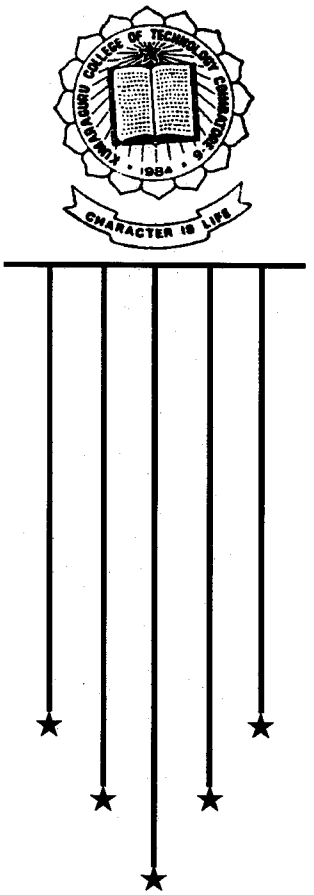
MICROCONTROLLER BASED MULTIDATA TESTER FOR SINGLE YARN TESTER

PROJECT REPORT



1999 -2000

SUBMITTED BY

A. CHANDRASUDESH

S. SIVAKUMAR

A. RAHAMATHULLAH

G. MANOJ

GUIDED BY

Prof. Mr. M. RAMASAMY, M.E., MISTE., Ph.D.

Head of the ECE Department

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE OF

BACHELOR OF ENGINEERING IN

ELECTRONICS & COMMUNICATION ENGINEERING

OF THE BHARATHIAR UNIVERSITY, COIMBATORE.

Department of Electronics & Communication Engineering

Kumaraguru College of Technology

Coimbatore - 641 006

Kumaraguru College of Technology

Department of Electronics and Communication Engineering

Coimbatore - 641 006.

CERTIFICATE

This is to certify that the contents of the project report entitled

'MICROCONTROLLER BASED MULTIDATA TESTER FOR SINGLE YARN TESTER'

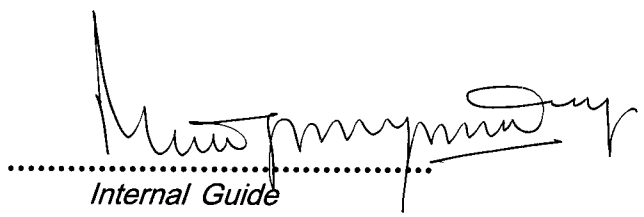
has been submitted by Mr / Ms .....

In Partial Fulfilment of the Requirements for the award of the Degree of Bachelor of Engineering

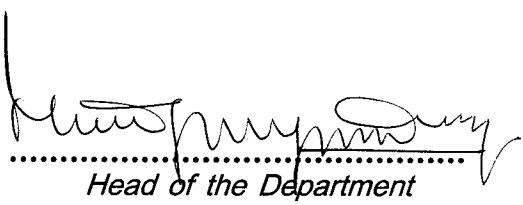
IN ELECTRONICS AND COMMUNICATION ENGINEERING

Branch of the Bharathiar University, Coimbatore.

During the academic year 1999 - 2000



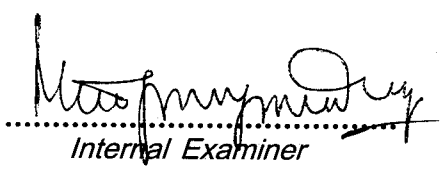
Internal Guide



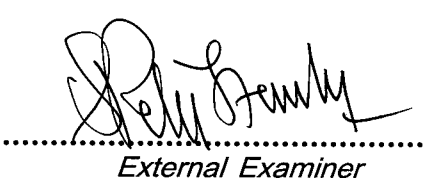
Head of the Department

Certified that the canditate with University Register Number .....

was examined by us in project Viva - Voce Examination held on.....



Internal Examiner



External Examiner

PREMIER

CERTIFICATE

This is to certify that the following students of **KUMARAGURU COLLEGE OF TECHNOLOGY**, Coimbatore, of Electronics and Communication Engineering department

1. CHANDRASUDESH A.
2. SIVAKUMAR S.
3. RAHAMATHULLAH .A
4. MANOJ G.

had undertaken their project entitled " Micro Controller Based MultiData Tester " (For Single Yarn Tester) from November '99 to March 2000 at our factory and have successfully completed it.

Their performance during the period was found to be good. We wish them all success.

V.HARIHARAKRISHNAN
Technologist – R&D
Premier Polytronics Limited,
304, TRICHY ROAD,
SINGANALLUR (POST),
COIMBATORE - 641 005.

Place : Coimbatore

Date : 06.03.2000

ACKNOWLEDGEMENT

We express our sincere thanks to our beloved Principal, Mr. K. Padmanaban B.Sc. (Engg), M. Tech., Ph.D., for all the facilities provided in carrying out this project work.

We express our gratitude and indebtedness to our respected guide Prof. M. Ramasamy M.E., MIEEE, MISTE, Ph.D., Head of the department of Electronics and Communication Engineering for his full fledged technical guidance, constant encouragement and suggestions in carrying out this project.

We also take this opportunity to thank our Assistant Professors Mr. K. Ramprakash, M.E., and Mr. S. Govindaraju M.E., for sharing us their indepth knowledge on the subject and providing us with tips as and when required.

We also like to thank from our bottom of our hearts Mr. V .Hariharakrishnan, and Mr. Socrates of Premier Polytronics Limited for being with us throughout our project.

We would also like to thank all the teaching and non-teaching staff for guiding us throughout, without which our project would not have been a great success.

SYNOPSIS

This project deals with the testing of various critical ICs used in the Single Yarn Testing equipment.

The project is based on the 8032 Microcontroller Conventional IC testers do not specify the fault present in defective chips simply display whether the IC is good or bad. Moreover these testing equipment are very expensive too.

In this project the various functions of the ICs such as the input/output ports, the timer/counter, the interrupts can be checked. This project is used to test the IC8032, IC8155, ADC574, DAC703, DS1230 for the above mentioned functions. Assembler ASM 51 is used for writing the software required for the testing system. Front-end software is written using 'C' language.

The test jig is interfaced to a terminal using the RS232C serial interface standard. Desired parameters can be tested by sending appropriate commands from the terminal. A set of protocols is framed

for serial communication between the terminal and microcontroller. The results of testing are displayed in the terminal.

The project was successfully completed and was tested using different critical IC's of testing equipment.

This project is very useful in ensuring the quality of the ICs used in the Single yarn testing equipment.

CONTENTS

1. INTRODUCTION

Basically this project deals with the testing of the Integrated Circuits that are used in the Single yarn testing equipment. Let us see some of the features of the ICs.

1.1. INTEGRATED CIRCUITS:

Electronic circuitry has undergone tremendous changes since the invention of a triode by Lee De Forest in 1907. In those days ,the active components (like triode) and passive components (like resistors, inductors and capacitors) of the circuits were separate and distinct units connected by soldered leads.

In the early 1960s, there was a need for reducing the size of electronic components used in Military applications. This drive for extreme reduction in size has led to the development of microelectronic circuits called Integrated Circuits (ICs) which are so small that their actual construction is done by technicians using microscopes.

An IC is a complex electronic circuit in which both the active and passive components are fabricated on an extremely single tiny chip of Silicon.

1.1.1 ADVANTAGES OF ICS:

- Extremely small physical size.
- Very small weight.
- Reduced cost.
- Extremely high reliability
- Suitability for small signal operation.
- Low power consumption.
- Easy replacement.

1.1.2 DISADVANTAGES OF ICs:

Though there are numerous advantages, the ICs also suffer from a few drawbacks. They are listed below,

1. Coils or Inductors cannot be fabricated.

3. They can handle only limited amount of power.
4. They are quite delicate and cannot withstand rough handling or excessive heat.

1.2 SINGLE YARN TESTER:

The single yarn tester is an automatic tensile tester manufactured by Premier Polytronics Limited, Coimbatore. In India, it is the sole manufacturer of this kind of products. Similar equipments are made only in three other places in the world.

This equipment is used to test the critical parameters of yarn, lea, fabric such as the tensile strength, elasticity etc..,

1.3 NEED FOR THIS PROJECT:

Complex electronic circuits are used in the yarn testing equipment. Microcontroller 8032 is generally used as the central processing unit of these equipments. In addition to this, various other expensive ICs form a part of this circuit. These ICs may get damaged due to accidental high voltages or due to rough handling and excessive heat etc.., So it is necessary to ensure the quality of these ICs before being used in the equipments..

Conventional IC testers available are very expensive. Moreover the functions of the parameters such as the ports, timers, interrupts etc., cannot be analysed in these testers.

Hence there arises a need for an interactive testing system which can analyze the different parameters of the ICs. This requirement is taken care of by this project.

2. MICROCONTROLLER 8032

2.1 ABOUT THE MICROCONTROLLER:

The microcontroller 8032 is widely used in the electronic industries. Since the microcontroller controls all the operations it is essential to discuss about its architecture.

The special function registers (SFR) are described below:

ACCUMULATOR:

ACC is the accumulator register. The mnemonics for accumulator-specific instructions, however, refer to the accumulator simply as A. The accumulator is one of the important special register.

B REGISTER:

The B register is used during multiply and divide operation. For other instructions, it can be treated as another scratch pad register.

PROGRAM STATUS WORD:

The program status word register contains program status information as detailed in the figure.

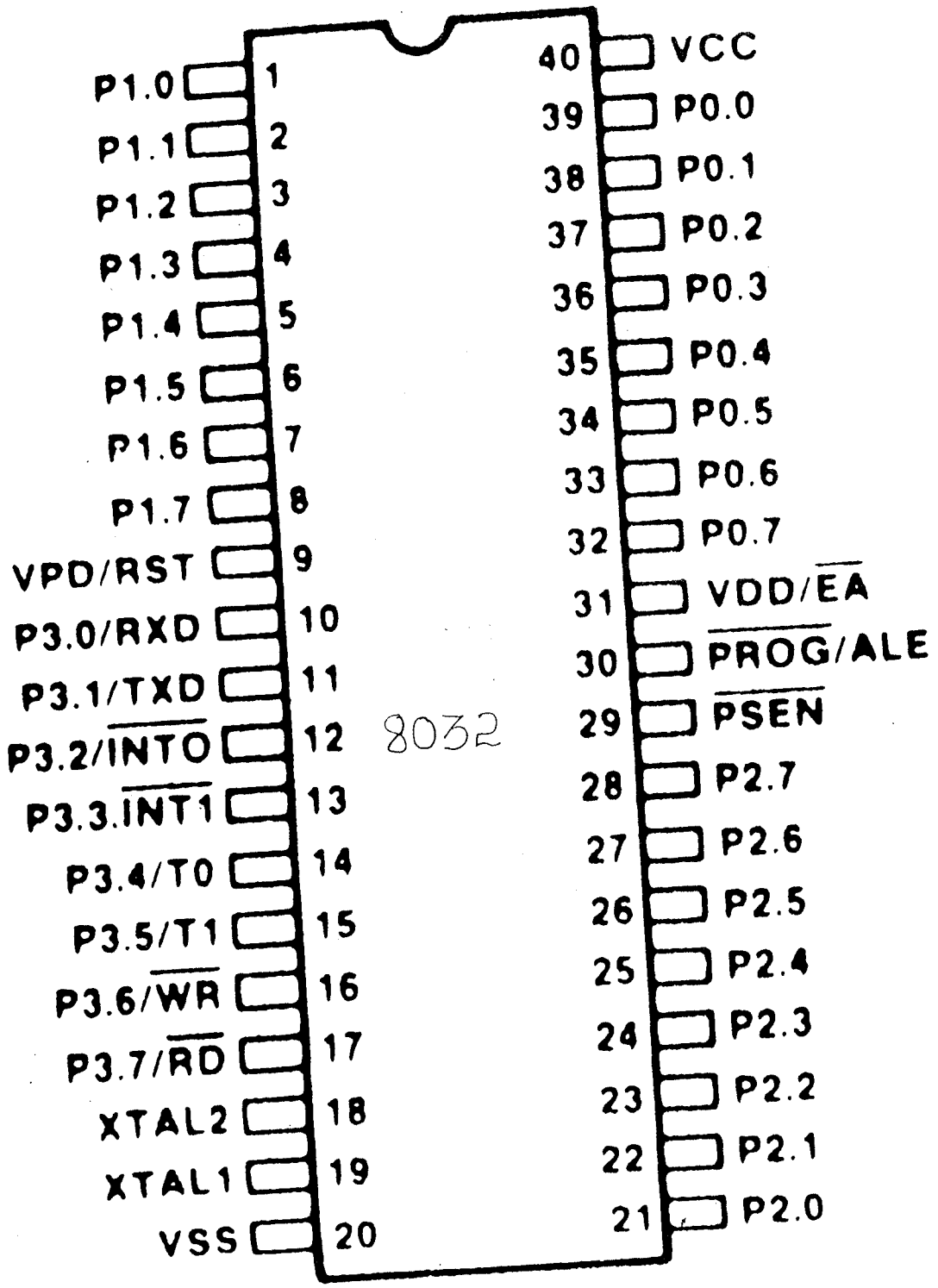
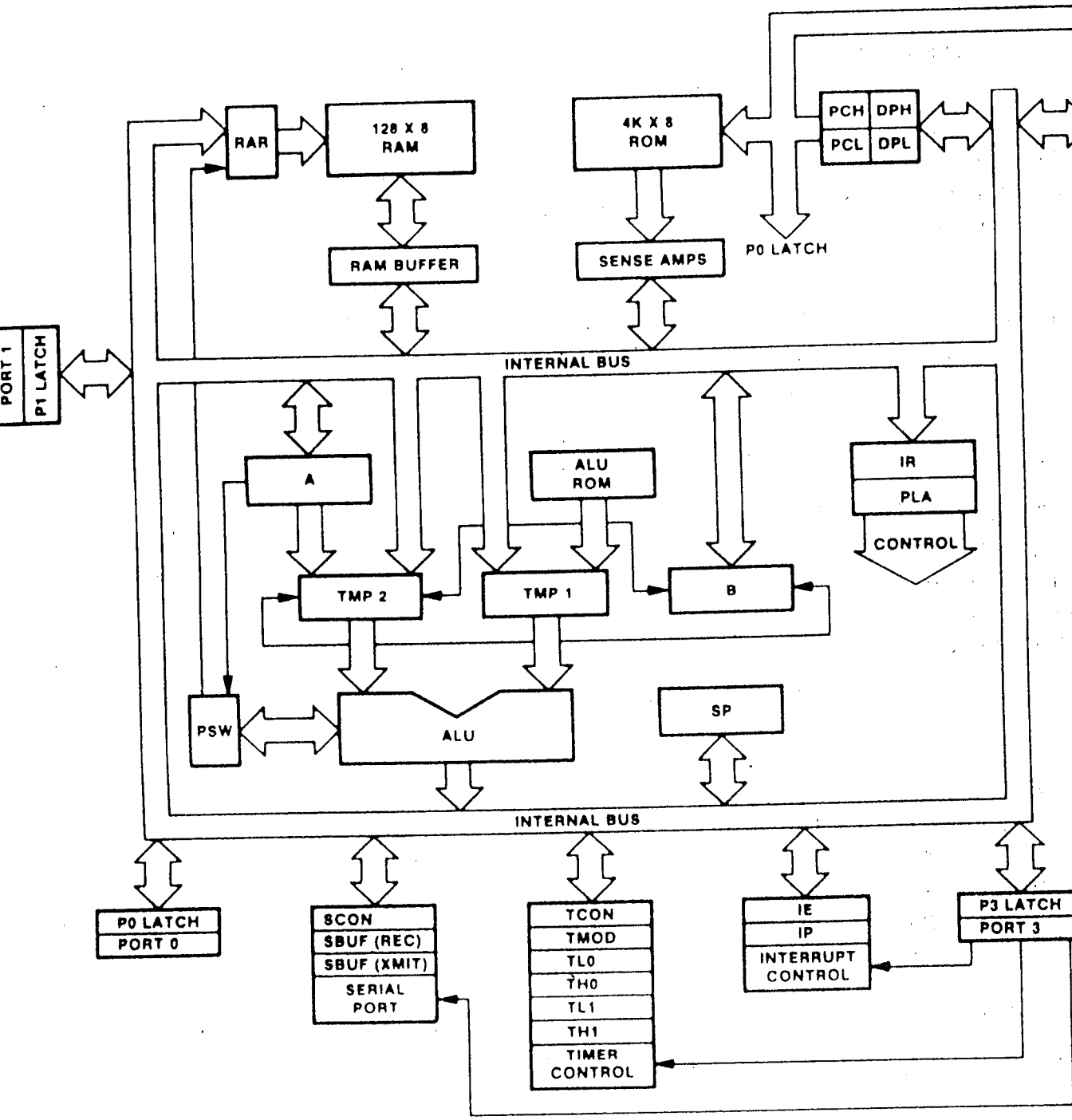


Fig. 2.1 8032 PIN DIAGRAM


RAR
128 X 8
RAM
RAM BUFFER
4K X 8
ROM
SENSE AMPS
P0 LATCH
PCH
DPH
PCL
DPL
PORT 1
P1 LATCH
INTERNAL BUS
A
ALU
ROM
IR
PLA
CONTROL
TMP 2
TMP 1
B
PSW
ALU
SP
INTERNAL BUS
P0 LATCH
PORT 0
SCON
SBUF (REC)
SBUF (XMIT)
SERIAL
PORT
TCON
TMOD
TL0
TH0
TL1
TH1
TIMER
CONTROL
IE
IP
INTERRUPT
CONTROL
P3 LATCH
PORT 3

STACK POINTER:

The stack pointer register is 8 bits wide. It is incremented before data is stored during PUSH and CALL executions. While the stack may reside anywhere in on chip RAM, the stack pointer is initialized to 07H after a reset. This causes the stack to begin at location 08H.

