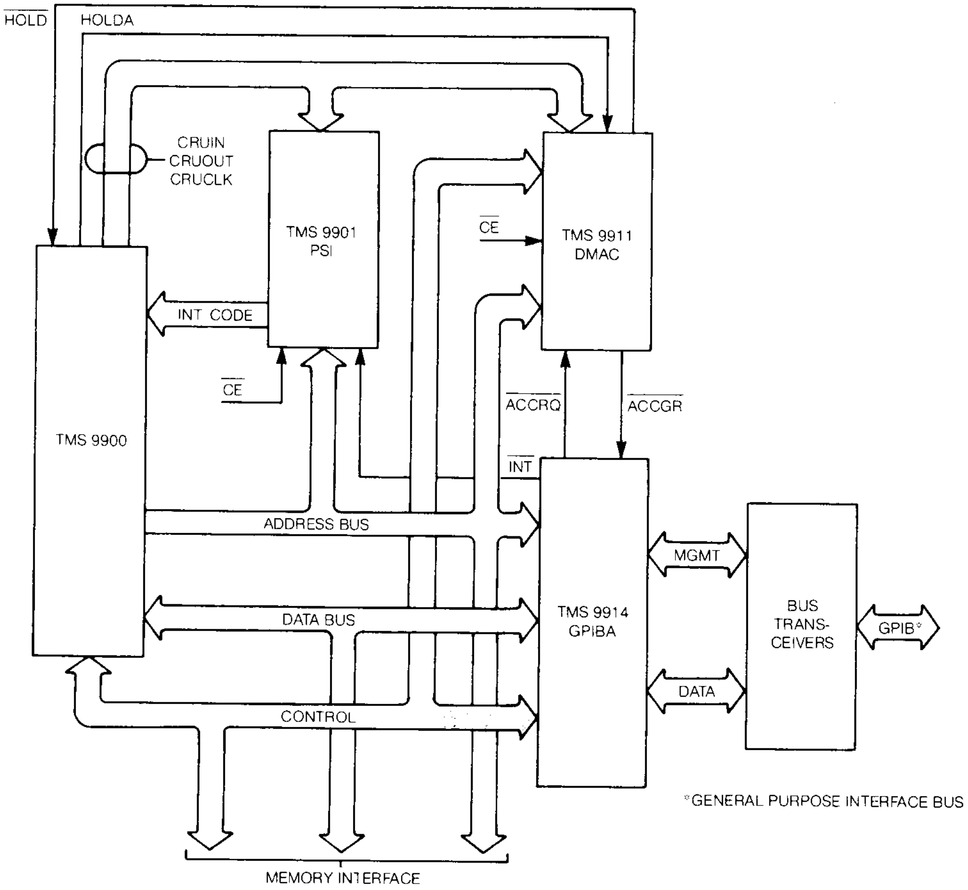
TMS 9914 GENERAL PURPOSE INTERFACE BUS ADAPTER

990/9900 FAMILY MICROCOMPUTER COMPONENTS

- IEEE Std. 488-1975 Compatible
- Source and Acceptor Handshake
- Complete Talker and Listener Functions with Extended Addressing
- Controller and System Controller Capability
- Service Request
- Remote and Local with Lockout
- Serial and Parallel Polling
- Device Clear
- Device Trigger
- Compatible with TMS 9911 DMA Controller
- Single +5 V Power Supply
- Interfaces directly to SN75160/1/2 Transceivers

DESCRIPTION

The TMS 9914 General Purpose Interface Bus Adapter is a microprocessor controlled versatile device which enables the designer to implement all of the functions or a subset described in the IEEE Std. 488-1975. Using this standard, a variety of instruments can be interconnected and remotely or automatically programmed and controlled. The TMS 9914 is fabricated with N-channel silicon-gate technology and is completely TTL compatible on all inputs and outputs including the power supply (+5 V). It needs a single phase clock (nominally 5 MHz) which may be independent of the microprocessor system clock and, therefore, it can easily be interfaced with most microprocessors. The general purpose interface bus adapter (GPIBA) performs the majority of the functions contained in IEEE STd. 488-1975 and is versatile enough to allow software implementation of those sections not directly implemented in hardware.



