+125°C or -40/+85°C

Symbol	Parameter	Min	Max	Unit
I_{CC}	Maximum Supply Current (V_{CC} = 5.5V; CLK = f_{max}; part in Reset)		136	mA
C_{in}	Capacitance (V_{IN} = 0, T_A = 25°C, f = 1MHz)(2)		20	pF
C_L	Output Load Capacitance		130	pF

Notes: 1. Test load, see Figure 5.
2. Capacitance is periodically sampled rather than 100% tested.

Dynamic (Switching) Characteristics

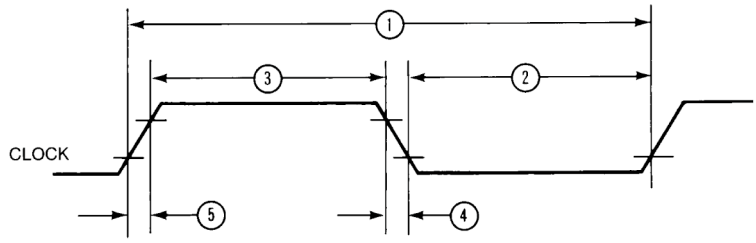
The limits and values given in this section apply over the full case temperature range -55°C to +125°C and V_{CC} in the range 4.5V to 5.5V, See "AC Electrical Specification Definitions" on page 13.

The numbers (N°) refer to the timing diagrams. See Figure 4, Figure 6, Figure 7, Figure 8 and Figure 9.

Table 6. AC Electrical Characteristics - Clock Input
V_{CC} = 5.0 V_{DC} ± 10%; GND = 0 $V_{DC;}$ Tc = -55°C to +125°C (see Figure 4)

		16.67 MHz		20 MHz		25 MHz		33.33 MHz		
N°	Parameter	Min	Max	Min	Max	Min	Max	Min	Max	Unit
	Frequency of Operation	8	16.67	12.5	20	12.5	25	16.7	33.33	MHz
1	Clck Time	60	125	50	80	40	80	30	60	ns
2, 3	Clock Pulse Width	24	95	20	54	15	59	14	66	ns
4, 5	Rise and Fall Times		5		5		4		3	ns

Figure 4. Clock Input Timing Diagram



Note: Timing measurements are referenced to and from a low voltage of 0.8V and a high voltage of 2.0V, unless otherwise noted. The voltage swing through this range should start outside, and pass through, the range such that the rise of fall will be linear between 0.8V and 2.0V.

Table 7. AC Electrical Characteristics – Read and Write Cycles[1]
V_{CC} = 5.0 V_{DC} ± 10%; GND = 0 $V_{DC;}$ Tc = -55°C/+125°C or Tc = -40°C/+85°C (see Figure 7, Figure 8, Figure 9)

N°	Parameter	16.67 MHz		20 MHz		25 MHz		33.33 MHz		Unit
		Min	Max	Min	Max	Min	Max	Min	Max	
6	Address valid to $\overline{AS}$ asserted[5]	15		10		5		5		ns
6a	Address valid to $\overline{DS}$ asserted (read)[5]	15		10		5		5		ns
6b	Address valid to $\overline{DS}$ asserted (write)[5]	50		50		35		26		ns
7	$\overline{AS}$ negated to address invalid[6]	10		10		5		5		ns
7a	$\overline{DS}$ negated to address invalid[6]	10		10		5		5		ns
8	$\overline{CS}$ asserted to $\overline{AS}$ asserted or $\overline{AS}$ asserted to $\overline{CS}$ asserted[9]	0		0		0		0		ns
8a	$\overline{CS}$ asserted to $\overline{DS}$ asserted or $\overline{DS}$ asserted to $\overline{CS}$ asserted (read)[9]	0		0		0		0		ns
8b	$\overline{CS}$ asserted to $\overline{DS}$ asserted or $\overline{DS}$ asserted to $\overline{CS}$ asserted (write)[9]	30		25		20		15		ns
9	$\overline{AS}$ negated to $\overline{CS}$ negated	10		10		5		5		ns
9a	$\overline{DS}$ negated to $\overline{CS}$ negated	10		10		5		5		ns
10	R/$\overline{W}$ high to $\overline{AS}$ asserted (read)	15		10		5		5		ns
10a	R/$\overline{W}$ high to $\overline{DS}$ asserted (read)	15		10		5		5		ns
10b	R/$\overline{W}$ low to $\overline{DS}$ asserted (write)	35		30		25		25		ns
11	$\overline{AS}$ negated to R/$\overline{W}$ low (read) or $\overline{AS}$ negated to R/$\overline{W}$ high (write)	10		10		5		5		ns
11a	$\overline{DS}$ negated to R/W low (read) or $\overline{DS}$ negated to R/W high (write)	10		10		5		5		ns
12	$\overline{DS}$ width asserted (write)	40		38		30		23		ns
13	$\overline{DS}$ width negated	40		38		30		23		ns
13a	$\overline{DS}$ negated to $\overline{AS}$ asserted[4]	30		30		25		18		ns
14	$\overline{CS}$, $\overline{DS}$ asserted to data-out valid (read)[2]		80		45		45		30	ns
15	$\overline{DS}$ negated to data-out invalid (read)	0		0		0		0		ns
16	$\overline{DS}$ negated to data-out high impedance (read)		50		35		35		30	ns
17	Data-in invalid to $\overline{DS}$ asserted (write)	15		10		5		5		ns
18	$\overline{DS}$ negated to data-in invalid (write)	15		10		5		5		ns
19	$\overline{START}$ true to $\overline{DSACK0}$ and $\overline{DSACK1}$ asserted[2]		50		35		25		20	ns
19a	$\overline{DSACK0}$ asserted to $\overline{DSACK1}$ asserted (skew)[7]	-15	15	-10	10	-10	10		5	ns
20	$\overline{DSACK0}$ or $\overline{DSACK1}$ asserted to data-out valid		50		43		32		17	ns
21	$\overline{START}$ false to $\overline{DSACK0}$ and $\overline{DSACK1}$ negated[8]		50		30		40		30	ns