DATA POINTER:

The data pointer (DPTR) consists of a high byte (DPH) and a low byte (DPL). Its intended function is to hold a 16-bit address. It may be manipulated as a 16- bit register or as two independent 8-bit registers.

PORTS 0 TO 3:

P0, P1, P2 and P3 are the special function registers (SFR) latches of ports 0,1 2, 3 respectively.

SERIAL DATA BUFFER:

The serial data buffer is actually two separate registers, a transmit buffer and a receive buffer register. When data is moved to SBUF , it goes to the transmit buffer where it is a held for serial transmission. When data is moved from SBUF, it comes from the receiver.

TIMER REGISTERS:

Register pairs (TH0, TL0), (TH1, TL1), and (TH2, TL2) are the 16-bit counting registers for timer/ counter 0, 1 or 2 respectively.

CAPTURE REGISTERS:

The register pair (RCAP2H, RCAP2L) are the capture registers for the timer2 "capture mode". In this mode, in response to a transition at the 8032's T2EX pin, TH2 AND TL2 are copied into RCAP2H and RCAP2L. Timer 2 also has a 16-bit auto reload mode.

CONTROL REGISTERS:

Special function registers IP, IE, TMOD, TCON, T2CON, SCON and PCON contents control and status bits for the interrupt system, the timer /counter, and the serial ports.

TIMER /COUNTER:

The 8032 has two 16-bit timer/counter register. Timer0 and Timer1. The 8032 has these two plus one more timer2. All the three can be configured to operate either as timers or counters. In the timer function, the register is incremented every machine cycle.

In addition to "TIMER" or " COUNTER" selection. Timer0 and Timer1 have 4 operating modes . Timer 2, In the 8032, has 3 modes of operation: "CAPTURE", AUTO RELOAD"and "BAUD RATE GENERATOR".

TIMER 0 AND TIMER 1:

These timer/ counter are present in the 8032. The "timer"or "counter" function is selected by control bits C/ T in the special function register TMOD .These two Timer/ Counter have four operating modes , which are selected by bit pairs(M1,M0)in TMOD. Modes 0, 1 and 2 are the same for both Timer/ Counters.

SERIAL INTERFACE:

The serial port is full duplex, meaning it can transmit and receive simultaneously. It is also receive buffered, meaning that it can commence reception of a second byte before a previously received byte has been read from the receive register. The serial port receive and transmit registers are both accessed at special function register SBUF. Writing to SBUF loads the transmit register, and reading SBUF accesses a physically separate receive register. The serial port can operate in 4 modes.

MODE 0:

Serial data enters and exists through RXD, TXD outputs this shift, clock, 8 bits are transmitted / received. The baud rate is fixed at 1/12 the oscillator frequency.

MODE 1:

10 bits are transmitted or received :a start bit (0), 8 data bits (LSB first) and a stop bit (1). On receive, the stop bit goes into RB8 in special function register SCON. The baud rate is variable.

MODE 2:

11 bits are transmitted or received ; a start bit (0), 8 data bits(LSB first), a programmable 9^{th} bit and a stop bit (1). On transmission, the 9^{th} data bit (TB8 in SCON) can be assigned the value of 0 or 1. The baud rate is programmable to either 1/32 or 1/64 the oscillator frequency.

MODE 3:

11 bits are transmitted or received: a start bit (0), in fact mode 3 is the same as mode 2 in all respects except the baud rate. The baud rate in mode 3 is variable.

In all four modes, transmission is initiated by any instruction that uses SBUF as a destination register. Reception is initiated in mode 0 by the condition Ri = 0- and RED = 1. Reception is initiated explained in the other modes by the incoming start bit if REN =1.

INTERRUPTS:

The 8032 provides 6 interrupt sources. The external interrupts INT0 and INT1 can each be either level activated or transition activated, depending on bits IT0 and IT1 in register TCON. The flags that actually generate these interrupts are bits IE0 and IE1 in TCON. When an external interrupt is generated, the flags that generated it,is cleared by the hardware when the service routine is vectored to only if the interrupt was transition-activated. If the interrupt was level –activated, then the external requesting source controls the request flag, rather than the on-chip hardware.

The timer 0 and timer 1 interrupts are generated by TF0 and TF1, which are set by a rollover in their respective Timer/Counter registers. When a timer interrupt is generated by the on-chip hardware when the service routine is vectored to. The serial port interrupt is

generated by the logical OR of R1 and T1. Neither of these flags is cleared by hardware when the service routine is vectored to.

The 8032, the Timer2 interrupt generated by the logical OR of TF2 and EXF2. Neither of these flags is cleared by hardware when the service routine is vectored to.In fact, the service routine may have to determine whether it was TF2 or EXF2 that generated the interrupt and the bit will have to be cleared in software. All of the bits that generate the interrupts can be set or cleared by software,with the same result as though it had been cleared by hardware. That is, interrupts can be generated or pending interrupts can be cancelled in software. Each of these interrupt source can be individually enabled or disabled by setting or clearing bit in the special function register IE. IE contains also a global disable bit, EA which disable all interrupt at once.

2.2 PIN DESCRIPTION:

VCC: Supply voltage (+5v DC)

VSS: Circuit ground potential.

PORT 0: Port 0 is an 8-bit open drain bidirectional I/O port.

PORT 1:Port 1 is an 8-bit bidirectional I/O port with internal pullups.

PORT 2:Port 2 is an 8-bit bidirectional I/O port with internal pullups.

PORT 3: Port 3 is an 8-bit bidirectional I/O port with internal pullups.

RST: Reset input .

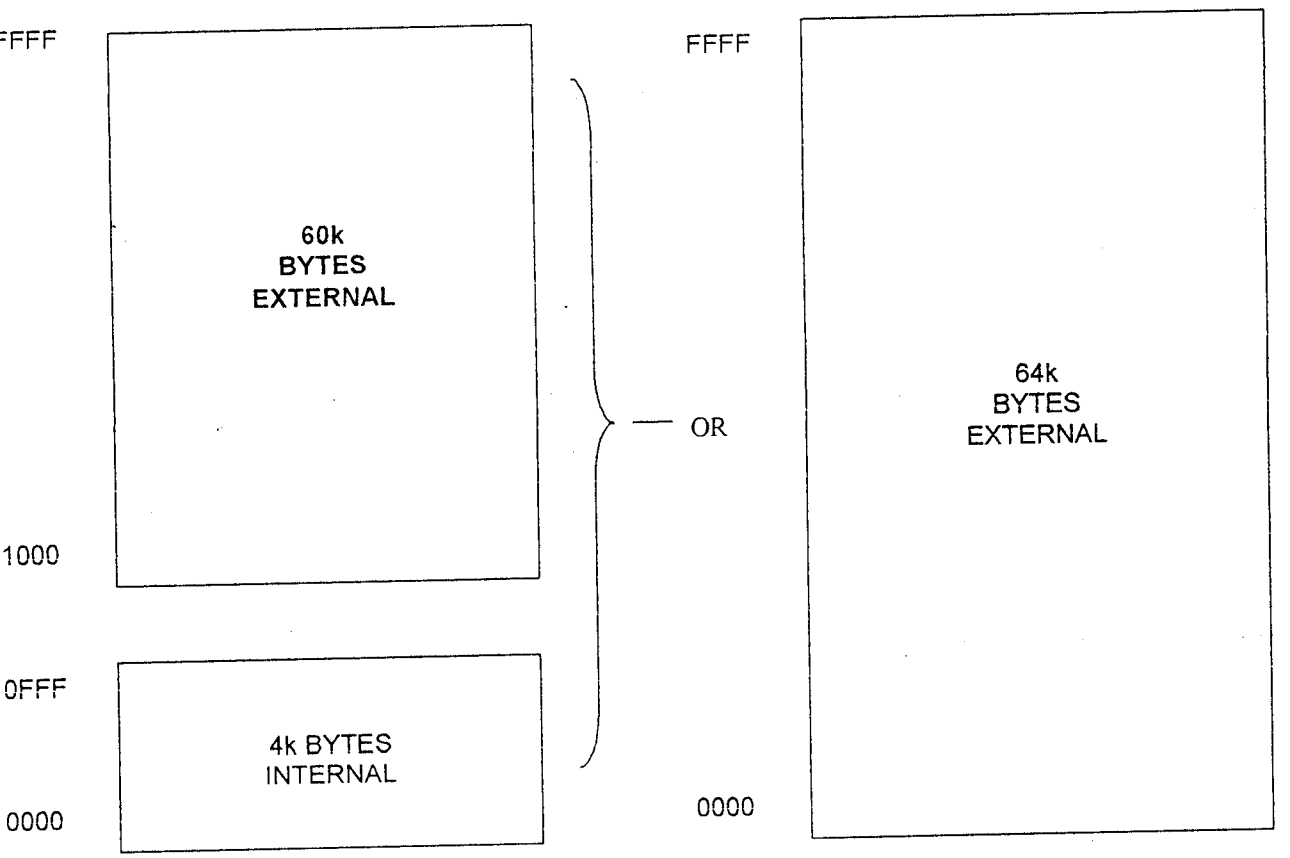
ALE/$\overline{\text{PROG}}$: Address latch enable/ programming.

$\overline{\text{PSEN}}$: Program store enable.

XTAL1:Input to the inverting oscillator amplifier.

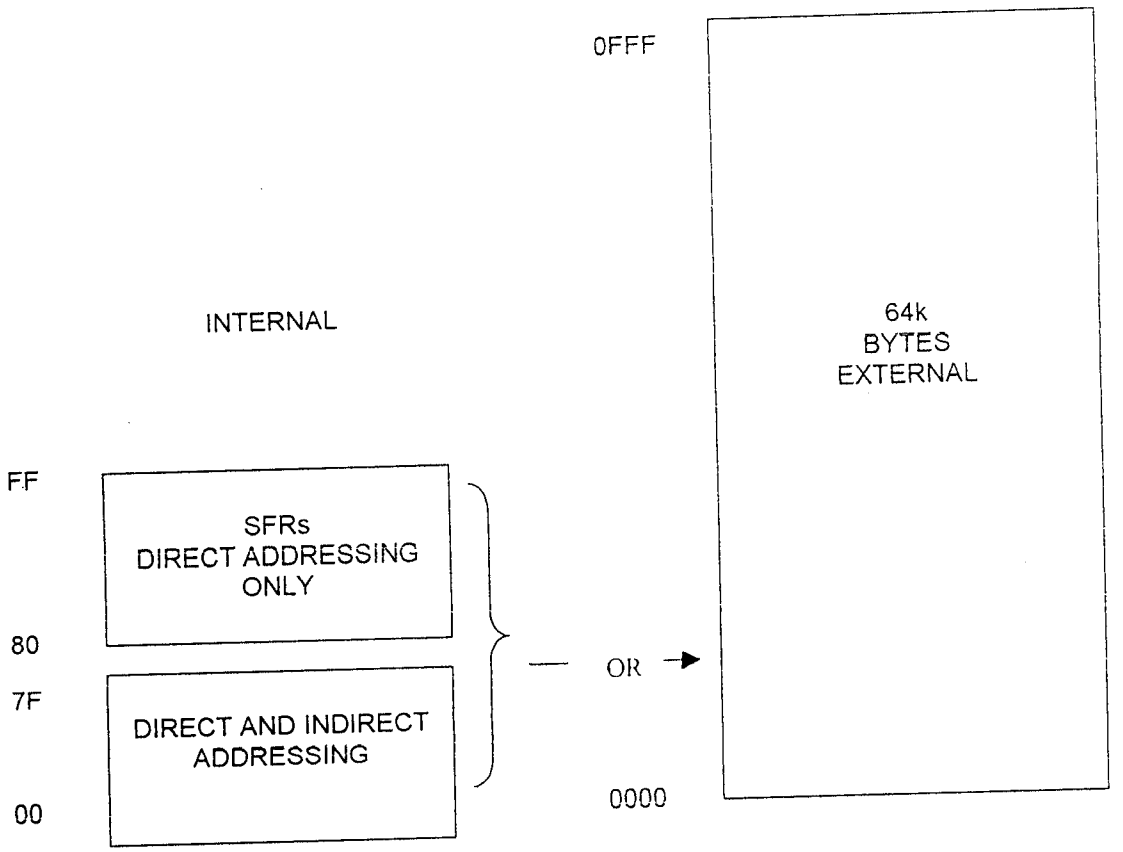
8032 MEMORY ORGANISATION

PROGRAM MEMORY :



The 8032 has separate address spaces for program and data memory.The program memory can be upto 64K bytes long.The lower 4K can reside on-chip.The figure above shows the map of the 8032 Program memory.The MOVX instruction is used to access the external memory.

DATA MEMORY :



The 8032 has 128 bytes of on-chip RAM,plus a number of Special Function Registers(SFRs).The lower 128 bytes of RAM which can be accessed by both direct and indirect addressing can be divided into three segments as,

Register Banks 0-3

Bit Addressable area

Scratch Pad Area.

Program Status Word (PSW) Register

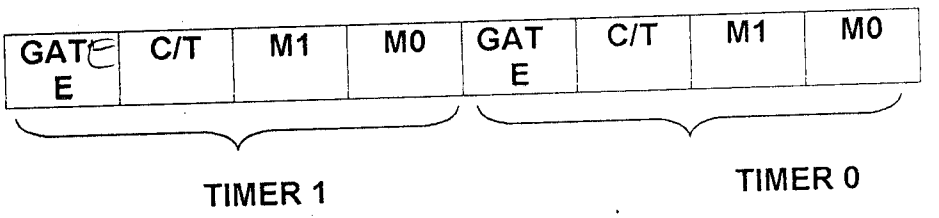
CY	AC	F0	RS1	RS0	OV	-	P

BIT	SYMBOL	FUNCTION
PSW.7	CY	Carry flag.
PSW.6	AC	Auxilliary flag. (For BSD operations.)
PSW.5	F0	Flag 0. (Available to the user for general purposes.)
PSW.4	RS1	Register bank select control bit 1. Set/cleared by software to determine working register bank.
PSW.3	RS0	Register bank select control bit 0. Set/cleared by software to determine working register bank.
PSW.2	OV	Overflow flag.
PSW.1	-	User-definable flag
PSW.0	P	Parity flag. Set/cleared by hardware each instruction cycle to indicate an odd/even number of "one" bits in the Accumulator, i.e., even parity.

NOTE : The contents of (RS1, RS0) enable the working register banks as follows:

(0,0) - Bank 0 (00H-07H)
(0,1) - Bank 1 (08H-0fH)
(1,0) - Bank 2 (10H-17H)
(1,1) - Bank 3 (18H-17H)

Timer / Counter Mode Control (TMOD) Register



GATE	Gating control when set. Timer/Counter "x" is enabled while "INTx" pin is high and "TRx" control pin is set. When cleared Timer "x" is enabled whenever "TRx" control bit is set.
C/T	Timer or Counter Selector cleared for Timer operation (input from internal system clock). Set for Counter operation (input from "Tx" input pin).

M1	M0	OPERATING
0	0	8048 Timer "TLx" serves as 5 bit prescaler.
0	1	16-bit Timer/Counter "THx" and "TLx" are cascaded; there is no prescaler.
1	0	8-bit auto-reload Timer/Counter "THx" holds a value which is to be reloaded into "TLx" each time it overflows.
1	1	(Timer 0) TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits. TH0 is an 8-bit timer only controlled by Timer 1 control bits.
1	1	(Timer 1) Timer/Counter 1 stopped.

Timer / Counter Control (TCON) Register

MSB LSB

TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

BIT	SYMBOL	FUNCTION
TCON.7	TF1	Timer 1 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine, or clearing the bit in software.
TCON.6	TR1	Timer 1 Run control bit. Set/cleared by software to run Timer/Counter on/off.
TCON.5	TF0	Timer 0 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine, or by clearing the bit in software.
TCON.4	TR0	Timer 0 Run control bit. Set/cleared by software to run Timer/Counter on/off.
TCON.3	IE1	Interrupt 1 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
TCON.2	IT1	Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.
TCON.1	IE0	Interrupt 0 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
TCON.0	IT0	Interrupt 0 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.