*GENERAL PURPOSE INTERFACE BUS

Figure 1. *Typical System Interconnect*

▸8

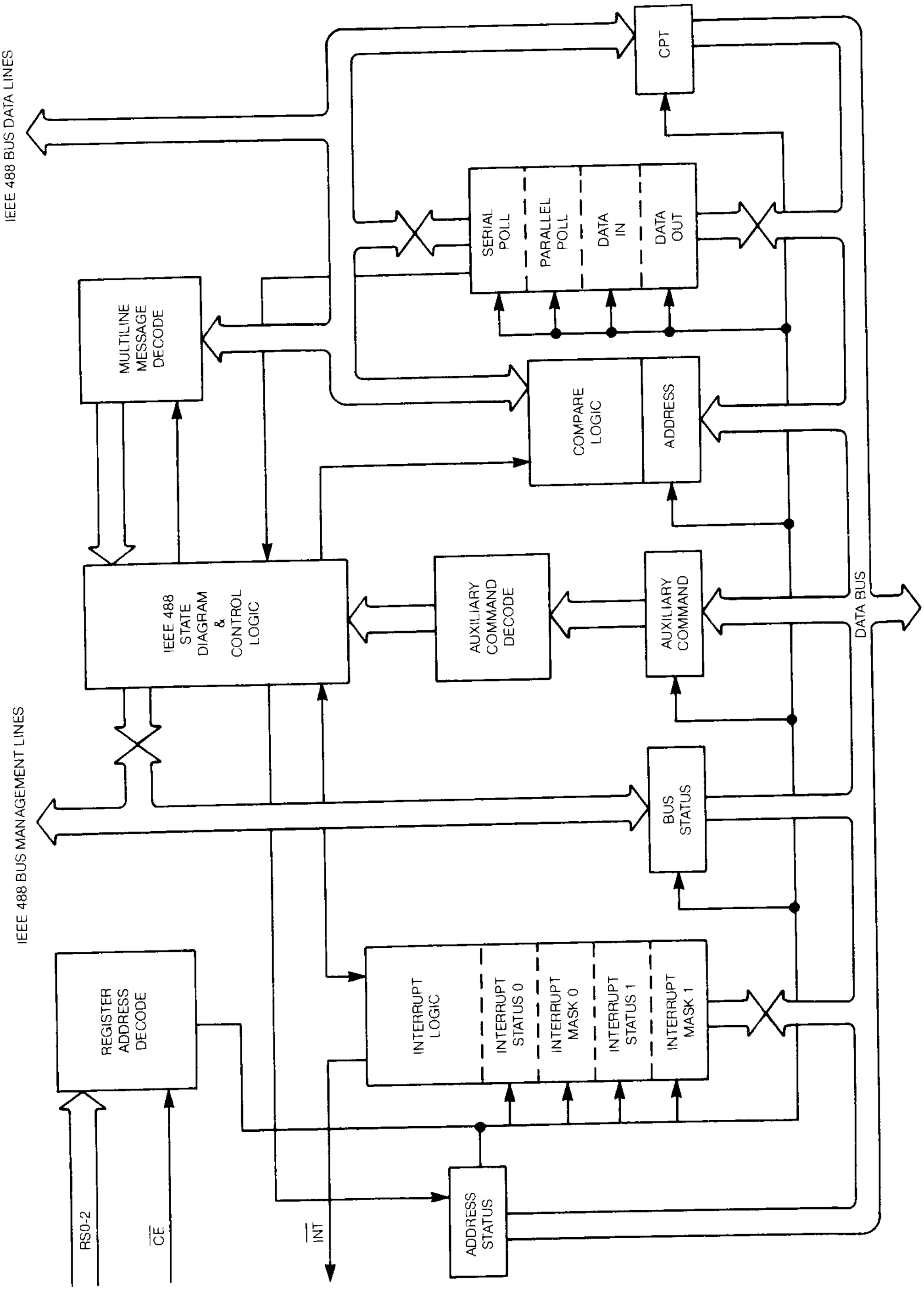


Figure 2. TMS 9914 Simplified Block Diagram

Table 1. Pin Description

Name	*I/O*	*Description*
DI01 through DI08	I/O	DATA I/O lines: allow data transfer between the TMS 9914 and the IEEE 488 data bus.
DAV	I/O	DATA VALID: Handshake Line. Sent by source device to indicate to acceptors that there is valid data on the IEEE bus data lines.
NRFD	I/O	NOT READY FOR DATA: Handshake Line. Sent by the acceptor to the source device to indicate when it is ready for a new byte of data.
NDAC	I/O	DATA NOT ACCEPTED: Handshake Line. Sent by acceptor to source device to indicate when it has accepted the current byte on the data bus.
ATN	I/O	ATTENTION: Management Line. Sent by the controller. When ATN is asserted, the information on the data lines is interpreted as commands, sent by the controller . . . When ATN is false, the data lines carry data.
IFC	I/O	INTERFACE CLEAR. Management Line. Sent by system controller to set the interface system, portions of which are contained in all interconnected devices in a known quiescent state. System controller assumes control. Open drain output with internal pullup.
REN	I/O	REMOTE ENABLE: Management Line. Sent by system controller and is used in conjunction with other messages to select between two alternate sources of programming data, e.g. via interface or front panel. Open drain output with internal pullup.
SRQ	I/O	SERVICE REQUEST: Management Line. Issued by a device on the bus to the controller to indicate a need for service.