Table 7. AC Electrical Characteristics – Read and Write Cycles[1] (Continued)
V_{CC} = 5.0 V_{DC} ± 10%; GND = 0 $V_{DC;}$ Tc = -55°C/+125°C or Tc = -40°C/+85°C (see Figure 7, Figure 8, Figure 9)

N°	Parameter	16.67 MHz		20 MHz		25 MHz		33.33 MHz		Unit
		Min	Max	Min	Max	Min	Max	Min	Max	
22	$\overline{START}$ false to $\overline{DSACK0}$ and $\overline{DSACK1}$ high impedance[8]		70		55		55		40	ns
23	$\overline{START}$ true to clock high (synchronous read)[3][8]	0		0		0		0		ns
24	Clock low to data-out valid synchronous read)[3]		105		80		60		45	ns
25	$\overline{START}$ true to data-out valid (synchronous read)[3][8]	0 1.5	105+ 2.5	1.5	80 + 2.5	1.5	60+ 2.5	1.5	45- 2.5	ns Clks
26	Clock low to $\overline{DSACK0}$ and $\overline{DSACK1}$ asserted (synchronous read[3]		75		55		45		30	ns
27	$\overline{START}$ true to $\overline{DSACK0}$ and $\overline{DSACK1}$ asserted (synchronous read) [3][8]	1.5	75+ 2.5	1.5	55+ 2.5	1.5	45+ 2.5	1.5	30- 2.5	ns Clks

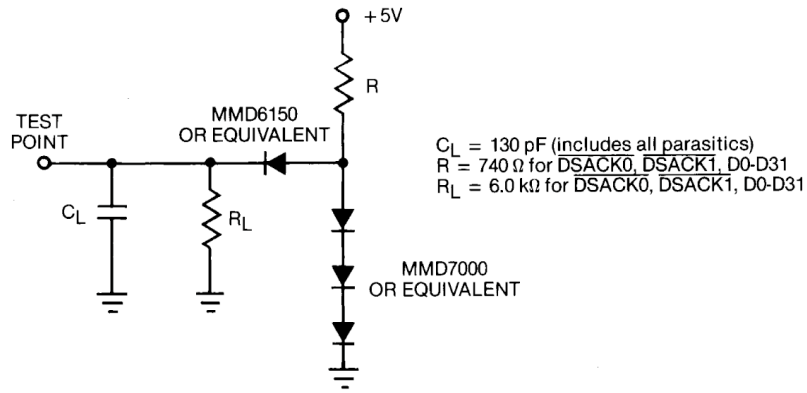
Notes:
1. Timing measurements are referenced to and from a low voltage of 0.8V and a high voltage of 2.0V, unless otherwise noted. The voltage swing through this range should start outside, and pass through, the range such that the rise or fall will be linear between 0.8V and 2.0V.
2. These specifications only apply if the TS68882 has completed all internal operations initiated by the termination of the previous bus cycle when $\overline{DS}$ was negated.
3. Synchronous read cycles occur only when the save or response CIR locations are read.
4. This specification only applies to systems in which back-to-back accesses (read-write or write-write) of the operand CIR can occur. When the TS68882 is used as a co-processor to the TS68020/68030, this can occur when the addressing mode is immediate.
5. If the $\overline{SIZE}$ pin is not strapped to either V_{CC} or GND, it must have the same setup times as do addresses.
6. If the $\overline{SIZE}$ pin is not strapped to either V_{CC} or GND, it must have the same hold times as do addresses.
7. This number is reduced to 5 nanoseconds if $\overline{DSACK0}$ and $\overline{DSACK1}$ have equal loads.
8. $\overline{START}$ is not an external signal; rather, it is the logical condition that indicates the start of an access. The logical equation for this condition is $\overline{START} = \overline{CS} + \overline{AS} + (R/\overline{W} \cdot \overline{DS})$.
9. If a subsequent access is not a FPCP access, $\overline{CS}$ must be negated before the assertion of $\overline{AS}$ and/or $\overline{DS}$ on the non-FPCP access. These specifications replace the old specifications 8 and 8A (the old specifications implied that in all cases, transitions in $\overline{CS}$ must not occur simultaneously with transitions of $\overline{AS}$ or $\overline{DS}$. This is not a requirement of the TS68882).

Test Conditions Specific to the Device

Test Load

The applicable loading network shall be as defined in column “Test conditions” of Table 2, referring to the loading network number as shown in Figure 5.