Serial Port Control (SCON) Register

SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Where SM0, SM1 specify the serial port mode, as follows:

SM0	SM1	Mode	Description	Baud Rate
0	0	0	Shift register	$f_{osc}/12$
0	1	1	8-bit UART	Variable
1	0	2	9-bit UART	$f_{osc}/64$ or $f_{osc}/32$
1	1	3	9-bit UART	variable

SM2 — Enables the multi processor communication feature in modes 2 and 3. In mode 2 or 3, if SM2 is set to 1, then RI will not be activated if the received 9th data bit (RB8) is 0. In mode1, if SM2 = 1 then RI will not be activated if a valid stop bit was not received. In mode 0, SM2 should be 0.

REN — Enables serial reception. Set by software to enable reception. Clear by software to disable reception.

TB8 — The 9th data bit that will be transmitted in modes 2 and 3. Set or cleared by software as desired.

RB8 — In modes 2 and 3, is the 9th data bit that was received. In mode 1, if SM2 = 0, RB8 is the stop bit that was received, in mode 0 RB 8 is not used.

TI — Transmit interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software.

RI — Receive interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or half way through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software.

Power Control (PCON) Register

MSB							LSB
SMOD	-	-	-	GF1	GF0	PD	IDL

BIT	SYMBOL	FUNCTION
PCON.7	SMOD	Double Baud rate bit. When set to 1 and Timer 1 is used to generate baud, and the Serial Port is used in modes 1,2 or 3.
PCON.6	-	Reserved
PCON.5	-	Reserved
PCON.4	-	Reserved
PCON.3	GF1	General-purpose flag bit
PCON.2	GF0	General-purpose flag bit
PCON.1	PD	Power-Down bit. Setting this bit activates power-down operation.
PCON.0	IDL	Idle mode bit. Setting this bit activate idle mode operation.

If 1s are written to PD and IDL at the same time, PD takes precedence. The reset value of PCON is (0XXX0000). In the NMOS devices, the PCON register only contains SMOD. The other four bits are implemented only in the CMOS devices. User software should never write 1s to unimplemented bits, since they may be used in future products.

3. DETAILS OF ICs TESTED

3.1 ADC 574

FEATURES:

- Complete 12 bit A/D converter with reference, clock, and 8-,12-,16-bit microprocessor bus interface
- Improved performance second source for 574A type A/D converter.
- Conversion time: 125 micro seconds maximum.
- Bus Access time: 150 micro seconds maximum.
- Dual in line plastic, PLCC and hermetic ceramic.
- Fully specified for operation on –12v to +12v or –15v to +15v supplies.
- No missing codes over temperature.

3.1.1 DESCRIPTION:

The ADC 574A is a 12 bit successive approximation analog to digital converter, utilizing state of the art CMOS and laser trimmed bipolar die custom designed for freedom from latch up and for optimum

AC performance if is complete with a self contained +10v reference internal clock, digital interface for microprocessor control and three state outputs.

The reference circuit, containing a buried Zener, is laser trimmed for minimum temperature coefficient. The clock oscillator is current controlled for excellent stability over temperature. Full scale and offset errors may be externally trimmed to zero.

Internal scaling resistors are provided for the selection of analog input signal ranges of 0v to +20v, 5v and 10 v. The converter may be externally programmed to provide 8- or 12- bit resolution. Output data are available in a parallel format from TTL- compatible three state output buffers.

3.2 *DAC703*

FEATURES:

- Dual- in -line plastic and hermetic ceramic and SOIC.
- Monotonic (At 15 bits) overfull specification temperature range .
- High accuracy.
- Output voltage and output current models.
- Pin compatible with DAC 70 ,71,72 families.

3.2.1 DESCRIPTION:

The DAC70x family comprises of complete 16 bit digital to analog converters that include a precision buried–zener voltage reference and a low noise all on one small monolithic chip. A combination of current–switch design techniques accomplishes not only 15-bit monotonicity over the entire specified temperature range, but also a maximum end point linearity error of +0.0015% or -0.0015% of full scale range.

Digital inputs are complementary binary coded

and are TTL-, LSTTL-, 54/74C- AND 54/74HC compatible over the entire temperature range outputs of 0 to +10v, +10v or –10v, 0 to –2mA and +1mA or –1mA are available.

These D/A converters are packaged in hermetic 24- pin ceramic side brazed are moulded plastic. The DIP packaged parts are pin compatible with the voltage and current output DAC71 and DAC72 model families. DAC703 is offered in a 24-pin SOIC package for surface mount applications.

3.3 PERIPHERAL IC 8155:

The 8155 is a multipurpose programmable device specifically designed to be compatible with the processor. The 8155 includes 256 bytes of R/W memory, three I/O ports and a timer. The programmable. I/O sections of these devices are illustrated in the following sections.

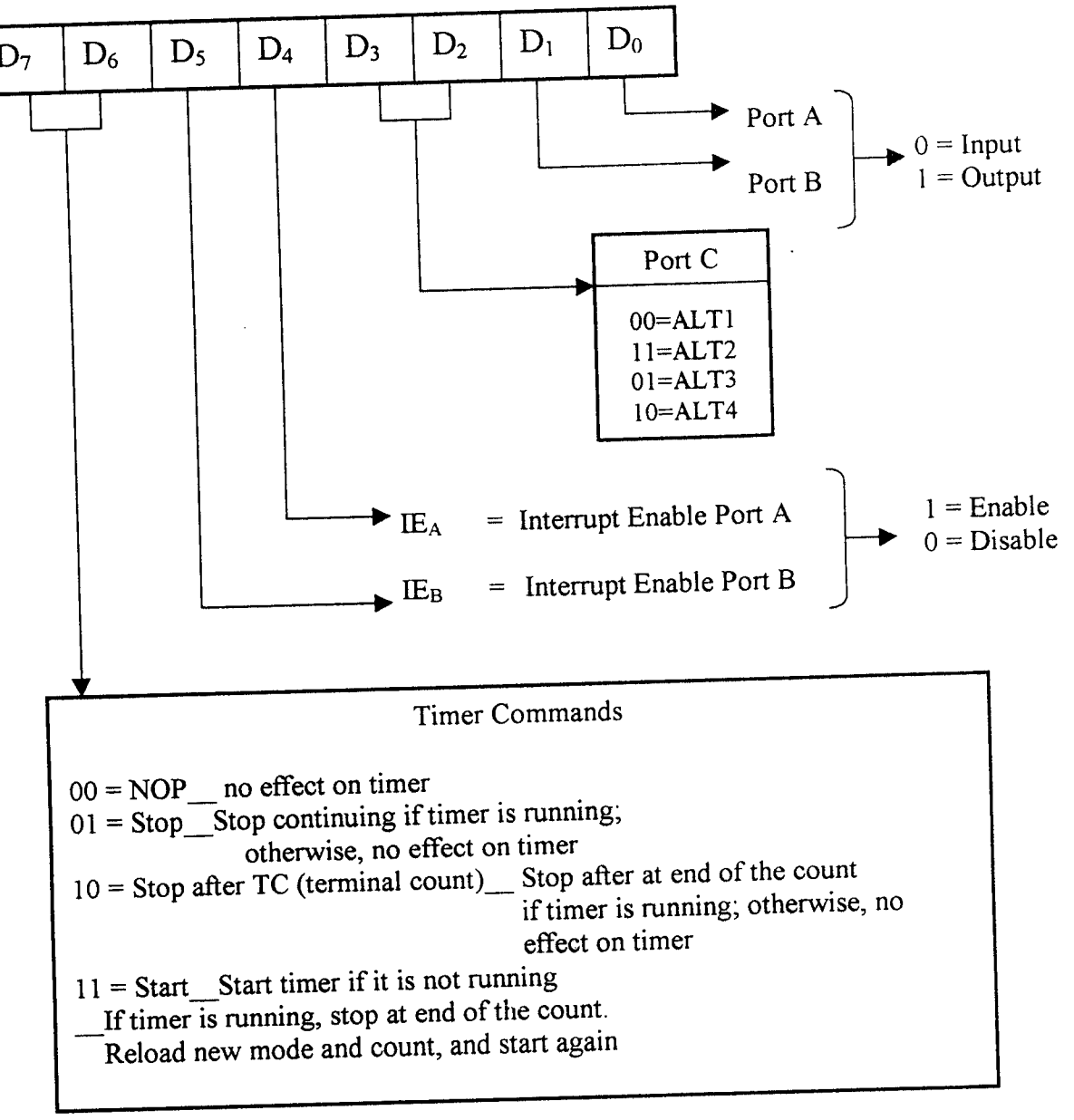
The 8155 has two sections, the first is 256 bytes of R/W memory, and the second is a programmable I/O. Functionally, these two sections can be viewed as two independent chips. The I/O

section includes two 8-bit parallel I/O ports (A&B), one 6 bit port(C) , and a timer. All the ports can be configured as input/ output ports. Ports A and B also can be programmed in the handshake mode, each port using three signals as handshake signals from portC. The timer is a 14- bit down-counter and has four modes.

3.3.1 THE 8155 I/O PORTS

The I/O section of the 8155 includes a control register, three ports, and two registers for the timer. The expanded block diagram of the I/O section represents a typical programmable I/O, two address lines plus the chip select logic were used to determine port addresses. The 8155 I/O section requires three address lines –AD2 TO AD0 and the chip enable logic to specify one of the seven registers.

To communicate with peripherals through the 8155, the following steps are necessary:



Control Word Definition for 8155

1. Determine the address based on the chip enable logic and address lines AD0-AD2.
2. Write a control word in the control register to specify i/o functions of the ports and the timer characteristics.
3. Write I/O instructions to port addresses to communicate with the peripherals.
4. Read the status register, if necessary, to verify the status of the I/O ports and the timer.In simple applications, this step is not necessary.

3.4 DS 1230(NVRAM)

***FEATURES*:**

- 10 Years minimum data retention in the absence of external power .
- Data is automatically protected during power loss.
- Replaces 32k*8 volatile static RAM,EEPROM or Flash memeory.
- Unlimited write cycles * low power CMOS.
- Read and write access times as fast as 70ns.
- Lithium energy source is electrically dieconnected to retain freshness until power is applied for the first time.
- Full +/- 10% Vcc operating range.

3.4.1 DESCRIPTION:

DS 1230 256k NVSRAM'S are 262,144-bit, fully static, non volatile SRAM'S organised as 32,768 words by 8 bits. Each NVSRAM has a self contained lithium energy source and control circuitry which constantly monitors Vcc for an out of tolerance condition. When such a condition occurs, the lithium energy source is automatically switched on and write protection is unconditionally enabled to prevent data corruption. There is no limit on the number of write cycles that can be executed and no additional support circuitry is required for microprocessor interfacing.

4. HARDWARE DESCRIPTION

4.1 SYSTEM DESIGN:

The overall block diagram of this project is represented in the figure 1.

The system design consists of the following blocks.

1) PC
2) Main Board
3) Testing Board
4) Power Supply

The various blocks listed above are explained as follows.

1) **PC**: A personal computer with a serial communication port, COM 1 / COM 2 is required for this project. Front end software written in 'C' language is stored in the PC. The PC is connected to the testing system using the RS232 C cable.
2) **Main Board:** This board is an integral port of the single yarn tester. It controls the various operations of the equipment. It has the microcontroller 8032 as its central processing unit with external program and data memory. The other ICs which are to

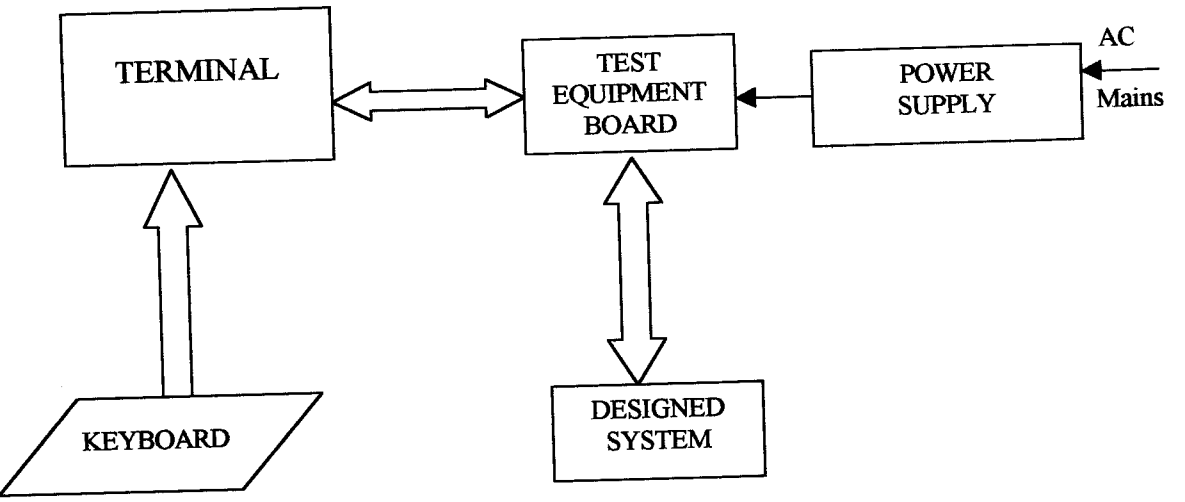


Fig. 1 OVERALL BLOCK DIAGRAM

be tested in this project are interfaced to the 8032 microcontroller. It also has the IC Max 232 for serial communication with the PC.

3) **Testing Board:** The testing board consists of comparators, DIP switches and LEDs which are used for conducting the various tests to be performed in this project.

4) **Power Supply:** The voltages required for this main board are +5V, +15V and -15V. These voltage are provided by a power supply unit.