EOI	I/O	END OR IDENTIFY: Management Line. If ATN is false, this signal is sent by the "talker" to indicate the end of a multiple byte transfer. If sent by the controller with ATN true, this will perform the parallel polling sequence.
$\overline{\text{CONTROLLER}}$	O	Bus transceiver control line. Indicates that the device is the controller.
TE	O	TALK ENABLE: Bus transceiver control line. Indicates the direction of data transfer on the data bus.
D0 through D7	I/O	Data I/O lines that allow transfer of data between TMS 9914 and the microprocessor.
RS0 through RS2	I	Address lines through which the TMS 9914 registers can be accessed by the microprocessor.
DBIN	I	When true (high) DBIN indicates to the TMS 9914 that the microprocessor is about to read from one of its registers. When false, that the microprocessor is about to write to one of its registers.
$\overline{\text{WE}}$	I	$\overline{\text{WRITE ENABLE}}$: indicates to the TMS 9914 that one of its registers is being written to.
$\overline{\text{CE}}$	I	$\overline{\text{CHIP ENABLE}}$: selects and enables the TMS 9914 for an microprocessor data transfer.
$\overline{\text{INT}}$	O	$\overline{\text{INT}}$: Open drain output. Sent to microprocessor to indicate the occurrence of an event on the bus requiring service.
$\overline{\text{ACCRQ}}$	O	$\overline{\text{ACCESS REQUEST}}$: Signal to TMS 9911 DMA controller requesting DMA.

PIN OUTS TO BE ASSIGNED

NOTE: The names of the IEEE bus lines have been maintained, and are therefore negative logic signals.

FUNCTIONAL DESCRIPTION

The TMS 9914 interfaces to the CPU with an eight-bit bidirectional data bus, three register select lines, two DMA control lines, reset and interrupt request lines, a DBIN and a $\overline{\text{WE}}$ line.