Figure 5. Test Loads



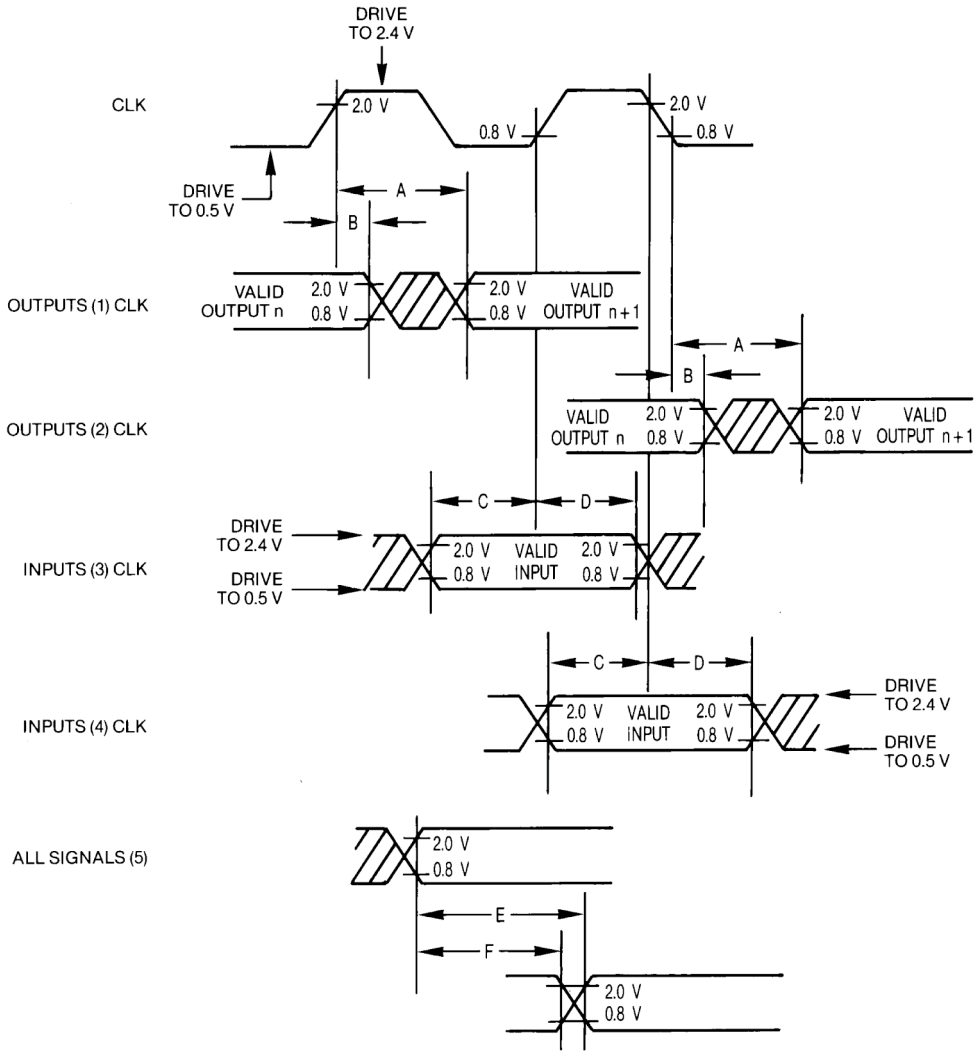
AC Electrical Specification Definitions

The AC specifications presented consist of output delays, input setup and hold times, and signal skew times. All signals are specified relative to an appropriate edge of the clock input and, possibly, relative to one or more other signals.

The measurement of the AC specifications is defined by the waveforms shown in Figure 6. In order to test the parameters guaranteed inputs must be driven to the voltage levels specified in Figure 6. Outputs are specified with minimum and/or maximum limits, as appropriate, and are measured as shown. Inputs are specified with minimum and, an appropriate maximum setup and hold times, and are measured as shown. Finally, the measurement for signal-to-signal specifications are also shown.

Note that the testing levels used to verify conformance to the AC specifications does not affect the guaranteed DC operation of the device specified in the DC electrical characteristics.

Figure 6. Drive Levels and Test Points for AC Specifications



Legend

A) Maximum output delay specification.

B) Minimum output hold time.

C) Minimum input setup time specification.

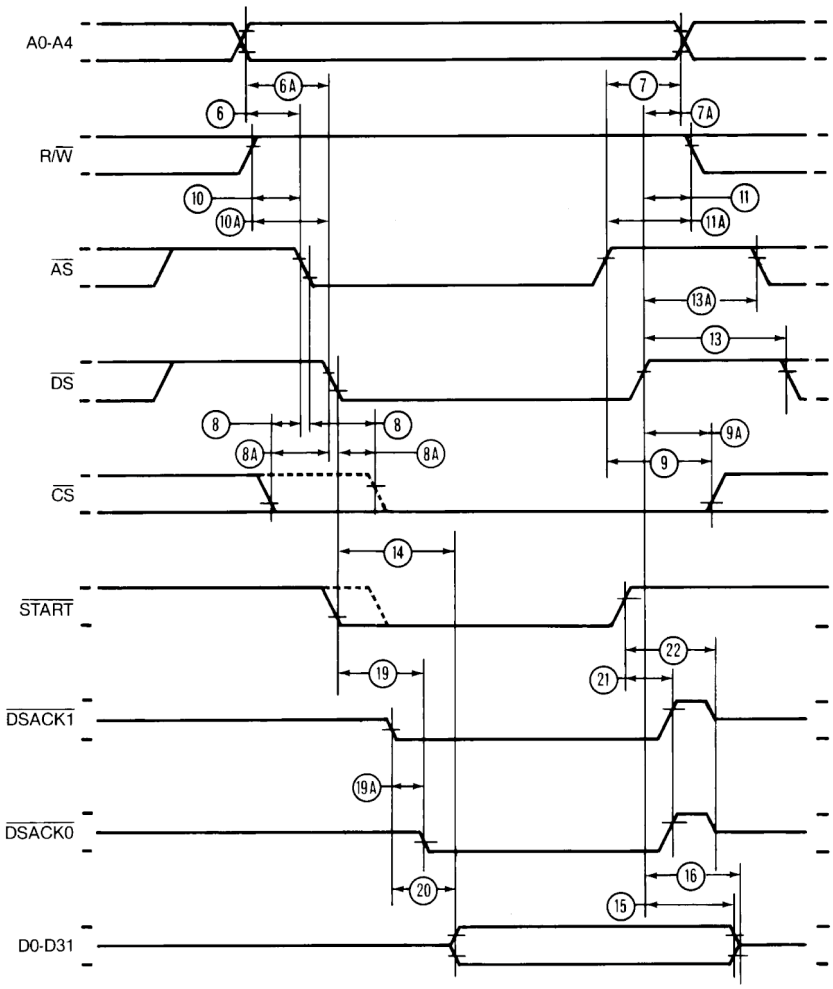
D) Minimum input hold time specification.

E) Signal valid to signal valid specification (maximum or minimum).

F) Signal valid to signal invalid specification (maximum or minimum).