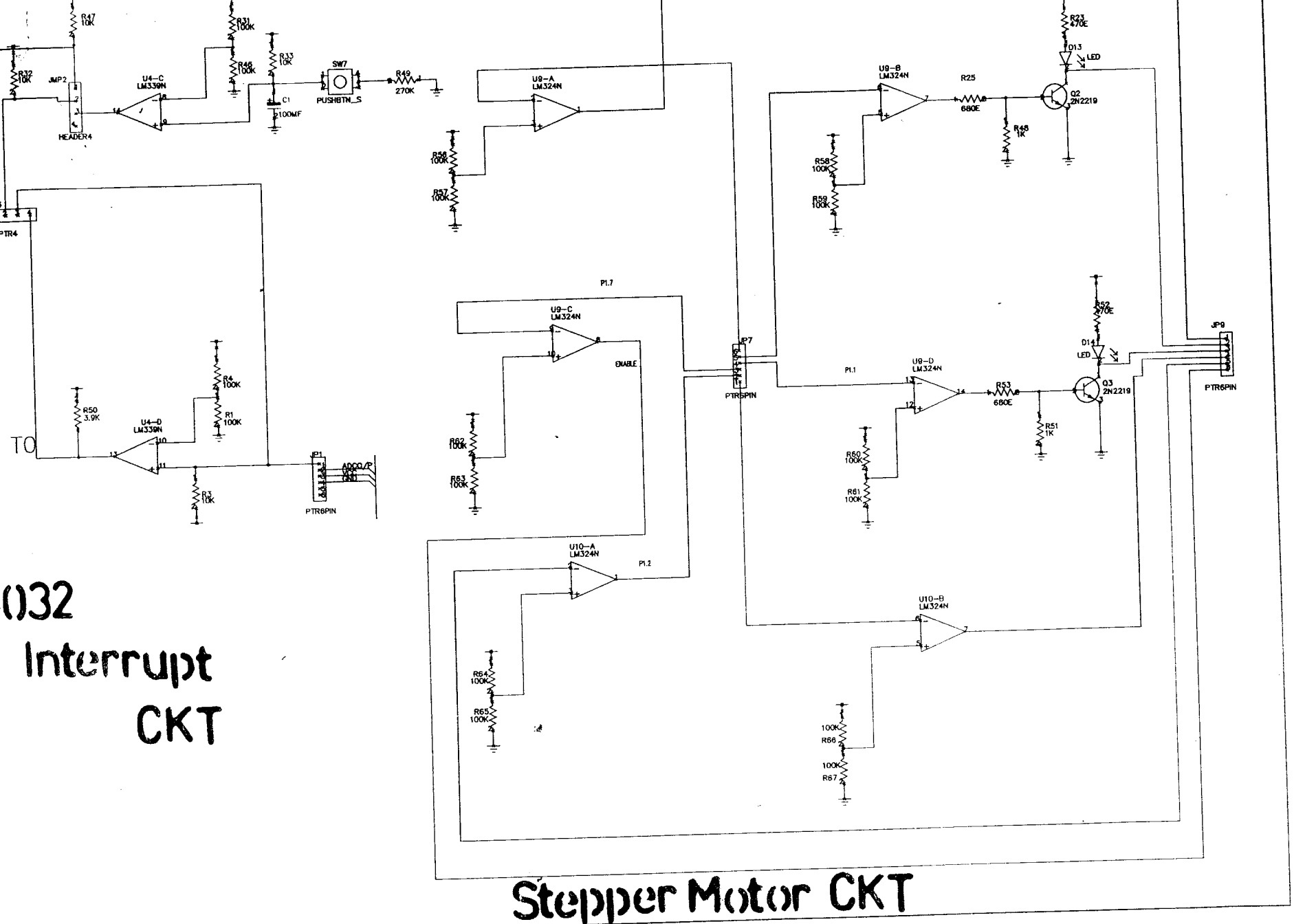
4.2 CIRCUIT DESCRIPTION

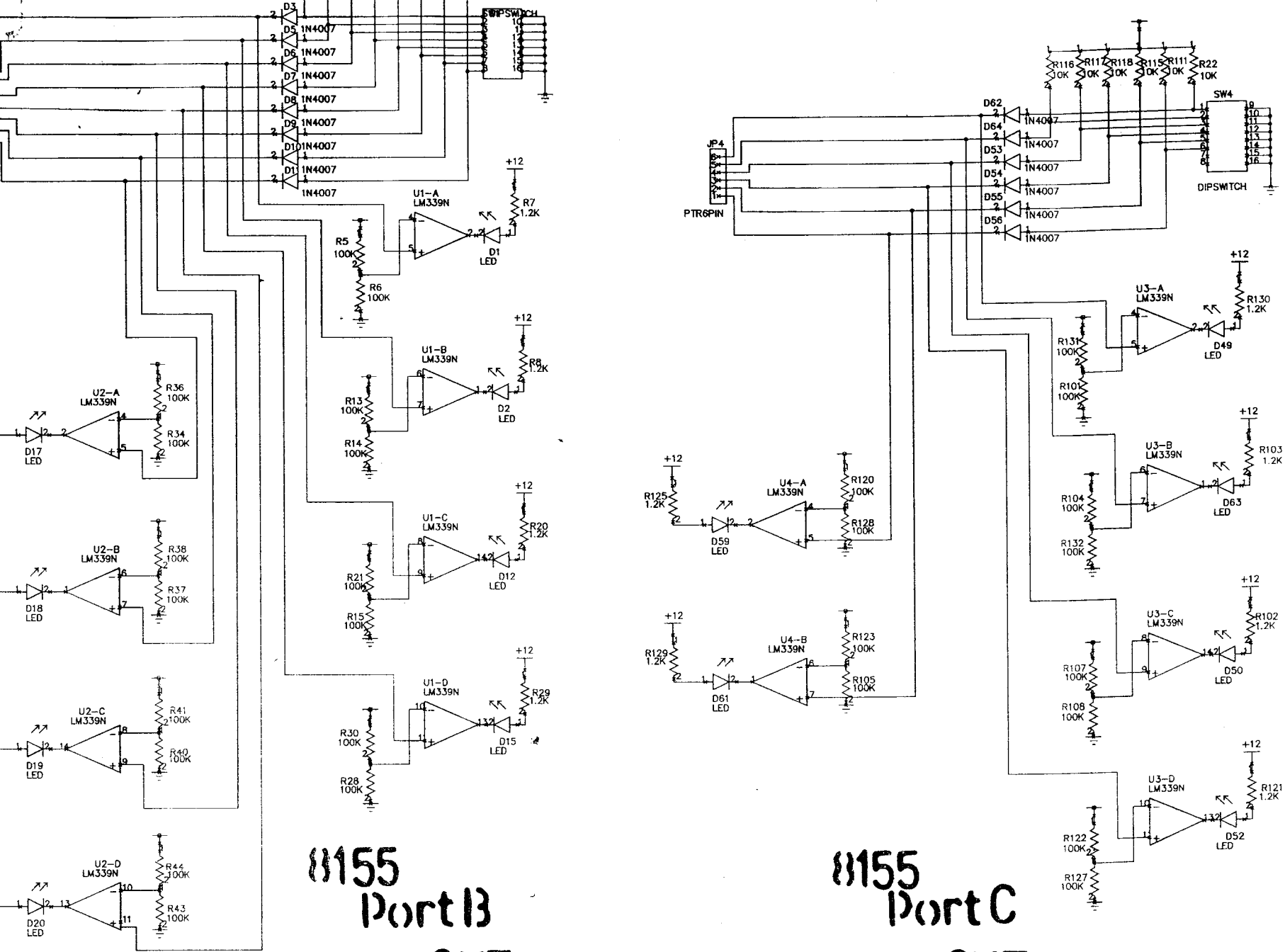
The circuit consists of 8 pin PTR connectors to which the port 1 of IC 8032, ports A, B and C of the IC 8155 are connected.

All the port lines are in turn connected to them inverting terminals of the comparators and separate DIP switches. Voltage dividers which provide an o/p voltage of 2.5 V are connected to the inverting input terminals of the comparator. The comparator circuit is constructed using the quad single supply comparator LM339. This IC is capable of level detection, low-level sensing of the input signals.

LEDs are provided at the outputs of the comparator. The circuit has a push button connected to a comparator switch to test the external interrupts of the IC 8032.

A separate 6 pin connector JMP2 is provided for power supply from the main board. The voltage levels required are,


R47 10K
R31 100K
R32 10K
JMP2
HEADER4
U4-C LM339N
R46 100K
R33 10K
C1 2100MF
SW7
PUSHBTN_S
R49 270K
U9-A LM324N
R58 100K
R57 100K
R23 470E
D13
LED
U9-B LM324N
R25
680E
Q2 2N2219
R48 1K
R58 100K
R59 100K
PTR4
P1.7
U9-C LM324N
ENABLE
JP7
PTR6PIN
P1.1
U9-D LM324N
R52 470E
D14
LED
JP9
PTR6PIN
R53 680E
Q3 2N2219
R51 1K
R62 100K
R63 100K
R60 100K
R61 100K
R50 3.9K
U4-D LM339N
R4 100K
R1 100K
R3 10K
JP1
PTR6PIN
U10-A LM324N
P1.2
U10-B LM324N
R64 100K
R65 100K
100K
R66
100K
R67
TO
0032
Interrupt
CKT
Stepper Motor CKT


D3 1N4007
D5 1N4007
D6 1N4007
D7 1N4007
D8 1N4007
D9 1N4007
D10 1N4007
D11 1N4007
1N4007
DIPSWITCH
+12
U1-A LM339N
R7 1.2K
R5 100K
R6 100K
D1 LED
U1-B LM339N
R8 1.2K
R13 100K
R14 100K
D2 LED
U1-C LM339N
R20 1.2K
R21 100K
R15 100K
D12 LED
U1-D LM339N
R29 1.2K
R30 100K
R28 100K
D15 LED
U2-A LM339N
R36 100K
R34 100K
D17 LED
U2-B LM339N
R38 100K
R37 100K
D18 LED
U2-C LM339N
R41 100K
R40 100K
D19 LED
U2-D LM339N
R44 100K
R43 100K
D20 LED
8155 Port B
R116 10K
R117 10K
R118 10K
R115 10K
R111 10K
R22 10K
SW4
DIPSWITCH
JP4
PTR6PIN
D62 1N4007
D64 1N4007
D53 1N4007
D54 1N4007
D55 1N4007
D56 1N4007
U3-A LM339N
R130 1.2K
R131 100K
R101 100K
D49 LED
U3-B LM339N
R103 1.2K
R104 100K
R132 100K
D63 LED
U3-C LM339N
R102 1.2K
R107 100K
R108 100K
D50 LED
U3-D LM339N
R121 1.2K
R122 100K
R127 100K
D52 LED
U4-A LM339N
R125 1.2K
R120 100K
R128 100K
D59 LED
U4-B LM339N
R129 1.2K
R123 100K
R105 100K
D61 LED
8155 Port C

VCC ---------> +5V

VPOS ---------> +15V

VNEG ---------> -15 V

GND

4.3 CIRCUIT OPERATION:

When a signal of 5V +, corresponding to logic 1 is present at the port lines, the output of the comparator is 5V (since V ref at the inverting terminal is 2.5 V). The LED at the comparator output becomes reverse biased and does not glow.

When a OV signal, i.e. (logic '0') is present at the port lines, the comparator o/p switches to -2.5V and the LEDs connected to it glow.

The DIP switches should be kept open, when the ports are tested for the output function. While testing the ports for input function, the DIP switches should be closed.

For testing the interrupts, the push button switch SW1 is pressed. The capacitor connected to it is charged to +5V volts through the 270 KΩ resistor. So the o/p of the comparator is at 5 volts. When SW1 is pressed again, the capacitor discharges through the 10KΩ resistor, making the comparator o/p 0V.

5. PROCEDURE FOR TESTING

To conduct a particular test, respective commands are sent from the terminal to the microcontroller. Results of testing are returned to the terminal. For this two way communication a set of protocols is framed.

Communication from the terminal to the microcontroller is called Downstream communication. The protocol used is

Starting character ---------- <

Ending character ---------- >

Communication from the micro controller to the terminal is called upstream communication the protocol used is,

Starting character ---------- {

Ending character ---------- }

5.1 LIST OF COMPONENTS TO BE TESTED:

1) IC 8032 - MICROCONTROLLER

2) IC 8155 - PERIPHERAL IC

3) IC ADC574 - ANALOG TO DIGITAL CONVERTER

4) IC DAC703 - DIGITAL TO ANALOG CONVERTER

5) DS 1230 - NVRAM

5.2 TESTING:

5.2.1 IC 8032:

5.2.1.a. Testing the port1:

As input:

Read the input(DIP) switch status and transmit the data to PC as and when read command is received from the PC.

Protocols used:

Downstream ------ <PI>

Upstream ------ {XX}

Where 'XX' is the return value corresponding to the DIP switch status.

As output:

All the switches connected to the port should be kept open. Move a known data, say, 55H or AAH to the port1 of the IC. The LED's connected to the port lines through the comparator glow according to the comparator output.

If the comparator o/p is low,then the LED glows.

If the comparator o/p is high,then the LED does not glow.

Protocols used:

Downstream ------ <PO>

Upstream ------ {O},if the IC is good.

5.2.1.b. Testing the Timer 0:

The timer of the 8032 microcontroller is tested by generating a pulse using the timer. The timer O is loaded with a particular value and the O/P pulse is produced at the port pin PI.O of the 8032 micro controller. The frequency of the pulse must be 1 KHz.

Protocols Used:

Downstream ---------- <T>

Upstream ---------- {O}

5.2.1.c. Testing the External Interrupts:

The external interrupt is tested by means of push button switch. When the switch is not pressed a return value from the micro controller appears when the push button switch is pressed the program executes the interrupt service routine which returns some other value. The result is discharged on the screen.

Protocols used

Downstream ------- <I0>,<I1>

Upstream ------- {O} for acknowledgement

{ }if the ISR is executed.

5.2.2 IC 8155

As input:

Read the input(DIP) switch status and transmit the data to PC as and when read command is received from the PC.

Protocols used:

Downstream ------ <8I>

Upstream ------ {XX }

Where 'XX' is the return value corresponding to the DIP switch status.

As output:

All the switches connected to the ports should be kept open. Move a known data, say, 55H or AAH to the ports A & B of the IC.The LED's connected to the port bits through the comparator glow according to the comparator output.

If the comparator o/p is low,then the LED glows.

If the comparator o/p is high,then the LED does not glow.

Protocols used:

Downstream ------ <8O>

Upstream ------ {O},if the IC is good.

{B},if the IC is bad.

5.2.3 ADC 574:

The known voltage from the voltage source is supplied externally to the analog in pin of the ADC. The 12 bit digital output from the ADC is obtained and displayed after the conversion.

Protocols Used:

Downstream ---------- <A>

Upstream ---------- {XXX}

XXX - refers to the 12 bit digital output value form the ADC.

5.2.4 DAC 703:

Move a known data to the input buffer of the DAC through the data bus from the 8032 IC. Using the formula calculate the equivalent analog output voltage. Then measure the output voltage of the IC at the analog output pin, using a voltmeter. If both the obtained values of AC voltage are equal, then the condition of the IC is good.

Digital Input	*Expected analog output*
0000H	---------- +10V
FFFFH	---------- -10 V
7FFFH	---------- 0V

Protocols used:

Downstream ------ <D1>, <D2>, <D3>

Upstream ------- {O}

Where D1 corresponds to +10 V, D2 to -10V and D3 to 0V.

5.2.5 DS 1230:

Load 55H in locations from 0000H to 7FFFH of the NVRAM IC, using write operation. Then verify if the same data, i.e., 55H is present in all the locations from 0000H to 7FFFH, using read operation.

Protocols used:

Downstream ----- <N>

Upstream ----- {O},if the IC is good.

{B},if the IC is bad.

If the loaded data is missing in any of these locations, then the LED connected to P 1.0 of 8032 glows, indicating that the IC is bad.