The internal architecture of the TMS 9914 is arranged into 13 registers, there being seven WRITE and six READ registers. Some are actually address ports through which current status can be obtained. Table 2 lists these registers and their addresses. The microporcessor accesses a TMS 9914 register by supplying the correct register address in conjunction with $\overline{\text{WE}}$ and DBIN. The $\overline{\text{CE}}$ is used to enable the address decode.

Table 2. TMS 9914 Registers and Addresses

NAME	TYPE	RS2	RS1	RS0	DBIN	$\overline{\text{WE}}$
INTERRUPT STATUS 0	R	0	0	0	1	1
INTERRUPT MASK 0	W	0	0	0	0	0
INTERRUPT STATUS 1	R	0	0	1	1	1
INTERRUPT MASK 1	W	0	0	1	0	0
ADDRESS STATUS	R	0	1	0	1	1
BUS STATUS	R	0	1	1	1	1
AUXILIARY COMMAND	W	0	1	1	0	0
ADDRESS SWITCH	R	1	0	0	1	1
ADDRESS	W	1	0	0	0	0
SERIAL POLL	W	1	0	1	0	0
COMMAND PASS THROUGH	R	1	1	0	1	1
PARALLEL POLL	W	1	1	0	0	0
DATA IN	R	1	1	1	1	1
DATA OUT	W	1	1	1	0	0

NOTE: The Address Switch register is external to the TMS 9914

In DMA operation the TMS 9911 supplies the memory address but not the peripheral device address (i.e., RS0-2, $\overline{\text{CE}}$) are not supplied). When the TMS 9914 sets $\overline{\text{ACCRQ}}$ low true, it is either because of a byte input or a byte output, and this will happen whether or not DMA transfer will take place. If in response to $\overline{\text{ACCRQ}}$ an $\overline{\text{ACCGR}}$ (access granted) is received, the $\overline{\text{ACCRQ}}$ will be reset and a DMA transfer will take place between the system memory and either the Data In or Data Out register. If the data transfer is with the microprocessor and if the microprocessor addresses either the Data In or Data Out register, the $\overline{\text{ACCRQ}}$ line will be reset. Note that in DMA mode the sense of DBIN is inverted.

Table 3 lists the commands which are directly handled by the TMS 9914, and those which require intervention by the microprocessor for their implementation.

Table 3. Remote Multiple Message Coding

		DIO8	DIO7	DIO6	DIO5	DIO4	DIO3	DIO2	DIO1		Note
Addressed Command Group	ACG	X	0	0	0	X	X	X	X	AC	
Device Clear	DCL	X	0	0	1	0	1	0	0	UC	
Group Execute Trigger	GET	X	0	0	0	1	0	0	0	AC	
Go To Local	GTL	X	0	0	0	0	0	0	1	AC	
Listen Address Group	LAG	X	0	1	X	X	X	X	X	AD	
Local Lock Out	LLO	X	0	0	1	0	0	0	1	UC	
My Listen Address	MLA	X	0	1	L	L	L	L	L	AD	1
My Talk Address	MTA	X	1	0	T	T	T	T	T	AD	2
My Secondary Address	MSA	X	1	1	S	S	S	S	S	SE	3, 4
Other Secondary Address	OSA									SE	4, 5
Other Talk Address	OTA				TAG • MTA					AD	
Primary Command Group	PCG									—	6
Parallel Poll Configure	PPC	X	0	0	0	0	1	0	1	AC	7
Parallel Poll Enable	PPE	X	1	1	0	S	P	P	P	SE	8, 9
Parallel Poll Disable	PPD	X	1	1	1	D	D	D	D	SE	8, 10
Parallel Poll Unconfigure	PPU	X	0	0	1	0	1	0	1	UC	11
Secondary Command Group	SCG	X	1	1	X	X	X	X	X	SE	
Selected Device Clear	SDC	X	0	0	0	0	1	0	0	AC	
Serial Poll Disable	SPD	X	0	0	1	1	0	0	1	UC	
Serial Poll Enable	SPE	X	0	0	1	1	0	0	0	UC	
Take Control	TCT	X	0	0	0	1	0	0	1	AC	12
Talk Address Group	TAG	X	1	0	X	X	X	X	X	AD	
Universal Command Group	UCG	X	0	0	1	X	X	X	X	UC	
Unlisten	UNL	X	0	1	1	1	1	1	1	AD	
Untalk	UNT	X	1	0	1	1	1	1	1	AD	