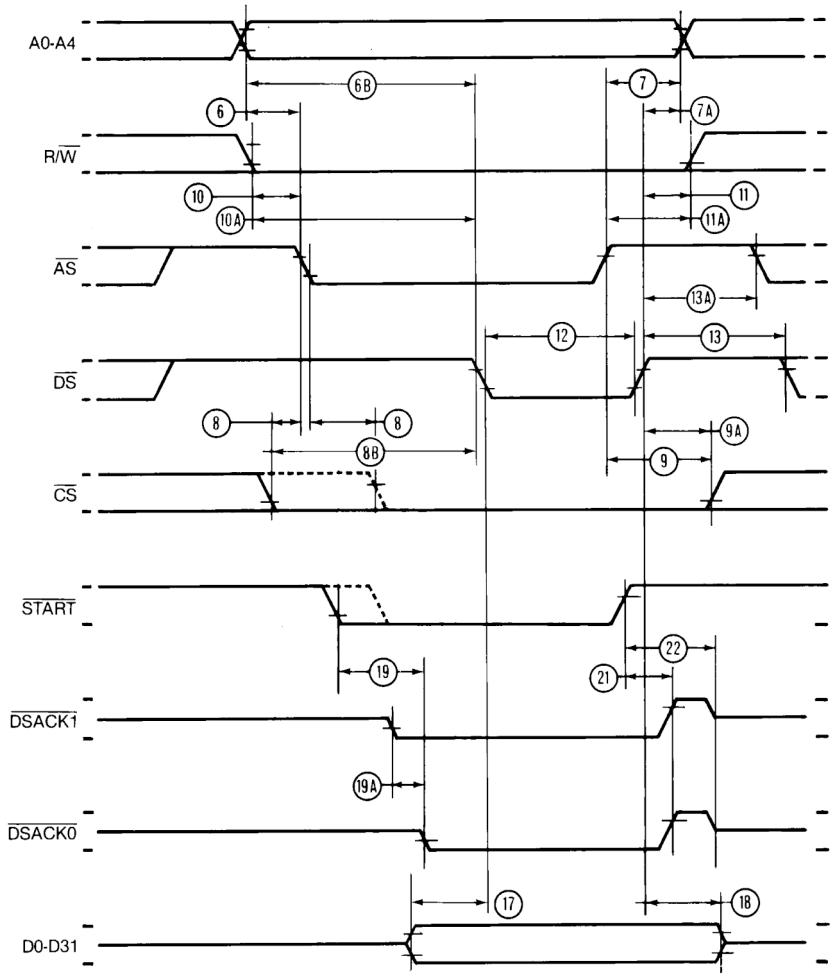
Notes:
1. This output timing is applicable to all parameters specified relative to the rising edge of the clock.
2. This output timing is applicable to all parameters specified relative to the falling edge of the clock.
3. This input timing is applicable to all parameters specified relative to the rising edge of the clock.
4. This input timing is applicable to all parameters specified relative to the falling edge of the clock.
5. This timing is applicable to all parameters specified relative to the assertion/negation of another signal.

Figure 7. Asynchronous Read Cycle Timing Diagram



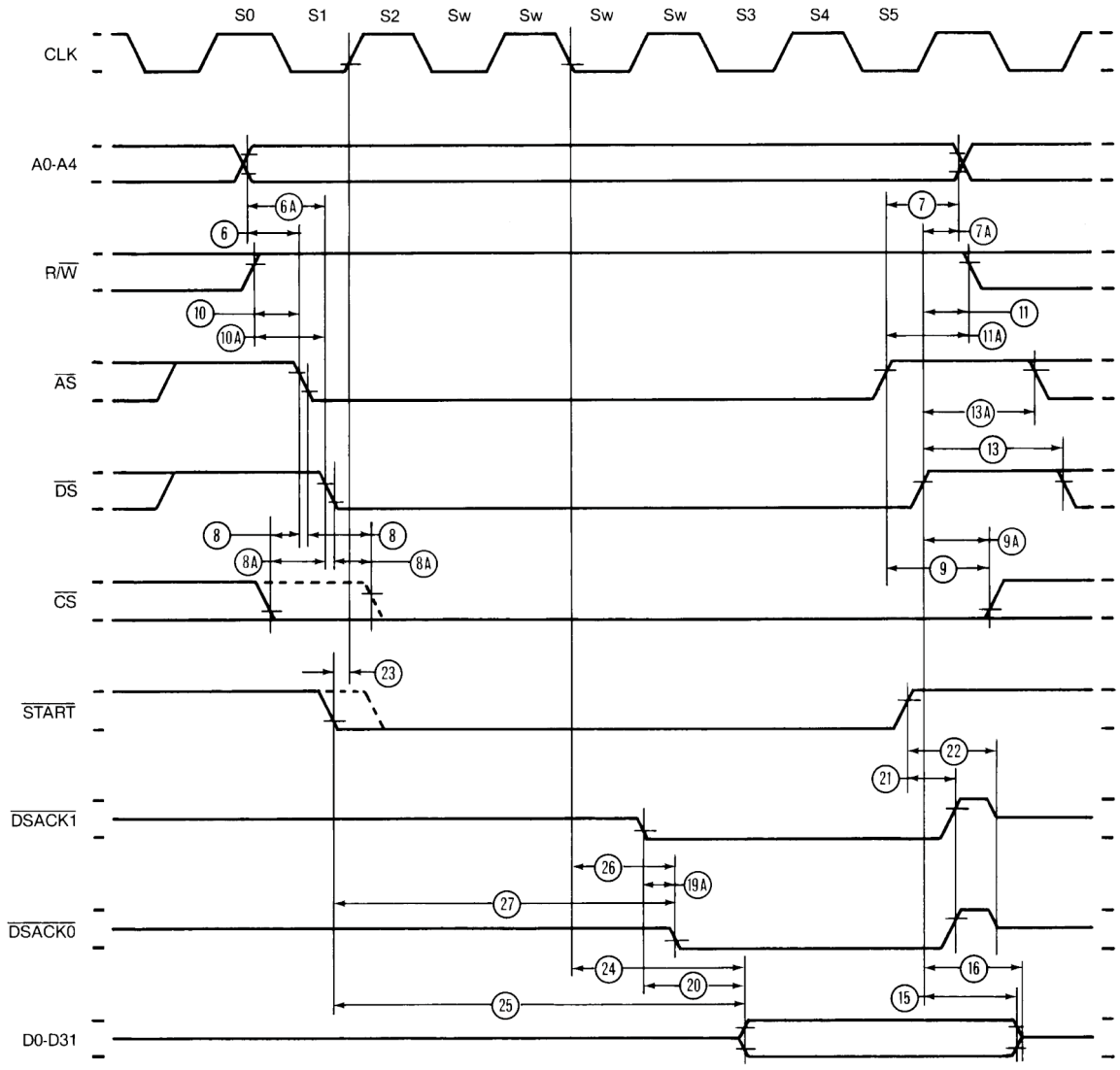
Note: $\overline{\text{START}}$ is actually a logical condition, but is shown as an active signal for clarity. The logical equation for this signal is: $\overline{\text{START}} = \overline{\text{CS}} + \overline{\text{AS}} + (\text{R}/\overline{\text{W}} \cdot \overline{\text{DS}})$.