6. ASSEMBLY LANGUAGE PROGRAMMING

```
$INCLUDE (MAX_MEM1.DEF)
;*************************************************************************
;                KCT_PROJECT_TENSOMAXX TEST JIG
;*************************************************************************
; 8032 INITIALISATIONS FOR ASM51

T2CON  DATA 0C8H
RCAP2L DATA 0CAH
RCAP2H DATA 0CBH
TL2    DATA 0CCH
TH2    DATA 0CDH
PCON   EQU  087H
ET2    BIT  0ADH
PT2    BIT  0BDH
T2EX   BIT  091H
T2     BIT  090H
;******** T2CON BITS ;*****************************************************
TF2    BIT  0CFH
EXF2   BIT  0CEH
RCLK   BIT  0CDH
TCLK   BIT  0CCH
EXEN2  BIT  0CBH
TR2    BIT  0CAH
C_T2   BIT  0C9H
CP_RL2 BIT  0C8H
;*************************************************************************
ZERO            DATA    000H
SERIAL_CONTROL_WORD  DATA    050H ;SCON DATA
TMOD_VALUE      DATA    021H
TCON_VALUE      DATA    022H
;*************************************************************************
    TEMP    EQU 7FFFH
    PORT1   EQU 90H
;************8155 DECLARATION*******************************************
      PORTA   EQU 0F401H
      PORTB   EQU 0F402H
      PORTC   EQU 0F403H
      CW_8155  EQU 0F400H
;**********************DAC DECLARATION**********************************
      DAC_HI_BYT   EQU 0E000H
      DAC_LO_BYT   EQU 0E400H
      DAC_BIOSLO_BYT EQU 0EC00H
      DAC_BIOSHI_BYT EQU 0E800H
;-----------------------------------------------------------------
; INTERNAL RAM DECLARATION
;-----------------------------------------------------------------
SRL_DATA_PTR_LO      EQU     008H
SRL_DATA_PTR_HI      EQU     009H
RCV_FLAG         EQU     00AH
START_CHAR         EQU     00BH
RST_FLAG         EQU     012H
END_FLAG         EQU     013H
;--------------EXTERNAL RAM DECLARATION-----------------------------
STATUS_PTR         EQU     1000H

ORG 00H
```

```
LJMP START

ORG 03H       ;EXTERNAL INTERRUPT0
LJMP INTE_0

ORG 0BH       ;TIMER0 INTERRUPT
LJMP TIMER_0

ORG 13H       ;EXTERNAL INTERRUPT1
LJMP INTE_1

ORG 1BH       ;TIMER1 INTERRUPT
RETI

ORG 23H       ;SERIAL PORT INTERRUPT
LJMP SERIAL

ORG 2BH       ;TIMER2 INTERRUPT
RETI

ORG 200H

START:
   MOV  SP,#60H
   MOV  PSW,#0H
   LCALL INIT_8032
   LCALL INIT_VARIABLES
   LJMP  MAIN_PROGRAM

INIT_8032:
;*************************************************************************
;Initialise MICRO CONTROLLER
;*************************************************************************
;
     CLR   TR0                ;Clear Timer 0
     CLR   TR1                ;Clear Timer 1
     CLR   TR2                ;Clear Timer 2

     MOV   SCON,#SERIAL_CONTROL_WORD  ;Scon Value
     SETB  EA                 ;to enable interrupts
     MOV   TMOD,#TMOD_VALUE        ;Timer Mode Word
     MOV   PCON,#080H
     MOV   TL2,#0AAH
     MOV   TH2,#02FH
     MOV   TL1,#0F3H  ; FOR 4800 BAUD RATE WITH 12MHZ CRYSTAL
     MOV   TH1,#0F3H
     MOV   TL0,#0
     MOV   TH0,#0
     SETB  ET2
     SETB  ET1
     SETB  ET0
     SETB  IT0
     SETB  IT1
     CLR   TF2
     CLR   C_T2
     SETB  ES
     SETB  TR1
     CLR   RI
     CLR   TI
```

```
    RET
;*************************************************************************
INIT_VARIABLES:
;*************************************************************************
   MOV   SRL_DATA_PTR_LO,#00H
   MOV   SRL_DATA_PTR_HI,#10H
   MOV   RCV_FLAG,#0FFH
   MOV   START_CHAR,#0FFH
   MOV   RST_FLAG,#0FFH
;*************************************************************************
 SERIAL:
;*************************************************************************
 ;CHECK FOR STARTING CHARACTER FROM PC
    JB   RI,CHK_FOR_START_CHAR
    CLR  TI
    RETI

 CHK_FOR_START_CHAR:
    PUSH ACC
    CLR  RI
    MOV  A,SBUF
    MOV  R1,A
    CJNE A,#'<',CHK_FOR_END_CHAR
    MOV  START_CHAR,#055H

  RET_SRL:
    POP  ACC
    RETI
 ;*************************************************************************
 CHK_FOR_END_CHAR:
    MOV  A,START_CHAR
    CJNE A,#055H,RET_SRL
    MOV  A,R1
    CJNE A,#'>',STORE_DATA
    MOV  RCV_FLAG,#055H
    MOV  START_CHAR,#0FFH
    MOV  SRL_DATA_PTR_LO,#00H
    MOV  SRL_DATA_PTR_HI,#10H

  DO_SRL_RTRN:
    POP ACC
    RETI

  STORE_DATA:
    MOV  DPL,SRL_DATA_PTR_LO
    MOV  DPH,SRL_DATA_PTR_HI
    MOV  A,R1
    MOVX @DPTR,A
    INC  DPTR
    MOV  SRL_DATA_PTR_LO,DPL
    MOV  SRL_DATA_PTR_HI,DPH
    LJMP DO_SRL_RTRN
  ;*************************************************************************
  CHECK_COMMAND:
  ;*************************************************************************
      MOV  DPTR,#STATUS_PTR
      MOVX A,@DPTR
      MOV  R2,A
```

```
     CJNE  A,#'P',CHK_TIMER
     LJMP  PORT
CHK_TIMER:
     MOV   A,R2
     CJNE  A,#'T',CHK_INTE
     LJMP  TIMER
CHK_INTE:
     MOV   A,R2
     CJNE  A,#'I',CHK_8155
     LJMP  INT
CHK_8155:
     MOV   A,R2
     CJNE  A,#'8',CHK_NVRAM
     LJMP  PR_8155
CHK_NVRAM:
     MOV   A,R2
     CJNE  A,#'N',CHK_DAC
     LJMP  NVRAM
CHK_DAC:
     MOV   A,R2
     CJNE  A,#'D',CHK_ADC
     LJMP  DAC
CHK_ADC:
     MOV   A,R2
     CJNE  A,#'A',ERROR
     LJMP  ADC
ERROR:
     RET
;*****************************************************************************
; TO CHECK THE PORTS OF 8032
;*****************************************************************************
PORT:
     INC  DPTR
     MOVX A,@DPTR
     MOV  R3,A
     CJNE A,#'I',CHK_OUT
     MOV  A,P1          ;CHK_IN
     MOV  R4,A
     LCALL SEND_START_CHAR
     LCALL ASC_CONV
     LCALL SEND_END_CHAR
     RET
CHK_OUT:
     MOV  P1,#55H
     LCALL DELAY
     MOV  P1,#0AAH
     LCALL DELAY
     LCALL SEND_START_CHAR
     LCALL SEND_DATA
     LCALL SEND_END_CHAR
     RET
;*****************************************************************************
TIMER:
;*****************************************************************************
     SETB  ET0
     SETB  P1.0
     MOV   TL0,#00BH
     MOV   TH0,#0FEH
```

```
    SETB  TR0
    RET
;*******************************************************************************
TIMER_0:
;*******************************************************************************
    CLR   TR0
    CPL   P1.0
    MOV   TL0,#0BH
    MOV   TH0,#0FEH
    SETB  TR0
    RETI
;*******************************************************************************
;INTERRUPT CHECK
;*******************************************************************************
INT:
    INC  DPTR
    MOVX A,@DPTR
    MOV  R2,A
    CJNE A,#'0',INT_1
    MOV  END_FLAG,#055H
    LCALL SEND_START_CHAR
    LCALL SEND_DATA
    LCALL SEND_END_CHAR
    RET
INT_1:
    MOV  END_FLAG,#055H
    LCALL SEND_START_CHAR
    LCALL SEND_DATA
    LCALL SEND_END_CHAR
    RET
;*******************************************************************************
INTE_0:
    MOV  A,END_FLAG
    CJNE A,#055H,SS
    LCALL SEND_START_CHAR
    LCALL SEND_DATA_0
    LCALL SEND_END_CHAR
    MOV  END_FLAG,#0FFH
 SS: RETI
;*******************************************************************************
INTE_1:
    MOV  A,END_FLAG
    CJNE A,#055H,DD
    LCALL SEND_START_CHAR
    LCALL SEND_DATA_1
    LCALL SEND_END_CHAR
    MOV  END_FLAG,#0FFH
 DD: RETI
;*******************************************************************************
PR_8155:
    INC  DPTR
    MOVX A,@DPTR
    MOV  R3,A
    CJNE A,#'I',C_OUT

    MOV  DPTR,#CW_8155
    MOV  A,#00H
    MOVX @DPTR,A
```

```
   MOV  DPTR,#PORTA
   MOVX A,@DPTR
   MOV  R4,A
   LCALL SEND_START_CHAR
   LCALL ASC_CONV
   LCALL SEND_END_CHAR

   MOV  DPTR,#PORTB
   MOVX A,@DPTR
   MOV  R4,A
   LCALL SEND_START_CHAR
   LCALL ASC_CONV
   LCALL SEND_END_CHAR

   MOV  DPTR,#PORTC
   MOVX A,@DPTR
   MOV  R4,A
   LCALL SEND_START_CHAR
   LCALL ASC_CONV
   LCALL SEND_END_CHAR

   RET
;*****************************************************************************
;
C_OUT:
    MOV   DPTR,#CW_8155
    MOV   A,#03H
    MOVX  @DPTR,A

    MOV   DPTR,#PORTA
    MOV   A,#55H
    MOVX  @DPTR,A

    MOV   DPTR,#PORTB
    MOV   A,#0AAH
    MOVX  @DPTR,A
    LCALL  DELAY

    MOV   DPTR,#PORTA
    MOV   A,#0AAH
    MOVX  @DPTR,A

    MOV   DPTR,#PORTB
    MOV   A,#55H
    MOVX  @DPTR,A

    LCALL  SEND_START_CHAR
    LCALL  SEND_DATA
    LCALL  SEND_END_CHAR

    RET
;*****************************************************************************
;
; TO TEST NVRAM
;*****************************************************************************
;
NVRAM:
    MOV   DPTR,#TEMP
WRITE: MOV   A,#05H
    MOVX  @DPTR,A
```

```
    LCALL  DECR
    MOV    A,DPL
    CJNE   A,#0,WRITE
    MOV    A,DPH
    CJNE   A,#0,WRITE
    LJMP   READ
;*********************DECREMENT DPTR************************************
DECR:
    CLR    C
    MOV    A,DPL
    SUBB   A,#01H
    MOV    DPL,A
    MOV    A,DPH
    SUBB   A,#00H
    MOV    DPH,A
    RET
;*****************************************************************************
READ:
    MOV    DPTR,#TEMP
READ_1:
    MOVX   A,@DPTR
    CJNE   A,#05H,ERR
    LCALL  DECR
    MOV    A,DPL
    CJNE   A,#0,READ_1
    MOV    A, DPH
    CJNE   A,#0,READ_1
    LCALL  SEND_START_CHAR
    LCALL  SEND_DATA
    LCALL  SEND_END_CHAR
    RET
ERR:
    LCALL  SEND_START_CHAR
    LCALL  SEND_DATA_B
    LCALL  SEND_END_CHAR
    RET
;*****************************************************************************
DAC:
;*****************************************************************************
    INC    DPTR
    MOVX   A,@DPTR
    MOV    R5,A
    CJNE   A,#'1',M10V
P10V:
    MOV    DPTR,#DAC_HI_BYT
    MOV    A,#00H
    MOVX   @DPTR,A
    MOV    DPTR,#DAC_LO_BYT
    MOV    A,#00H
    MOVX   @DPTR,A
    LCALL  SEND_START_CHAR
    LCALL  SEND_DATA
    LCALL  SEND_END_CHAR
    RET
M10V:
    MOV    A,R5
    CJNE   A,#'2',OVAL
    MOV    DPTR,#DAC_HI_BYT
```

```
    MOV   A,#0FFH
    MOVX  @DPTR,A
    MOV   DPTR,#DAC_LO_BYT
    MOV   A,#0FFH
    MOVX  @DPTR,A
    LCALL SEND_START_CHAR
    LCALL SEND_DATA
    LCALL SEND_END_CHAR
    RET
OVAL:
    MOV   A,R5
    MOV   DPTR,#DAC_HI_BYT
    MOV   A,#7FH
    MOVX  @DPTR,A
    MOV   DPTR,#DAC_LO_BYT
    MOV   A,#0FFH
    MOVX  @DPTR,A
    LCALL SEND_START_CHAR
    LCALL SEND_DATA
    LCALL SEND_END_CHAR
    RET
;*************************************************************************
;
ADC:
;*************************************************************************
;
    MOV   OP_0,#045H   ;60H.01H
WAIT_FOR_STABLE:
    MOV   INTER3,#010H
    MOV   INTER0,#0
    MOV   INTER1,#0
    MOV   INTER2,#0
DO_TARE:
    MOV   DATA_HI,#0
    MOV   DATA_LO,#0
    CLR   A
    MOV   DPTR,#STRT_CONV
    MOVX  @DPTR,A        ;START CONVERSION
    NOP
    NOP
    NOP
    NOP
    NOP

STL_CON:
    SETB  P1.0
    JB    P1.0,STL_CON
    MOVX  A,@DPTR          ;HIGHER 8 BITS
    MOV   DATA_HI,A
    MOV   DPTR,#LS_DATA_RD   ;LOWER 4 BITS AND TRAILING ZEROES
    MOVX  A,@DPTR
    CLR   C
    ADDC  A,INTER0
    MOV   INTER0,A
    MOV   A,DATA_HI
    ADDC  A,INTER1
    MOV   INTER1,A
    MOV   A,#0
    ADDC  A,INTER2
```

```
    DJNZ  INTER3,DO_TARE
    DJNZ  OP_0,WAIT_FOR_STABLE

    MOV   S1,INTER2
    MOV   S0,INTER1
    MOV   DATA_HI,S1
    MOV   DATA_LO,S0
    MOV   TMP_1,TH0
    MOV   TMP_0,TL0

    LCALL SEND_START_CHAR
    MOV   A,S1
    ANL   A,#00FH
    LCALL CHECK_NUM
    MOV   R4,S0
    LCALL ASC_CONV
    LCALL SEND_END_CHAR
    RET
;*****************************************************************************
SEND_START_CHAR:
    CLR   TI
    CLR   ES
    MOV   SBUF,#'{'
TX3:  JNB   TI,TX3
    CLR   TI
    SETB  ES
    RET
;*****************************************************************************
SEND_END_CHAR:
    CLR   TI
    CLR   ES
    MOV   SBUF,#'}'
TX4:  JNB   TI,TX4
    CLR   TI
    SETB  ES
    RET
;*****************************************************************************
SEND_DATA:
    CLR   TI
    CLR   ES
    MOV   SBUF,#'O'
TX5:  JNB   TI,TX5
    CLR   TI
    SETB  ES
    RET
;*****************************************************************************
SEND_DATA_B:
    CLR   TI
    CLR   ES
    MOV   SBUF,#'B'
TX6:  JNB   TI,TX6
    CLR   TI
    SETB  ES
    RET
;*****************************************************************************
SEND_DATA_0:
    CLR   TI
    CLR   ES
```

```
    MOV   SBUF,#'Z'
TX7:  JNB   TI,TX7
    CLR   TI
    SETB  ES
    RET
;*****************************************************************************
SEND_DATA_1:
    CLR   TI
    CLR   ES
    MOV   SBUF,#'B'
TX8:  JNB   TI,TX8
    CLR   TI
    SETB  ES
    RET
;*****************************************************************************
CHECK_NUM:
;*****************************************************************************
    MOV   R3,A
    SUBB  A,#0AH
    JNC   ADD_37H
    MOV   A,R3
    CLR   C
    ADD   A,#030H
    CLR   ES
    MOV   SBUF,A
TX1:  JNB   TI,TX1
    CLR   TI
    SETB  ES
    RET
ADD_37H:
    MOV   A,R3
    CLR   C
    ADD   A,#037H
    CLR   ES
    MOV   SBUF,A
TX2:  JNB   TI,TX2
    CLR   TI
    SETB  ES
    RET
;*****************************************************************************
ASC_CONV:
;*****************************************************************************
    MOV   A,R4
    CLR   C
    ANL   A,#0F0H
    SWAP  A
    LCALL CHECK_NUM
    MOV   A,R4
    CLR   C
    ANL   A,#0FH
    LCALL CHECK_NUM
    RET
;*****************************************************************************
DELAY:
    MOV   R0,#02EH
XY:   LCALL LONG_DELAY_1
    DJNZ  R0,XY
    RET
```

```
LONG_DELAY_1:
;*************************************************************************
;
    MOV  R1,#0FFH
WW: MOV  R2,#0FFH
ZZ: DJNZ R2,ZZ
    DJNZ R1,WW
    RET
;*************************************************************************
;
MAIN_PROGRAM:
;*************************************************************************
;
   MOV  A,RCV_FLAG
   CJNE A,#055H,MAIN_PROGRAM
   MOV  RCV_FLAG,#0FFH
   LCALL CHECK_COMMAND
   SJMP MAIN_PROGRAM
;*************************************************************************
;
END
```

7. FRONT END USING 'C'

7.1 ABOUT FRONT END:

In this project, a front-end software is written using the 'C' language. The purpose of this software is to provide a user friendly and interactive testing system.

The front end displays main menu consisting the list of components to be tested. When a particular component is selected, a sub-menu is displayed, which contains the list of tests to be performed on that component. Finally when the desired test is selected, the corresponding commands (protocols are defined in earlier chapter) are transmitted serially from the PC to the microcontroller through the serial port COM1.

After the testing is over, characters are returned from the microcontroller to the PC. 'C' identifies the characters and displays appropriate message. Thus the results of testing can be viewed in the screen.

To enable the serial communication between the PC and the microcontroller, the function BIOSSERIAL COM / BIOS COM in 'C' language is used. This function is present in the header file BIOS.H and it offers a lot of options for data communication.

7.2 BIOSCOM

bioscom and _bios_serialcom <BIOS.H>

RS-232 communications (serial I/O)

Declaration:

int bioscom(int cmd, char abyte, int port);

unsigned _bios_serialcom(int cmd, int port, char abyte);

Remarks

Both bioscom and _bios_serialcom use BIOS interrupt 0x14 to perform various

RS-232 communications over the I/O port given in port.

Arg. ³ What It Is/Does

abyte ³ OR combination of bits that specifies COM port settings

³ (ignored if cmd = 2 or 3)

cmd ³ Specifies the I/O operation to perform

port ³ Identifies the I/O port; 0 = COM1, 1 = COM2, etc.

Return Value:

For all values of cmd, both functions return a 16-bit integer.

The upper 8 bits of the return value are status bits.

If one or more status bits is set to 1, an error has occured.

If no status bits are set to 1, the byte was received without error.

The lower 8 bits of the return value depend on the value of cmd specified:

Value of cmd Lower 8 bits of return value

0 (_COM_INIT) or 3 (_COM_STATUS) The lower bits are defined as shown in the preceding diagram.

1 (_COM_SEND) ...

2 (_COM_RECEIVE) The byte read is in the lower bits of the return value--if there is no error (no upper bits are set to 1).

cmd argument of bioscom and _bios_serialcom

cmd can be one of the following (defined in BIOS.H):

bioscom_bios_serialcom What Function Does

0 _COM_INIT Sets the communications parameters to the value in abyte

1 _COM_SEND Sends the character in abyte out over the communications line

2 _COM_RECEIVE Receives a character from the communications line

3 _COM_STATUS Returns current status of the communications port

When cmd = 2 or 3 (_COM_RECEIVE or _COM_STATUS), the abyte argument is ignored.