Symbols: AC — Addressed Command
AD — Address (Talk or Listen)
UC — Universal Command
SE — Secondary (Command or Address)
0 — Logical Zero (high level on IEEE Bus; Low level within 9914).
1 — Logical One (Low level on IEEE Bus; High level within 9914).
X — Don't Care (received message)
X — Must Not Drive (transmitted message)

Notes to Table 3:

1. L L L L L: Represents the coding for the device listen address.
2. T T T T T: Represents the coding for the device talk address.
3. S S S S S: Represents the coding for the device secondary address.
4. Secondary addresses will be handled via address pass through.
5. OSA will be handled as an invalid secondary address pass through by the MPU.
6. PCG = ACG v UCG v LAG v TAG
7. PPC will be handled in software by the MPU via Unrecognized Address Command Group pass through.
8. PPE, PPD will be handled via pass through next secondary feature.
9. S P P P represents the sense and bit for remote configurable parallel poll.
10. D D D D specify don't care bits that must be sent all zeroes, but need not be decoded by receiving device.
11. PPU is handled via Unrecognized Universal Command Group pass through.
12. TCT will be handled via Unrecognized Addressed Command Group pass through. However, in this case, the device must be in TADS before the pass through will occur.

Interrupt Status Registers 0 and 1

INT0	INT1	BI	BO	END	SPAS	RLC	MAC
GET	UUCG	UACG	APT	DCAS	MA	SRQ	IFC

INT0	An interrupt occurred in register 0
INT1	An interrupt occurred in register 1
BI	A byte has been received
BO	A byte has been output
END	An EOI occurred with ATN false
SPAS	Serial Poll Active State has occurred with rsv set in the Serial Poll register
RLC	A REMOTE/LOCAL change has occurred
MAC	An address change has occurred

GET	A Group Execute Trigger has occurred
UUCG	An Undefined Universal Command has been received
UACG	An Undefined Addressed Command has been received. This bit will also be set on receipt of a secondary command when the pts feature in the Auxiliary Command register is utilized.
APT	A secondary address has occurred
DCAS	Device Clear Active State has occurred
MA	My Address (MLAVMTA)•$\overline{\text{SPSM}}$
SRQ	A Service Request has been received
IFC	An IFC has been received

INT0 is the logical OR of each bit of Interrupt Status Register 0 ANDed with the respective bit of Interrupt Mask Register 0. INT1 is the same but applies to Interrupt Mask and Status Register 1. Reading either Interrupt Status Register will also clear it. The INT line will be cleared only when the interrupt status register which caused the interrupt is read.

Interrupt Mask Registers 0 and 1

☒	☒	BI	BO	END	IFC	RLC	MAC
GET	UUCG	UACG	APT	DCAS	MA	SRQ	SPAS

The Interrupt Mask Registers 0 and 1 correspond to the Interrupt Status Registers 0 and 1 respectively, with the exception of INT0 and INT1.

Address Status Register

REM	LLO	ATN	LPAS	TPAS	LADS V LACS	TADS V TACS	ulpa

The Address Status Register is used to convey the addressed state of the talker/listener and the remote/local and local lockout condition. This information is derived from the TMS 9914 internal logic states at the time of reading. The ulpa bit is used for dual addressing and indicates the state of the LSB of the bus at last primary addressed time.

Bus Status Register

ATN	DAV	NDAC	NRFD	EOI	SRQ	IFC	REN

The Bus Status Register allows the microprocessor to obtain the current status of the IEEE 488 Bus Management Lines.