Figure 8. Asynchronous Write Cycle Timing Diagram



Note: $\overline{START}$ is actually a logical condition, but is shown as an active signal for clarity. The logical equation for this signal is: $\overline{START} = \overline{CS} + \overline{AS} + (R/\overline{W} \cdot \overline{DS})$.

Figure 9. Synchronous Read Cycle Timing Diagram



Note: $\overline{START}$ is actually a logical condition, but is shown as an active signal for clarity. The logical equation for this signal is: $\overline{START} = \overline{CS} + \overline{AS} + (R/\overline{W} \cdot \overline{DS})$.

Additional Information

Additional information shall not be for any inspection purposes.

Capacitance (Not for Inspection Purposes)

Symbol	Parameter	Test Conditions	Min	Max	Unit
C_{in}	Input Capacitance	Vin = 0 T_{amb} = 25°C f = 1 MHz		20	pF

Functional Description

The Co-processor Concept

The TS68882 functions as a co-processor in systems where the TS68020 or TS68030 is the main processor via the TS68000 co-processor interface. It functions as a peripheral processor in systems where the main processor is the TS68000, TS68010.

The TS68882 utilizes the TS68000 Family co-processor interface to provide extension of the TS68020 /TS68030 registers and instruction set in a manner which is transparent to the programmer. The programmer perceives the MPU/FPCP execution model as if both devices are implemented on one chip.

A fundamental goal of the TS68000 Family co-processor interface is to provide the programmer with an execution model based upon sequential instruction execution by the TS68020/TS68030 and the TS68882. For optimum performance, however, the co-processor interface allows concurrent operations in the TS68882 with respect to the TS68020/TS68030 whenever possible. In order to simplify the programmer's model, the co-processor interface is designed to emulate, as closely as possible, non-concurrent operation between the TS68020/TS68030 and the TS68882.

The TS68882 is s non-DMA type co-processor which uses a subset of the general-purpose co-processor interface supported by the TS68020/TS68030. Features of the interface implemented in the TS68882 are as follows:

- The main processor(s) and TS68882 communicate via standard TS68000 bus cycles
- The main processor(s) and TS68882 communications are not dependent upon the instruction sets or internal details of the individual devices (i.e., instruction pipes or caches, addressing modes)
- The main processor(s) and TS68882 may operate at different clock speeds
- TS68882 instructions utilize all addressing modes provided by the main processor; all effective addresses are calculated by the main processor at the request of the co-processor
- All data transfers are performed by the main processor at the request of the TS68882; thus memory management, bus errors, address errors, and bus arbitration function as if the TS68882 instructions are executed by the main processor
- Overlapped (concurrent) instruction execution enhances throughput while maintaining the programmer's model of sequential instruction execution
- Co-processor detection of exceptions which require a trap to be taken are serviced by the main processor at the request of the TS68882 thus exception processing functions as if the TS68882 instructions were executed by the main processor
- Support of virtual memory/virtual machine systems is provided via the FSAVE and FRESTORE instructions
- Up to eight co-processors may reside in a system simultaneously: multiple co-processors of the same type are also allowed
- Systems may use software emulation of the TS68882 without reassembling or relinking user software

The TS68882 programming model is shown in Figure 10 through 15, and consists of the following:

- Eight 80-bit floating-point data registers (FP0-FP7). These registers are analogous to the integer data registers (D0-D7) and are completely general-purpose (i.e., any instruction may use any register)
- A 32-bit control register that contains enable bits for each class of exceptions trap, and mode bits to set the user-selectable rounding and precision modes
- A 32-bit status register that contains floating-point condition codes, quotient bits, and exception status information
- A 32-bit instruction address register that contains the main processor memory address of the last floating-point instruction that was executed. This address is used in exception handling to locate the instruction that caused the exception

The connection between the TS68020/TS68030 and the TS68882 is a simple extension of the TS68000 bus interface. The TS68882 is connected as a co-processor to the TS68020/TS68030, and the selection of the TS68882 is based upon a chip select (CS), which is decoded from the TS68020/TS68030 function codes and address bus. Figure 16 illustrates the TS68882/TS68020 or TS68030 configuration.