When cmd = 0 (_COM_INIT), abyte is an OR combination of the following bits (one from each group):

bioscom[3]_bios_serialcom	Meaning
0x02 [3]_COM_CHR7	7 data bits
0x03 [3]_COM_CHR8	8 data bits
0x00 [3]_COM_STOP1	1 stop bit
0x04 [3]_COM_STOP2	2 stop bits
0x00 [3]_COM_NOPARITY	No parity
0x08 [3]_COM_ODDPARITY	Odd parity
0x18 [3]_COM_EVENPARITY	Even parity
0x00 [3]_COM_110	110 baud
0x20 [3]_COM_150	150 baud

0x40 ³_COM_300 300 baud

0x60 ³_COM_600 600 baud

0x80 ³_COM_1200 1200 baud

0xA0 ³_COM_2400 2400 baud

0xC0 ³_COM_4800 4800 baud

0xE0 ³_COM_9600 9600 baud

For example, if

abyte = 0xEB = (0xE0 | 0x08 | 0x00 | 0x03)

= (_COM_9600 | _COM_ODDPARITY | _COM_STOP1 |

_COM_CHR8)

the communications port is set to

9600 baud (0xE0 = _COM_9600)

Odd parity (0x08 = _COM_ODDPARITY)

1 stop bit (0x00 = _COM_STOP1)

8 data bits (0x03 = _COM_CHR8)

Upper 8 bits of the bioscom and _bios_serialcom return value

15 Time out (set to 1 if abyte value could not be sent)

14 Transmit shift register empty

13 Transmit holding register empty

12 Break detect

11 Framing error

10 Parity error

9 Overrun error

8 Data ready

Lower bits (these vary, depending on the value of cmd)

Status bits 0 Received line signal detect

1 Ring indicator

2 Data set ready

3 Clear to send

4 Change in receive line signal detector

5 Trailing edge ring detector

6 Change in data set ready

7 Change in clear to send

7.3 SOURCE CODE

```
/* FRONTEND SOFTWARE FOR TENSOMAXX TESTING EQUIPMENT */
#include <bios.h>
#include <conio.h>
#include<stdio.h>
#include <string.h>
#define COM1      0
#define DATA_READY 0x100
#define TRUE      1
#define FALSE     0
#define SETTINGS ( 0xC0| 0x00| 0x00|0x03)

void chk_8032();
void chk_8155();
void chk_ADC574();
void chk_DAC703();
void chk_ds1230();
void setting_mode();
void main(void)
 {
  int option;
  char conf;
  clrscr();
  bioscom(0, SETTINGS, COM1);
  printf(" \n\n\t MICROCONTROLLER BASED MULTIDATA TESTER \n");
  printf(" \n\n\t -BY KUMARAGURU COLLEGE OF TECHNOLOGY \n");
  printf(" \n\n\t ENTER YOUR CHOICE \n\n\n");
  printf("\t 1 = TO TEST IC8032  \n");
  printf("\t 2 = TO TEST IC8155  \n");
  printf("\t 3 = TO TEST ADC574  \n");
  printf("\t 4 = TO TEST DAC703  \n");
  printf("\t 5 = TO TEST DS1230-NVRAM \n");

  scanf("%d",&option);
  fflush(stdin);
  switch(option)
    {
  case 1:

 clrscr();
 printf("\n YOU HAVE ENTERED IN 8032 TEST MODE \n");
 printf("\n ARE YOU SURE TO CHECK THIS DEVICE \n");
 printf("\n ENTER Y OR N \n");
 conf=getch();
 fflush(stdin);
 while ((conf=='Y')||(conf=='y') && (conf!='N')||(conf!='n'))
 {
 chk_8032();
 getch();
 }
      break;
 case 2:
 clrscr();
      printf("YOU HAVE ENTERED IN 8155 TEST MODE \n");
 printf("ARE YOU SURE TO CHECK THIS DEVICE \n");
 printf("ENTER Y OR N \n");
```

```
conf=getch();
fflush(stdin);
while ((conf =='Y')||(conf =='y') && (conf !='N')||(conf !='n'))
{
chk_8155();
}
    break;
case 3:
    clrscr();
        printf("\t YOU HAVE ENTERED IN ADC 574 TEST MODE \n");
        printf("\t ARE YOU SURE TO CHECK THIS DEVICE \n");
        printf("\t ENTER Y OR N \n");
        conf=getch();
        fflush(stdin);

        while ((conf =='Y')||(conf =='y') && (conf !='N')||(conf !='n'))
                {
printf("\n\n\t MEASURE VOLTAGE BETWEEN PIN 8 AND Gnd. (AD574) (10v ñ 0.05v\n");
printf("\t MEASURE VOLTAGE BETWEEN PIN 10 AND Gnd.(Value -5v ñ 0.05v) \n");
printf("\t IF NOT , ADJUST POT 8 TO GET REQUIRED OUTPUT (5v ñ 0.05v ) \n");
printf("\t MEASURE VOLTAGE BETWEEN PIN 12 AND Gnd(0v ñ 0.05v) [else trim POT 6] \n");
chk_ADC574();
getch();
exit(0);
                }
break;

case 4:
   /* TO CHECK THE IC DAC703    */

clrscr();
printf("\t YOU HAVE ENTERED IN DAC 03 TEST MODE \n");
printf("\t ARE YOU SURE TO CHECK THIS DEVICE \n");
printf("\t ENTER Y OR N \n");
conf=getch();
fflush(stdin);

while ((conf =='Y')||(conf =='y') && (conf !='N')||(conf !='n'))
{
chk_DAC703();
getch();
        }
break;
case 5:

/* TO TEST THE NVRAM-DS1230  */
clrscr();
printf("\t YOU HAVE ENTERED IN DS1230 TEST MODE \n");
printf("\t ARE YOU SURE TO CHECK THIS DEVICE \n");
printf("\t ENTER Y OR N \n");
conf=getch();
fflush(stdin);
while ((conf =='Y')||(conf =='y') && (conf !='N')||(conf !='n'))
{
chk_ds1230();
}
break;
```

```
  default:
                printf(" SORRY,YOU HAVE GIVEN A WRONG ENTRY\n");
                getch();
                exit();
          }
 }

void chk_8032()
{
        int a, j,l, status;
        char k,string1[6],string2[6],string11[6],string12[6],string3[6],out;
        printf(" \n\n\t ENTER YOUR CHOICE TO CHECK  \n");
        printf("\t 1 - TO TEST THE PORTS OF 8032  \n");
        printf("\t 2 - TO TEST THE TIMER0  \n");
        printf("\t 3 - TO TEST THE EXTERNAL INTERRUPTS 0 & 1 \n");
        scanf("%d",&j);
        fflush(stdin);

        switch(j)
{
   case 1:

                printf("\t TO CHECK PORT1 FOLLOW THE PROCEDURES GIVEN \n");
                printf("\t ENTER THE OPTION \n");
                printf("\t I TO CHECK AS INPUT \n");
                printf("\t O TO CHECK AS OUTPUT \n");
                scanf("%c",&k);
                fflush(stdin);

                if(k=='I')
          {
           strcpy ( string1, " <PI>");
           a=strlen(string1);
           for( l=0;l<a;l++)
                 {
                 bioscom(1,string1[l],COM1);
                  }
                  status = bioscom(3, 0, COM1);
                   if (status & DATA_READY)
                  {
                  if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
           {
            putch(out);
            if ( out == '}')
            break;
           }
                 }
          }

                if(k=='O')
  {
           strcpy ( string2, "<PO>");
           a=strlen(string2);
           for( l=0;l<a;l++)
                 {
                 bioscom(1,string2[l],COM1);
                 }
                 status = bioscom(3, 0, COM1);
```

```
        if (status & DATA_READY)
        {
         if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
        {
         putch(out);
         if ( out == '}')
         break;
         }
                }

      }   getch();
          break;
    case 2:
                printf("\nTO CHECK TIMER FOLLOW THE PROCEDURES GIVEN \n");
                printf("ENTER THE OPTION \n");
                printf("T TO CHECK TIMER MODE  \n");
                scanf("%c",&k);
                fflush(stdin);
                if(k=='T')


{
        strcpy ( string3, "<T>");
        a=strlen(string3);
        for( l=0;l<a;l++)
                {
                bioscom(1,string3[l],COM1);
                }
                status = bioscom(3, 0, COM1);
                if (status & DATA_READY)
                {
                if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
         {
         putch(out);
         if ( out == '}')
         break;
         }
                }
        }
        break;
   case 3:
                printf("\n\t TO CHECK INTERRUPT FOLLOW THE PROCEDURES GIVEN \n");
                printf("\t ENTER THE OPTION \n");
                printf("\t I TO CHECK AS INTERRUPT 0 \n");
                printf("\t P TO CHECK AS INTERRUPT 1 \n");
                scanf("%c",&k);
                fflush(stdin);

                if(k=='I')
   {
          strcpy ( string11, " <I0>");
          a=strlen(string11);
          for( l=0;l<a;l++)
                {
                bioscom(1,string11[l],COM1);
                }
          status = bioscom(3, 0, COM1);
```

```
        if (status & DATA_READY)
        {
        if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
        {
         putch(out);
         if ( out == '}')
         break;
         }
               }

                printf("\n\t PRESS THE PUSH BUTTON T0 CHECK INTERRUPT0 AFTER
{O}APPEARS ON SCREEN \n");
                printf("\t WAITING FOR RECEPTION \n");
        { status = bioscom(3, 0, COM1);
          if (status & DATA_READY)
          {
          if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
          {
          putch(out);
          if ( out == '}')
          break;
          }
        }
    }
    }
          if(k=='P')

    {
          strcpy ( string12, "<I1>");
          a=strlen(string12);
          for( l=0;l<a;l++)
                {
           bioscom(1,string12[l],COM1);
           }
           status = bioscom(3, 0, COM1);
           if (status & DATA_READY)
           {
           if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
          {
           putch(out);
           if ( out == '}')
           break;
           }
                 }
           }
   printf("\n\t PRESS THE PUSH BUTTON T0 CHECK INTERRUPT1 AFTER {O}APPEARS ON
 SCREEN \n");
   printf("\t WAITING FOR RECEPTION \n");
          {
             status = bioscom(3, 0, COM1);
             if (status & DATA_READY)
             {
             if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
            {
             putch(out);
             if ( out == '}')
             break;
```

```
        }
      }
  }
        getch();
         break;
              default:
              printf("YOU HAVE GIVEN A WRONG ENTRY \n");
              getch();
              exit();
        }
  }


void chk_8155()
{
        int a,i,j,l,status;
        char k, string4[6],string5[6],out;
        printf(" \n\n ENTER YOUR CHOICE TO CHECK  \n");
        printf(" 1 TO  THE PORT OF 8155  \n");
        scanf("%d",&j);
        fflush(stdin);

        switch(j)
        {
         case 1:
              printf("TO CHECK PORT1 FOLLOW THE PROCEDURES GIVEN \n");
              printf("ENTER THE OPTION \n");
              printf("I TO CHECK AS INPUT \n");
              printf("O TO CHECK AS OUTPUT \n");
              scanf("%c",&k);
              fflush(stdin);
              if(k=='I')
 {
          strcpy ( string4, " <8I>");
         a=strlen(string4);
         for( l=0;l<a;l++)
              {
              bioscom(1,string4[l],COM1);
              }
              status = bioscom(3, 0, COM1);
              if (status & DATA_READY)
         {
         if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
         {
         putch(out);
         if ( out == '}')
         break;
         }
        }
              status = bioscom(3, 0, COM1);
              if (status & DATA_READY)
         {
         if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
         {
         putch(out);
         if ( out == '}')
         break;
         }
```

```
        }       status = bioscom(3, 0, COM1);
                if (status & DATA_READY)
              {
              if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
              {
              putch(out);
              if ( out == '}')
              break;
              }

        }
   }
                if(k=='O')
{
                 printf("\n THE LED'S THAT SHOULD GLOW ARE \n");
                 printf("LED1 IN FIRST PAIR OF ROW1 .\n");
                 printf("LED2 IN SECOND PAIR OF ROW1 . \n");
                 printf("LED2 IN THIRD PAIR OF ROW1 .\n");
                 printf("LED1 IN FOURTH PAIR OF ROW1  . \n");
                 printf("\n\nLED1 IN FIRST PAIR OF ROW2 .\n");
                 printf("LED2 IN SECOND PAIR OF ROW2 . \n");
                 printf("LED2 IN THIRD PAIR OF ROW2 .\n");
                 printf("LED1 IN FOURTH PAIR OF ROW2  . \n");
                 printf("\nAFTER 5 SEC'S THE REST OF LED'S SHOULD GLOW \n");
                 printf( "\n\n\t IF THE RETURN CHAR IS {O} IC IS GOOD \n");
                 printf("\n\t IF THE RETURN CHAR IS {B} IC IS BAD  \n");
              strcpy ( string5, "<8O>");
              a=strlen(string5);
              for( l=0;l<a;l++)
                      {
                      bioscom(1,string5[l],COM1);
                      }
                      status = bioscom(3, 0, COM1);
                      if (status & DATA_READY)
              {

                      if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
              {
               putch(out);
               if ( out == '}')
               break;
               }
                      }

         }    getch();
                      break;
          default:
                      printf("YOU HAVE GIVEN A WRONG ENTRY\n");
                      getch();
                      exit();
             }
        }



  void chk ds1230()
```

```
        {
          int  a,b,l,status;
          char string9[6],out;
          printf( "\n\t IF THE RETURN CHAR IS {O} IC IS GOOD \n");
          printf("\n\t IF THE RETURN CHAR IS {B} IC IS BAD AND LED AT PORT P1.0 WILL
GLOW  \n");
           strcpy ( string9, "<N>");
           a=strlen(string9);
           for( l=0;l<a;l++)
                 {
                  bioscom(1,string9[l],COM1);
                   }
                 status = bioscom(3, 0, COM1);
                 if (status & DATA_READY)
                                                                                    {

            if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
            {
             putch(out);
                    }
                  }
                  getch();
                  exit();
    }
   void  chk_DAC703()
          {
           int a, j,l,status;
           char k, string6[6],string7[6],string8[6],out;
           printf(" \n\n ENTER YOUR CHOICE TO CHECK   \n");
           printf("\t 1  OUTPUT IS 10V       \n");
           printf("\t 2  OUTPUT IS -10V       \n");
           printf("\t 3  OUTPUT IS 0V       \n");
           scanf("%d",&j);
           fflush(stdin);

           switch(j)
           {
            case 1:
             strcpy ( string6, " <D1>");
             a=strlen(string6);
            for( l=0;l<a;l++)
                  {
                        bioscom(1,string6[l],COM1);
                        }
                        status = bioscom(3, 0, COM1);
                  if (status & DATA_READY)
             {

                  if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
             {
              putch(out);
              if ( out == '}')
              break;
              }
                   }
             break;
             case 2:
             strcpy ( string7, " <D2>");
```

```
        for( l=0;l<a;l++)
              {
                        bioscom(1,string7[l],COM1);
              if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
        {
         putch(out);
         if ( out == '}')
         break;
         }
              }
        break;
        case 3:
         strcpy ( string8, " <D3>");
         a=strlen(string8);
         for( l=0;l<a;l++)
              {
                        bioscom(1,string8[l],COM1);
              if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
         {
          putch(out);
          if ( out == '}')
          break;
          }
              }
        break;
       default:
              printf("YOU HAVE GIVEN A WRONG ENTRY \n");
              getch();
              exit();
          }
       }
void chk_ADC574()
       {
         int a,b,l,status;
         char string13[6],out;
          strcpy ( string13, "<N>");
          a=strlen(string13);
          for( l=0;l<a;l++)
                {
                 bioscom(1,string13[l],COM1);
                  }
                status = bioscom(3, 0, COM1);
                if (status & DATA_READY)
          {
           if ((out = bioscom(2,0, COM1 ) & 0x7F) != 0)
           {
           putch(out);
                }
                }
                getch();
                exit();
  }
```

8. CONCLUSION

The project was initiated to perform testing of IC's for the inward of PPL R&D. the IC's that are being tested are primarily used in the TENSOMAXX-7000 (single yarn tester). This machinery is manufactured only in a 3 places all over the world. Hence the IC's used in them must be thoroughly tested before they are in field. Hence this project is of great commercial use. Also the commercial IC tester available is costly and we do not know what parameters are being tested in them.

The project designed is cheaper, efficient and showed an excellent performance. The project is flexible in the sense that it can be extended to test other IC's by simply changing the program.

In this project the attempt was made to support the R&D and production line of PREMIER POLYTRONICS LTD. If this target system is incorporated in all the machines, it will help the customers to solve most of their problems immediately without wanting for the company personnel to visit the site.

BIBLIOGRAPHY

- RAMESH. S. GAONKAR – "MICROPROCESSOR ARCHITECTURE AND APPLICATIONS WITH 8085", Published by PENRAM INTERNATIONAL.

- ROY CHOUDHARY – "LINEAR INTEGRATED CIRCUITS", Published by NEW AGE INTERNATIONAL.

- DALLAS SEMICONDUCTORS, "SYSTEMS EXTENSIONS DATABOOK", 1995.

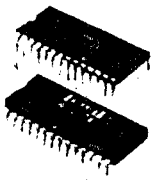
- DATA CONVERSION PRODUCTS, "BURR-BROWN IC DATABOOK", 1995.

- E. BALAGURUSAMY, "PROGRAMMING IN ANSI C", Published by TATA McGRAW HILL, THIRD EDITION.

- INTEL MEMORY COMPONENT BOOKS, 1989.