Auxiliary Command Register

C/S	☒	☒	f4	f3	f2	f1	f0

The Auxiliary Command Register allows control of additional features on chip and provides a means of inputting some of the local messages to the interface functions. Table 4 lists these messages and commands. If C/S = 1, the feature will be set and if C/S = 0, the feature will be cleared. If C/S = NA, it should be sent as zero.

Table 4. Auxiliary Commands

Function	*Mnemonic*	*C/S*	*f4*	*f3*	*f2*	*f1*	*f0*
Chip Reset	rst	0/1	0	0	0	0	0
Release ACDS holdoff	dacr	0/1	0	0	0	0	1
Release RFD holdoff	rhfd	NA	0	0	0	1	0
Holdoff on all data	hdfa	0/1	0	0	0	1	1
Holdoff on EOI only	hdfe	0/1	0	0	1	0	0
Set new byte available false	nbaf	NA	0	0	1	0	1
Force group execute trigger	fget	0/1	0	0	1	1	0
Return to local	rtl	0/1	0	0	1	1	1
Return to local immediate	rtli	0	0	0	1	1	1
Send EOI with next byte	feoi	NA	0	1	0	0	0
Listen only	lon	0/1	0	1	0	0	1
Talk only	ton	0/1	0	1	0	1	0
Take control synchronously	tcs	NA	0	1	1	0	1
Take control asynchronously	tca	NA	0	1	1	0	0
Go to standby	gts	NA	0	1	0	1	1
Request parallel poll	rpp	0/1	0	1	1	1	0
Send interface clear	sic	0/1	0	1	1	1	1
Send remote enable	sre	0/1	1	0	0	0	0
Request contol	rqc	NA	1	0	0	0	1
Release control	rlc	NA	1	0	0	1	0
Disable all interrupts	dai	0/1	1	0	0	1	1
Pass through next secondary	pts	NA	1	0	1	0	0
Set T1 delay	stdl	0/1	1	0	1	0	1

Address Register

edpa	dal	dat	A5	A4	A3	A2	A1

edpa	enable dual primary addressing	dat	disable the talk function
dal	disable the listen function	A1 – A5	primary device address

The Address Switch Register corresponds to the Address Register. A power-up RESET or a rst command with C/S = 1 will leave the chip in a totally idle state. At this point, the Address Switch Register is read and the value is written into the Address Register. The reset condition is then cleared by sending rst with C/S = 0.

Serial Poll Register

S8	rsv	S6	S5	S4	S3	S2	S1

The Serial Poll register is used to establish the status byte that is sent out when the controller conducts a serial poll. Bits 1 through 6 and 8 contain status information, while bit 7, rsv, is used to enable the SRQ line and to indicate to the controller which device(s) was responsible for making a service request.

Command Pass Through Register

DIO8	DIO7	DIO6	DIO5	DIO4	DIO3	DIO2	DIO1

The Command Pass Through Register is used to pass through to the microprocessor any commands or secondary addresses that are not automatically handled in the TMS 9914.

Parallel Poll Register

PP8	PP7	PP6	PP5	PP4	PP3	PP2	PP1

This register contains the status bit that is output when the controller conducts a parallel poll.

Data-In Register

DIO8	DIO7	DIO6	DIO5	DIO4	DIO3	DIO2	DIO1

The data-in register is used to move data from the interface bus when the chip is addressed as a listener. Upon receipt of a data byte, the chip will hold NRFD true until the microprocessor reads the data-in register, when NRFD will be set false automatically.

Data-Out Register

DIO8	DIO7	DIO6	DIO5	DIO4	DIO3	DIO2	DIO1

The data-out register is used to move data from the TMS 9914 onto the IEEE std 488-1975 data bus.

After sending a byte out on the bus, the device can take part in a new handshake only after a new byte is placed in the data-out register, when it will be able to send DAV true again.