Figure 10. TS68882 Programming Model

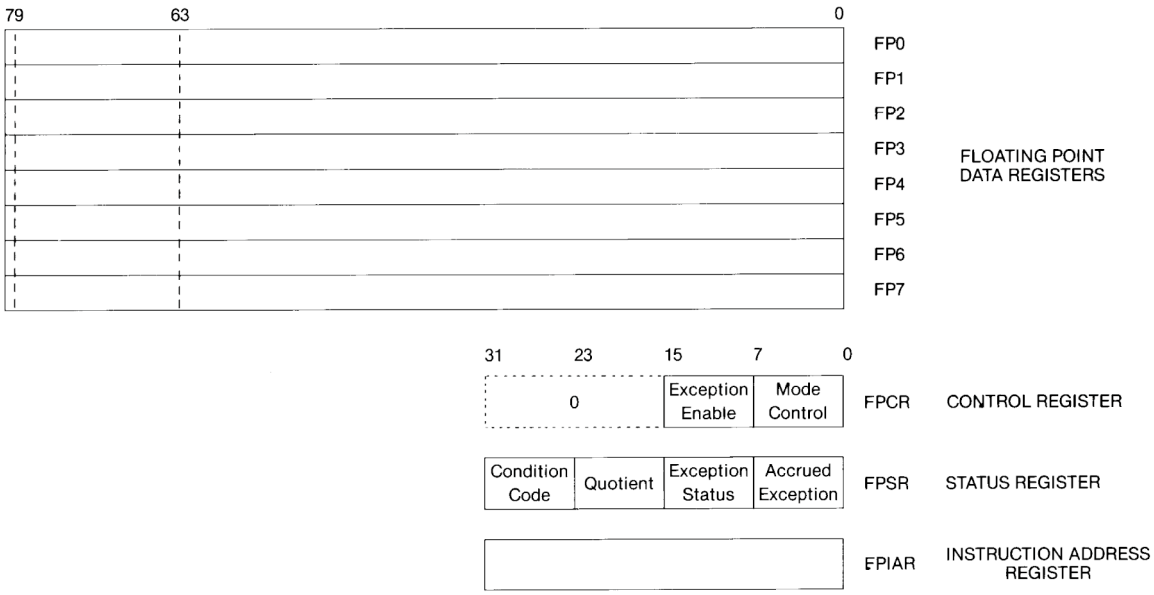


Figure 11. Exception Status/Enable Byte

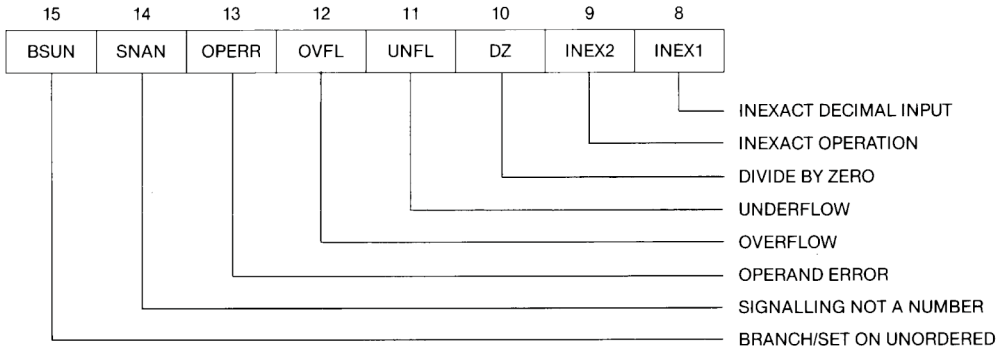


Figure 12. Mode Control Byte

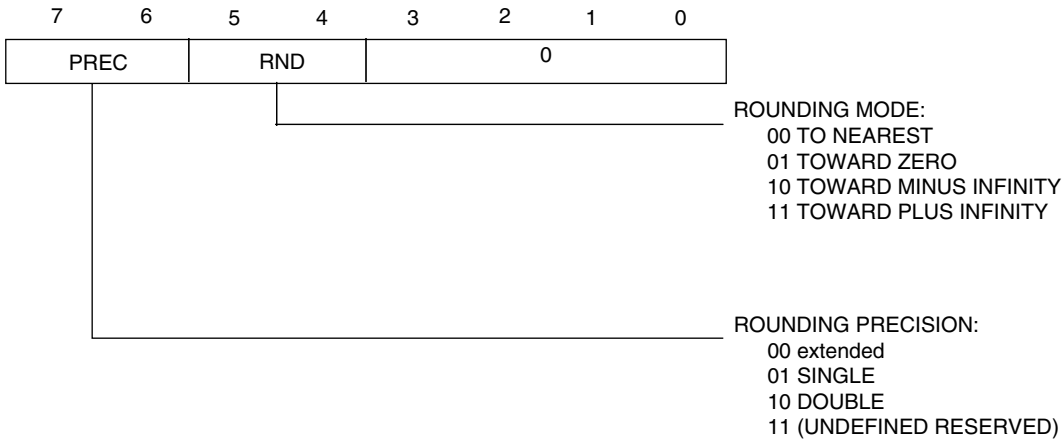


Figure 13. Condition Code Byte

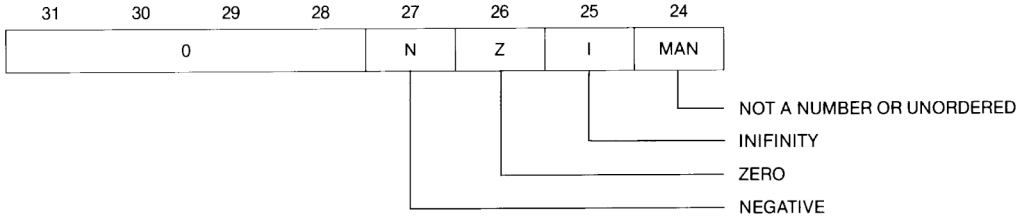


Figure 14. Quotient Byte

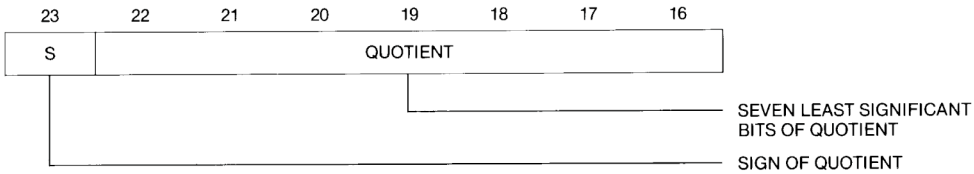


Figure 15. Accrued Exception Byte

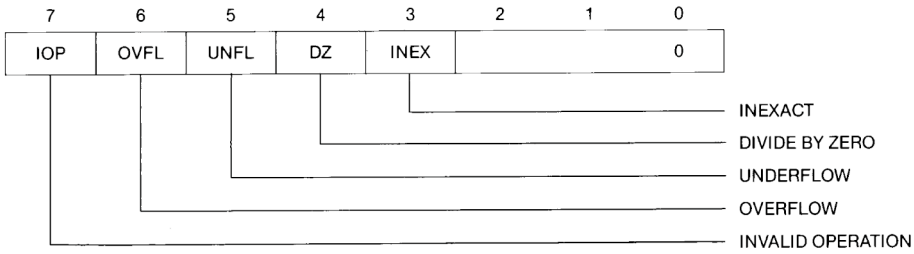
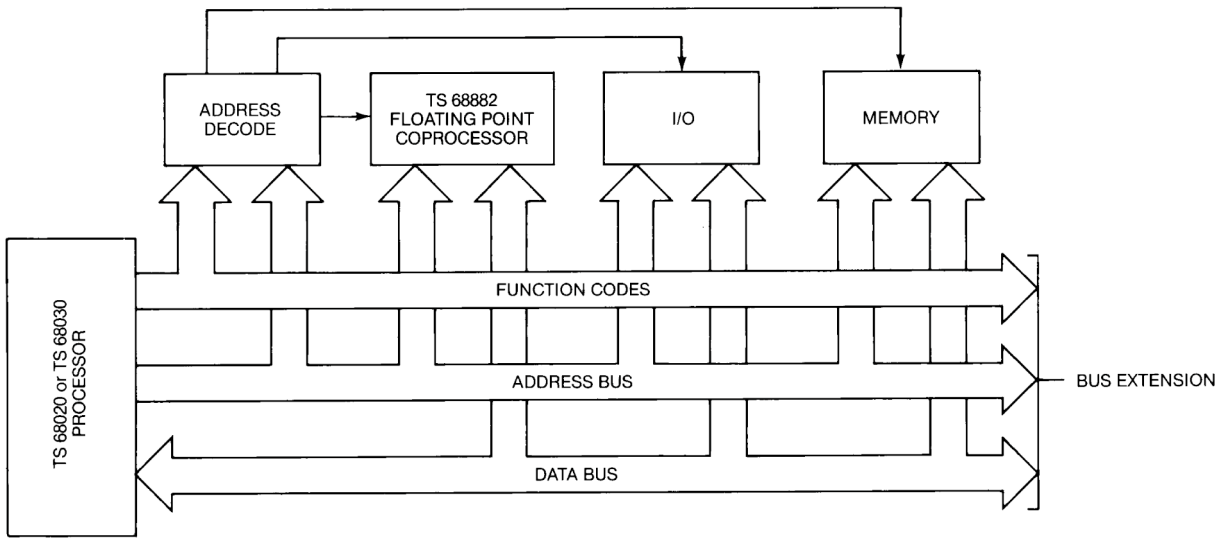


Figure 16. Typical Co-processor Configuration



Bus Interface Unit

All communications between the TS68020/TS68030 and the TS68882 occur via standard TS68000 Family bus transfers. The TS68882 is designed to operate on 8-, 16-, or 32-bit data buses.

The TS68882 contains a number of co-processor interface registers (CIRs) which are addresses in the same manner as memory by the main processor. The TS68000 Family co-processor interface is implemented via a protocol of reading and writing to these registers by the main processor. The TS68020 and TS68030 implements this general-purpose co-processor interface protocol in hardware and microcode.

When the TS68020/TS68030 detects a typical TS68882 instruction, the MPU writes the instruction to the memory-mapped command CIR, and reads the response CIR. In this response, the BIU encodes requests for any additional action required of the MPU on behalf of the TS68882. For example, the response may request that the MPU fetch an operand from the evaluated effective address and transfer the operand to the operated CIR. Once the MPU fulfills the co-processor request(s), it is free to fetch and execute subsequent instructions.

A key concern in a co-processor interface that allows concurrent instruction execution is synchronization during main processor and co-processor communication. If a subsequent instruction is written to the TS68882 before the CCU has passed the operands for the previous instructions to the ECU, the response instructs the TS68020/TS68030 to wait. Thus, the choice of concurrent or nonconcurrent instruction execution is determined on an instruction-by-instruction basis by the co-processor.