- POWER ICs DATABOOK Published by NATIONAL SEMICONDUCTORS.

BURR-BROWN®



ADC574A

Microprocessor-Compatible ANALOG-TO-DIGITAL CONVERTER

FEATURES

- COMPLETE 12-BIT A/D CONVERTER WITH REFERENCE, CLOCK, AND 8-, 12-, or 16-BIT MICROPROCESSOR BUS INTERFACE
- IMPROVED PERFORMANCE SECOND SOURCE FOR 574A-TYPE A/D CONVERTERS
  Conversion Time: 25µs max
  Bus Access Time: 150ns max
  A_0 Input: Bus Contention During Read Operation Eliminated
- DUAL IN-LINE PLASTIC, PLCC AND HERMETIC CERAMIC
- FULLY SPECIFIED FOR OPERATION ON ±12V OR ±15V SUPPLIES
- NO MISSING CODES OVER TEMPERATURE:
  0°C to +75°C: ADC574AJ and K Grades
  –55°C to +125°C: ADC574ASH, TH

DESCRIPTION

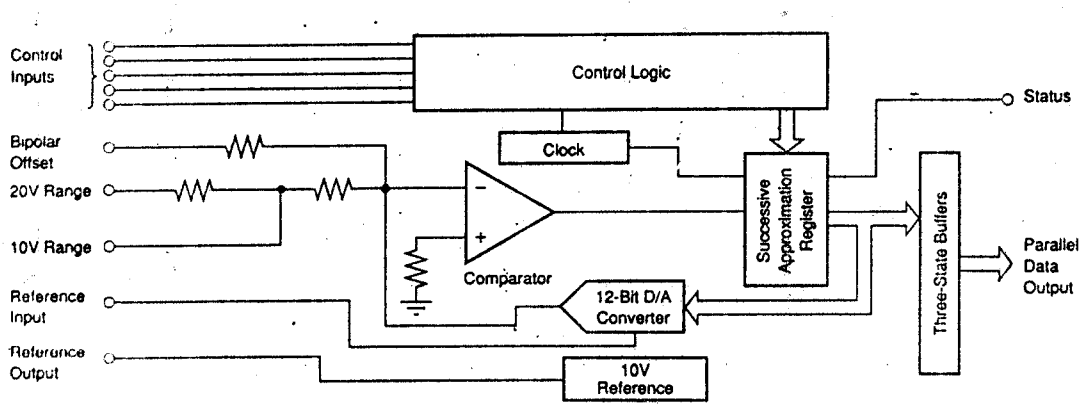
The ADC574A is a 12-bit successive approximation analog-to-digital converter, utilizing state-of-the-art CMOS and laser-trimmed bipolar die custom-designed for freedom from latch-up and for optimum AC performance. It is complete with a self-contained +10V reference, internal clock, digital interface for microprocessor control, and three-state outputs.

The reference circuit, containing a buried zener, is laser-trimmed for minimum temperature coefficient. The clock oscillator is current-controlled for excellent stability over temperature. Full-scale and offset errors may be externally trimmed to zero. Internal scaling resistors are provided for the selection of analog input signal ranges of 0V to +10V, 0V to +20V, ±5V, and ±10V.

The converter may be externally programmed to provide 8- or 12-bit resolution. The conversion time for 12 bits is factory set for 25µs maximum.

Output data are available in a parallel format from TTL-compatible three-state output buffers. Output data are coded in straight binary for unipolar input signals and bipolar offset binary for bipolar input signals.

The ADC574A, available in both industrial and military temperature ranges, requires supply voltages of +5V and ±12V or ±15V. It is packaged in a 28-pin plastic DIP, and a hermetic side-brazed ceramic DIP.



International Airport Industrial Park • Mailing Address: PO Box 11400 • Tucson, AZ 85734 • Street Address: 6730 S. Tucson Blvd. • Tucson, AZ 85706
Tel: (602) 746-1111 • Twx: 910-952-1111 • Cable: BBRCORP • Telex: 066-6491 • FAX: (602) 889-1510 • Immediate Product Info: (800) 548-6132

SPECIFICATIONS (CONT)

ELECTRICAL

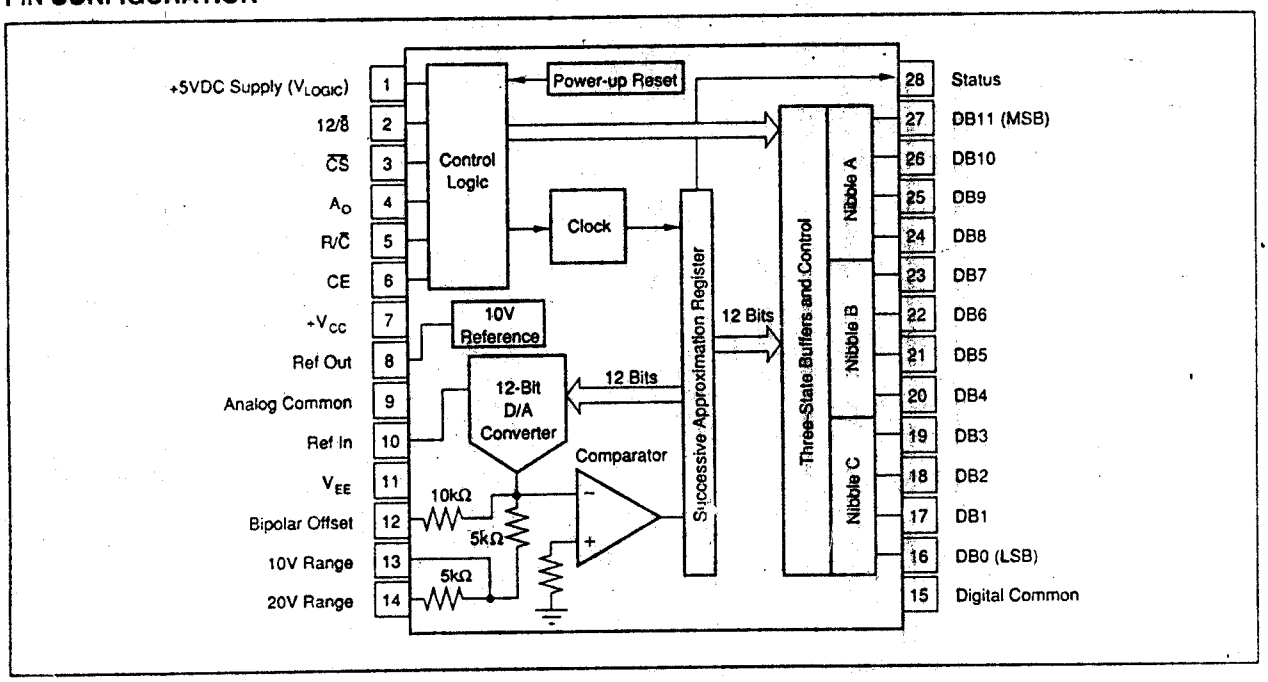
T_A = +25°C, V_{CC} = +12V or +15V, V_{EE} = −12V or −15V, V_{LOGIC} = +5V unless otherwise specified.

PARAMETERS	ADC574AJP, JH, SH			ADC574AKP, KH, TH			UNITS
	MIN	TYP	MAX	MIN	TYP	MAX	
INTERNAL REFERENCE VOLTAGE							
Voltage	+9.9	+10.0	+10.1	•	•	•	V
Source Current Available for External Loads [5]	2.0			•			mA
POWER SUPPLY REQUIREMENTS							
Voltage: V_{CC}	+11.4		+16.5	•		•	V
V_{EE}	−11.4		−16.5	•		•	V
V_{LOGIC}	+4.5		+5.5	•		•	V
Current: I_{CC}		3.5	5		•	•	mA
I_{EE}		15	20		•	•	mA
I_{LOGIC}		9	15		•	•	mA
Power Dissipation (±15V Supplies)		325	450		•	•	mW
TEMPERATURE RANGE (Ambient: T_{MIN}, T_{MAX})							
Specifications: J, K Grades	0		+75	•		•	°C
S, T Grades	−55		+125	•		•	°C
Storage	−65		+150	•		•	°C

* Same specifications as ADC574AJP, AJH, ASH.

NOTES: (1) With fixed 50Ω resistor from REF OUT to REF IN. This parameter is also adjustable to zero at ±25°C (see Optional External Full Scale and Offset Adjustments section). (2) FS in this specification table means Full Scale Range. That is, for a ±10V input range, FS means 20V; for a 0 to +10V range, FS means 10V. The term Full Scale for these specifications instead of Full-Scale Range is used to be consistent with other vendors' 574 and 574A type specifications tables. (3) Using internal reference. (4) See Controlling the ADC574A section for detailed information concerning digital timing. (5) External loading must be constant during conversion. The reference output requires no buffer amplifier with either ±12V or ±15V power supplies.

PIN CONFIGURATION



DISCUSSION OF SPECIFICATIONS

LINEARITY EPROR

Linearity error is defined as the deviation of actual code transition values from the ideal transition values. Ideal transition values lie on a line drawn through zero (or minus full scale for bipolar operation) and plus full scale. The zero value is located at an analog input value 1/2LSB before the first code transition (000_H to 001_H). The full-scale value is located at an analog value 3/2LSB beyond the last code transition (FFE_H to FFF_H) (see Figure 1).

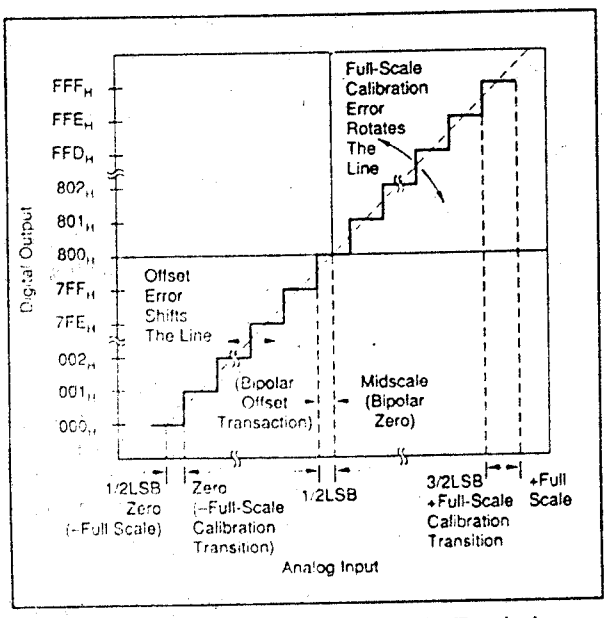


FIGURE 1. ADC574A Transfer Characteristics Terminology.

Thus, for a converter connected for biopolar operation and with a full-scale range (or span) of 20V (±10V), the zero value of –10V is 2.44mV below the first code transition (000_H to 001_H at –9.99756V) and the plus full-scale value of +10V is 7.32mV above the last code transition (FFE_H to FFF_H at +9.99268) (see Table I).

NO MISSING CODES (DIFFERENTIAL LINEARITY ERROR)

A specification which guarantees no missing codes requires that every code combination to appear in a monotonically-increasing sequence as the analog input is increased throughout the range. Thus, every input code width (quantum) must have a finite width. If an input quantum has a value of zero (a differential linearity error of –1LSB), a missing code will occur.

ADC574AKP, KN, KH and TH grades are guaranteed to have no missing codes to 12-bit resolution over their respective specification temperature ranges.

UNIPOLAR OFFSET ERROR

An ADC574A connected for unipolar operation has an analog input range of 0V to plus full scale. The first output code transition should occur at an analog input value 1/2 LSB above 0V. Unipolar offset error is defined as the deviation of the actual transition value from the ideal value. The unipolar offset temperature coefficient specifies the change of this transition value versus a change in ambient temperature.

BIPOLAR OFFSET ERROR

A/D converter specifications have historically defined bipolar offset as the first transition value above the minus full-scale value. The ADC574A specification, however, follows the terminology defined for the 574 converter several years ago. Thus, bipolar offset is located near the midscale value of 0V (bipolar zero) at the output code transition $7FF_H$ to 800_H.

Bipolar offset error for the ADC574A is defined as the deviation of the actual transition value from the ideal transition value located 1/2LSB below 0V. The bipolar offset temperature coefficient specifies the maximum change of the code transition value versus a change in ambient temperature.

FULL SCALE CALIBRATION ERROR

The last output transiton (FFE_H to FFF_H) occurs for an analog input value 3/2LSB below the nominal full-scale value. The full-scale calibration error is the deviation of the actual analog value at the last transition point from the ideal value. The full-scale calibration temperature coefficient specifies the maximum change of the code transition value versus a change in ambient temperature.

POWER SUPPLY SENSITIVITY

Electrical specifications for the ADC574A assume the application of the rated power supply voltages of +5V and ±12V or ±15V. The major effect of power supply voltage

BINARY (BIN) OUTPUT	INPUT VOLTAGE RANGE AND LSB VALUES				
Analog Input Voltage Range	Defined as:	±10V	±5V	0 to +10V	0 to +20V
One Least Significant Bit (LSB)	$\frac{FSR}{2^n}$ n = 8 n = 12	$\frac{20V}{2^n}$ 78.13mV 4.88mV	$\frac{10V}{2^n}$ 39.06mV 2.44mV	$\frac{10V}{2^n}$ 39.06mV 2.44mV	$\frac{20V}{2^n}$ 78.13mV 4.88mV
Output Transition Values FFE_H to FFF_H $7FF_H$ to 800_H	+Full-Scale Calibration Midscale Calibration (Bipolar Offset)	+10V – 3/2LSB 0 – 1/2LSB	+5 – 3/2LSB 0 – 1/2LSB	+10V – 3/2LSB +5V – 1/2LSB	+10V – 3/2LSB ±10V – 1/2LSB

TABLE I. Input Voltages, Transition Values, and LSB Values.

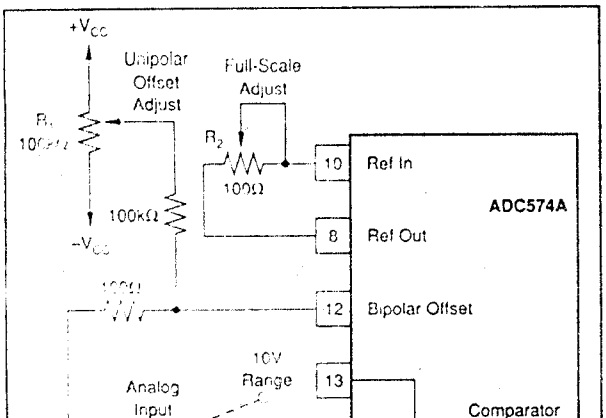


FIGURE 2. Unipolar Configuration.

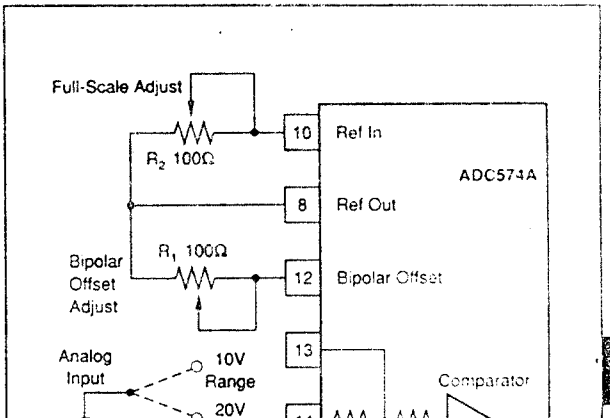


FIGURE 3. Bipolar Configuration.

for the 20V range) that causes the output code to be DB0 ON (high). Adjust potentiometer R_1 until DB0 is alternately toggling ON and OFF with all other bits OFF. Then adjust full scale by applying an input voltage of nominal full-scale value minus 3/2LSB, the value which should cause all bits to be ON. This value is +9.9963V for the 10V range and +19.9927V for the 20V range. Adjust potentiometer R_2 until bits DB1-DB11 are ON and DB0 is toggling ON and OFF.

CALIBRATION PROCEDURE—BIPOLAR RANGES

If external adjustments of full-scale and bipolar offset are not required, the potentiometers may be replaced by 50Ω, 1% metal film resistors.

If adjustments are required, connect the converter as shown in Figure 3. The calibration procedure is similar to that described above for unipolar operation, except that the offset adjustment is performed with an input voltage which is 1/2LSB above the minus full-scale value (–4.9988V for the ±5V range, –9.9976V for the ±10V range). Adjust R_1 for DB0 to toggle ON and OFF with all other bits OFF. To adjust full-scale, apply a DC input signal which is 3/2LSB below the nominal plus full-scale value (+4.9963V for ±5V range, +9.9927V for ±10V range) and adjust R_2 for DB0 to toggle ON and OFF with all other bits ON.