The only difference between a co-processor bus transfer and any other bus transfer is that the TS68020/TS68030 issues a function code to indicate the CPU address space during the cycle (the function codes are generated by the TS68000 Family processors to identify eight separate address spaces). Thus, the memory-mapped co-processor interface registers do not infringe upon instruction or data address spaces. The TS68020/TS68030 places a co-processor ID field from the co-processor instruction onto three of the upper address lines during co-processor accesses. This ID, along with the CPU address space function code, is decoded to select one of eight co-processors in the system.

Since the co-processor interface protocol is based solely on bus transfers, the protocol is easily emulated by software when the TS68882 is used as a peripheral with any processor capable of memory-mapped I/O over on TS68000 style bus. When used as a peripheral processor with the 8-bit TS68008 or the 16-bit TS68000, or TS68010, all TS68882 instructions are trapped by the main processor to an exception handler at execution time. Thus, the software emulation of the processor interface protocol can be totally transparent to the user. The system can be quickly upgraded by replacing the main processor with a TS68020/TS68030 without changes to the user software.

Since the bus is asynchronous, the TS68882 need not run at the same clock speed as the main processor. Total system performance may therefore be customized. For example, a system requiring very fast floating-point arithmetic with relatively slow integer arithmetic can be designed with an inexpensive main processor and a fast TS68882.

Co-processor Interface

The TS68000 Family co-processor interface is an integral part of the TS68882 and TS68020/TS68030 designs, with the interface tasks shared between the two. The interface is fully compatible with all present and future TS68000 Family products. Tasks are partitioned such that the TS68020/TS68030 does not have to decode co-processor instructions and, the TS68882 does not have to duplicate main processor functions such as effective address evaluation.

This partitioning provides an orthogonal extension of the instruction set by permitting TS68882 instructions to utilize all TS68020/TS68030 addressing modes and to generate execution time exception traps. Thus, from the programmer's view, the CPU and co-processor appear to be integrated onto a single chip. While the execution of the majority of TS68882 instructions may be overlapped with the execution of TS68020/TS68030 instructions, concurrency is completely transparent to the programmer. The TS68020/TS68030 single-step and program flow (trace) modes are fully supported by the TS68882 and the TS68000 Family co-processorco-processor interface.

While the TS68000 Family co-processor interface permits co-processors to be bus masters, the TS68882 is never a bus master. The TS68882 requests that the TS68020/TS68030 fetch all operands and store all results. In this manner, the TS68020/TS68030 32-bit data bus provides high speed transfer of floating-point operands and results while simplifying the design of the TS68882.

Since the co-processor interface is based solely upon bus cycles and the TS68882 is never a bus master, the TS68882 can be placed on either the logical or physical side of the system memory management unit. This provides a great deal of flexibility in the system design.

The virtual machine architecture of the TS68000 Family is supported by the co-processor interface and the TS68882 through the FSAVE and FRESTORE instructions. If the TS68020/TS68030 detects a page fault and/or task time out, it can force the TS68882 to stop whatever operation is in process at any time (even in the middle of the execution of an instruction) and save the TS68882 internal state in memory.

The size of the saved internal state of the TS68882 is dependent upon what the CCU and ECU are doing at the time that the FSAVE is executed. If the TS68882 is in the reset state when the FSAVE instruction is received, only one word of state is transferred to memory, which may be examined by the operating system to determine that the co-processor programmer's model is empty. If the co-processor is idle when the save instruction is received, only a few words of internal state are transferred to memory. If the TS68882 is in the middle of performing a calculation, it may be necessary to save the entire internal state of the machine. Instructions that can complete execution in less time than it would take to save the larger state in mid-instruction are allowed to complete execution and then save the idle state.

Thus the size of the saved internal state is kept to a minimum. The ability to utilize several internal state sizes greatly reduces the average context switching time.

The FRESTORE instruction permits reloading of an internal state that was saved earlier, and continue any operation that was previously suspended. Restoring of the reset internal state functions just like a hardware reset to the TS68882 in that defaults are re-established.

Note: Though the TS68882 is instruction set compatible with the TS68881, the idle and busy state frames are both 32 bytes larger on the TS68882 than on the TS68881. A unique format word is generated by the TS68882 so that system software can detect this difference.

Operand Data Formats

The TS68882 supports the following data formats:

- Byte Integer (B)
- Word Integer (W)
- Long Word Integer (L)
- Single Precision Real (S)
- Double Precision Real (D)
- Extended Precision Real (X)
- Packed Decimal String Real (P)

The capital letters contained in parenthesis denote suffixes added to instructions in the assembly language source to specify the data format to be used.

Integer Data Formats

The three Integer data formats (byte, word, and long word) are the standard data formats supported in the TS68000 Family architecture. Whenever an integer is used in a floating-point operation, the integer is automatically converted by the TS68882 to an extended precision floating-point number before being used. For example, to add an integer constant of five to the number contained in floating-point data register 3 (FP3), the following instruction can be used:

```
FADD.W #5.FP3
```

The ability to effectively use integers in floating-point operations saves user memory since an integer representation of a number, if representable, is usually smaller than the equivalent floating-point representation.

Floating-point Data Formats

The floating-point data formats single precision (32-bits) and double precision (64-bits) are as defined by the IEEE standard. These are the main floating-point formats and should be used for most calculations involving real numbers. Table 8 lists the exponent and mantissa size for single, double, and extended precision. The exponent is biased, and the mantissa is in sign and magnitude form. Since single and double precision require normalized numbers, the most significant bit of the mantissa is implied as one and is not included, thus giving one extra bit of precision.

Table 8. Exponent and Mantissa Sizes

Data Format	Exponent Bits	Mantissa Bits	Bias
Single	8	23 (+1)	127
Double	11	52 (+1)	1023
Extended	15	64	16383

The extended precision data format is also in conformance with the IEEE standard, but the standard does not specify this format to the bit level as it does for single and double precision. The memory format on the TS68882 consists of 96 bits (three long words). Only 80 bits are actually used, the other 16 bits are for future expandability and for long-word alignment of floating-point data structures. Extended format has a 15-bit exponent, a 64-bit mantissa, and a 1-bit mantissa sign.

Extended precision numbers are intended for use as temporary variables, intermediate values, or in places where extra precision is needed. For example, a compiler might select extended precision arithmetic for evaluation of the right side of an equation with mixed sized data and then convert the answer to the data type on the left side of the equation. It is anticipated that extended precision data will not be stored in large arrays, due to the amount of memory required by each number.

Packed Decimal String Real Data Format

The packed decimal string data format allows packed BCD strings to be input to and output from the TS68882. The strings consist of a 3-digit base 10 exponent and a 17-digit base 10 mantissa. Both the exponent and mantissa have a separate sign bit. All digits are packed BCD, such that an entire string fits in 96 bits (three long words). As is the case with all data formats, when packed BCD strings are input to the TS68882, the strings are automatically converted to extended precision real values. This allows packed BCD numbers to be used as inputs to any operation. For example:

```
FADD.P # - 6.023E + 24, FP5
```

BCD numbers can be output from the TS68882 in a format readily used for printing by a program generated by a high-level language compiler. For example:

```
FMOVE.P FP3.BUFFER (# -5)
```

instructs the TS68882 to convert the floating-point data register 3 (FP3) contents into a packed BCD string with five digits to the right of the decimal point (FORTRAN F format).

Data Format Summary

All data formats described above are supported orthogonally by all arithmetic and transcendental operations, and by all appropriate TS68000 Family addressing modes. For example, all of the following are legal instructions:

FADD.B	**# 3.FP0**
FADD.W	D2.FP3
FADD.L	BIGINT.FP7
FADD.S	# 3.14159.FP5
FADD.D	(SP) + .FP6
FADD.X	[(TEMP -PTR.A7)].FP3
FADD.P	# 1.23E25.FP0

On-chip calculations are performed to extended precision format, and the eight floating-point data registers always contain extended precision values. All data used in an operation is converted to extended precision by the TS68882 before the specific operation is performed, and all results are in extended precision. This ensures accuracy without sacrificing performance.

Refer to Figure 17 for a summary of the memory formats for the seven data formats supported by the TS68882.