CONTROLLING THE ADC574A

The Burr-Brown ADC574A can be easily interfaced to most microprocessor systems and other digital systems. The microprocessor may take full control of each conversion, or the converter may operate in a stand-alone mode, controlled only by the R/$\overline{C}$ input. Full control consists of selecting an 8- or 12-bit conversion cycle, initiating the conversion, and reading the output data when ready—choosing either 12 bits all at once, or 8 bits followed by 4 bits in a left-justified format. The five control inputs (12/$\overline{8}$, $\overline{CS}$, A_0, R/$\overline{C}$, and CE) are all TTL/CMOS-compatible. The functions of the control inputs are described in Table II. The control function truth table is listed in Table III.

PIN DESIGNATION	DEFINITION	FUNCTION
CE (Pin 6)	Chip Enable (active high)	Must be high ("1") to either initiate a conversion or read output data. 0-1 edge may be used to initiate a conversion.
$\overline{CS}$ (Pin 3)	Chip Select (active low)	Must be low ("0") to either initiate a conversion or read output data. 1-0 edge may be used to initiate a conversion.
R/$\overline{C}$ (Pin 5)	Read/Convert ("1" = read) ("0" = convert)	Must be low ("0") to initiate either 8- or 12-bit conversions. 1-0 edge may be used to initiate a conversion. Must be high ("1") to read output data. 0-1 edge may be used to initiate a read operation.
A_0 (Pin 4)	Byte Address Short Cycle	In the start-convert mode, A_0 selects 8-bit (A_0 = "1") or 12-bit (A_0 = "0") conversion mode. When reading output data in two 8-bit bytes, A_0 = "0" accesses 8 MSBs (high byte) and A_0 = "1" accesses 4 LSBs and trailing "0s" (low byte).
12/$\overline{8}$ (Pin 2)	Data Mode Select ("1" = 12 bits) ("0" = 8 bits)	When reading output data, 12/$\overline{8}$ = "1" enables all 12 output bits simultaneously. 12/$\overline{8}$ = "0" will enable the MSBs or LSBs as determined by the A_0 line.

TABLE II. ADC574A Control Line Functions.

SYMBOL	PARAMETER	MIN	TYP	MAX	UNITS
Convert Mode					
t_{HEC}	STS Delay from CE		60	200	ns
t_{WCE}	CE Pulse Width	50	30		ns
t_{SSC}	$\overline{CS}$ to CE Setup time	50	20		ns
t_{HSC}	$\overline{CS}$ low during CE high	50	20		ns
t_{SRC}	R/$\overline{C}$ to CE setup	50	0		ns
t_{HRC}	R/$\overline{C}$ low during CE high	50	20		ns
t_{SAC}	A_0 to CE setup	0			ns
t_{HAC}	A_0 valid during CE high	50	20		ns
t_C	Conversion time, 12-bit cycle	15	20	25	µs
	8-bit cycle	10	13	17	µs
Read Mode					
t_{DD}	Access time from CE		75	150	ns
t_{HD}	Data valid after CE low	25	35		ns
t_{HL}	Output float delay		100	150	ns
t_{SSR}	$\overline{CS}$ to CE setup	50	0		ns
t_{SRR}	R/$\overline{C}$ to CE setup	0			ns
t_{SAR}	A_0 to CE setup	50	25		ns
t_{HSR}	$\overline{CS}$ valid after CE low	0			ns
t_{HRR}	R/$\overline{C}$ high after CE low	0			ns
t_{HAR}	A_0 valid after CE low	50			ns
t_{HS}	STS delay after data valid	300	400	1000	ns

NOTE: Specifications are at +25°C and measured at 50% level of transitions.

TABLE V. Timing Specifications.

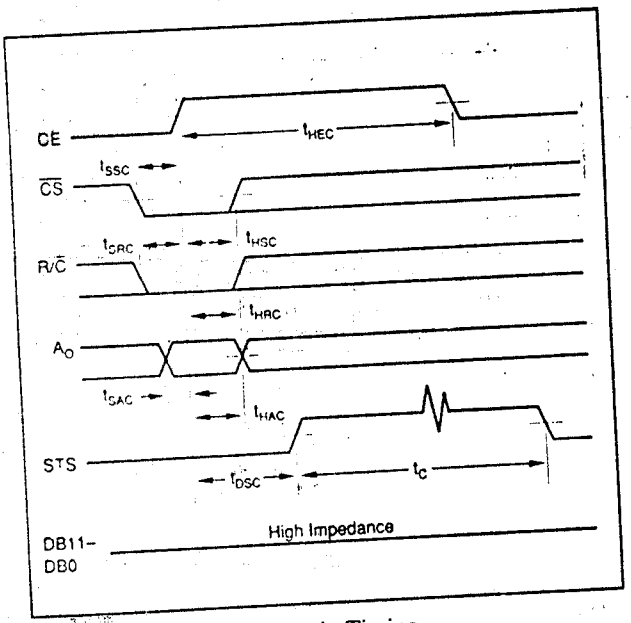


FIGURE 6. Conversion Cycle Timing.

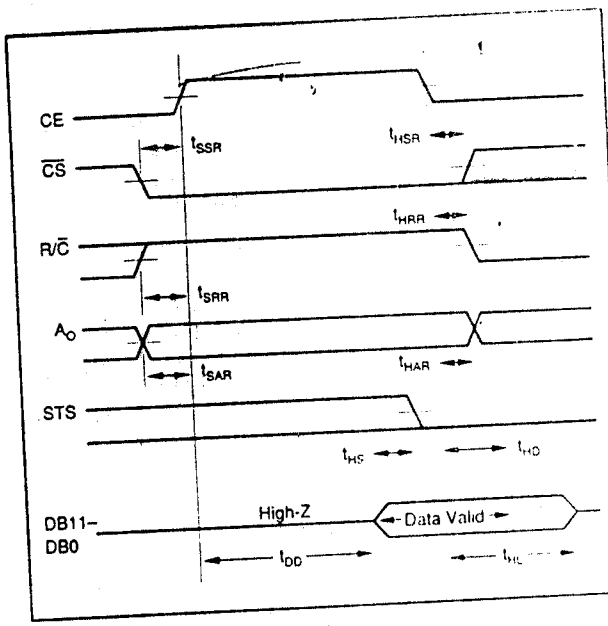


FIGURE 7. Read Cycle Timing.

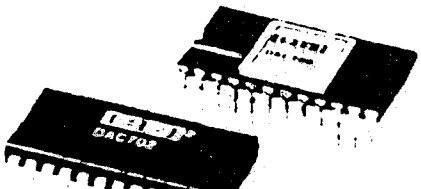
READING OUTPUT DATA

After conversion is initiated, the output data buffers remain in a high-impedance state until the following four logic conditions are simultaneously met: R/$\overline{C}$ high, STATUS low, CE high, and $\overline{CS}$ low. Upon satisfaction of these conditions the data lines are enabled according to the state of inputs 12/$\overline{8}$ and A_0. See Figure 7 and Table V for timing relationships and specifications.

In most applications the 12/$\overline{8}$ input will be hard-wired in either the high or low condition, although it is fully TTL- and CMOS-compatible and may be actively driven if desired. When 12/$\overline{8}$ is high, all 12 output lines (DB0-DB1 are enabled simultaneously for full data word transfer t 12-bit or 16-bit bus. In this situation the A_0 state is ignor

When 12/$\overline{8}$ is low, the data is presented in the form of t 8-bit bytes, with selection of the byte of interest acc plished by the state of A_0 during the read cycle. Connect of the ADC574A to an 8-bit bus for transfer of left justi data is illustrated in Figure 8. The A_0 input is usually dri by the least significant bit of the address bus, allow storage of the output data word in two consecutive mem locations.

BURR-BROWN®



DAC700/702
DAC701/703

Monolithic 16-Bit DIGITAL-TO-ANALOG CONVERTERS

FEATURES

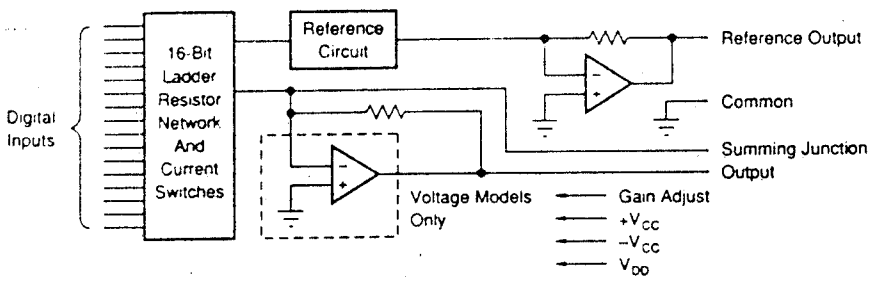
- V_{OUT} AND I_{OUT} MODELS
- HIGH ACCURACY:
  Linearity Error ±0.0015% of FSR max
  Differential Linearity Error ±0.003% of FSR max
- MONOTONIC (at 15 bits) OVER FULL SPECIFICATION TEMPERATURE RANGE
- PIN-COMPATIBLE WITH DAC70, DAC71, DAC72
- DUAL-IN-LINE PLASTIC AND HERMETIC CERAMIC AND SOIC

DESCRIPTION

The DAC70X family comprise of complete 16-bit digital-to-analog converters that includes a precision buried-zener voltage reference and a low-noise, fast-settling output operational amplifier (voltage output models), all on one small monolithic chip. A combination of current-switch design techniques accomplishes not only 15-bit monotonicity over the entire specified temperature range, but also a maximum end-point linearity error of ±0.0015% of full-scale range. Total full-scale gain drift is limited to ±10ppm/°C maximum (LH and CH grades).

Digital inputs are complementary binary coded and are TTL-, LSTTL-, 54/74C- and 54/74HC-compatible over the entire temperature range. Outputs of 0 to +10V, ±10V, 0 to –2mA, and ±1mA are available.

These D/A converters are packaged in hermetic 24-pin ceramic side-brazed or molded plastic. The DIP-packaged parts are pin-compatible with the voltage and current output DAC71 and DAC72 model families. The DAC700 and DAC702 are also pin-compatible with the DAC70 model family. In addition, the DAC703 is offered in a 24-pin SOIC package for surface mount applications.



CTRICAL (CONT)

EL	DAC702/703J, D			DAC700/701/702/703K			DAC700/701/702/703B, S			DAC700/701/702/703L, C			
AMETER	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	UNITS
ERENCE VOLTAGE													
ge		+6.3		+6.0	+6.3	+6.6	+6.24	+6.3	+6.36	•	•	•	V
e Current Available													
External Loads		+2.5		+1.5	•		•	•		•	•		mA
erature Coefficient		±10			•	±25		•	±15		•	•	ppm/°C
Circuit to Common													
ation		Indefinite			•			•			•		
ER SUPPLY REQUIREMENTS													
ge: $+V_{CC}$	13.5	15	16.5	•	•	•	•	•	•	•	•	•	V
$-V_{CC}$	13.5	15	16.5	•	•	•	•	•	•	•	•	•	V
V_{DD}	+4.5	+5	+16.5	•	•	•	•	•	•	•	•	•	V
nt (No Load):													
700/702													
Models)													
V_{CC}		+10	+25		•	•		•	•		•	•	mA
V_{CC}		−13	−25		•	•		•	•		•	•	mA
DD		+4	+8		•	•		•	•		•	•	mA
701/703													
Models)													
V_{CC}		+16	+30		•	•		•	•		•	•	mA
V_{CC}		−18	−30		•	•		•	•		•	•	mA
DD		+4	+8		•	•		•	•		•	•	mA
er Dissipation:													
= +5.0V) [11]													
AC700/702		365			•	790		•	630		•	•	mW
AC701/703		530			•	940		•	780		•	•	mW
er Supply Rejection:													
CC		±0.0015	±0.006		•	•		•	±0.003		•	•	% of FSR/%V_{CC}
CC		±0.0015	±0.006		•	•		•	±0.003		•	•	% of FSR/%V_{CC}
EE		±0.0001	±0.001		•	•		•	•		•	•	% of FSR/%V_{EE}
PERATURE RANGE													
ification:													
C Grades							−25		+85	•		•	°C
Grades							−55		+125				°C
K, L Grades	0		+70	•		•				0		+70	°C
ge Ceramic				−60		+150	•		•	•		•	°C
Plastic, SOIC	−60		+100	•		•							°C

ification same as model to the left.

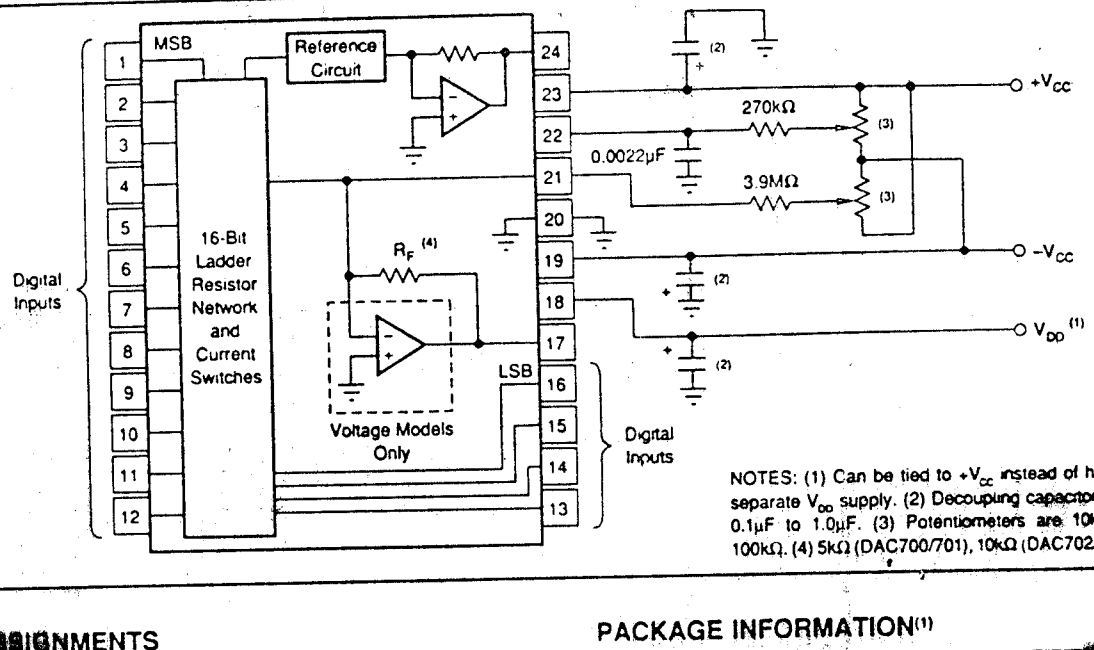
S. (1) Digital inputs are TTL, LSTTL, 54/74C, 54/74HC, and 54/74HTC compatible over the operating voltage range of V_{DD} = +5V to +15V and over the specified temperature The input switching threshold remains at the TTL threshold of 1.4V over the supply range of V_{DD} = +5V to +15V. As logic "0" and logic "1" inputs vary over 0V to +0.8V .4V to +10V respectively, the change in the D/A converter output voltage will not exceed ±0.0015% of FSR for the LH and CH grades, ±0.003% of FSR for the BH grade 006% of FSR for the KG grade. (2) DAC700 and DAC702 (current-output models) are specified and tested with an external output operational amplifier connected using ernal feedback resistor in all parameters except settling time. (3) FSR means full-scale range and is 20V for the ±10V range (DAC703), 10V for the 0 to +10V range 01). FSR is 2mA for the ±1mA range (DAC702) and the 0 to +2mA range (DAC700). (4) ±0.0015% of full-scale range is equivalent to 1LSB in 15-bit resolution. ±0.003% scale range is equivalent to 1LSB in 14-bit resolution. ±0.006% of full-scale range is equivalent to 1LSB in 13-bit resolution. (5) Adjustable to zero with external trim ometer. Adjusting the gain potentiometer rotates the transfer function around the zero point. (6) Error at input code $FFFF_H$ for DAC700 and DAC701, $7FFF_H$ for DAC702 AC703. (7) With gain and zero errors adjusted to zero at +25°C. (8) Maximum represents the 3σ limit. Not 100% tested for this parameter. (9) At the major carry, $7FFF_H$ 0_H and 8000_H to $7FFF_H$. (10) Tolerance on output impedance and output current is ±30%. (11) Power dissipation is an additional 40mW when V_{DD} is operated at +15V.

OLUTE MAXIMUM RATINGS

to Common	0V, +18V
to Common	0V, −18V
o Common	0V, +18V
al Data Inputs to Common	−1V, +18V
rence Out to Common	Indefinite Short to Common
nal Voltage Applied to R_F (DAC700/702)	±18V
nal Voltage Applied to D/A Output (DAC701/703)	−5V to +5V
V_{OUT} (DAC701/703)	Indefinite Short to Common
Power Dissipation	1W
Storage Temperature	−60°C to +150°C
Lead Temperature (soldering, 10s)	300°C

NOTE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. Exposure to absolute maximum conditions for extended periods may affect device reliability.

CONNECTION DIAGRAMS



PIN ASSIGNMENTS

PIN #	ALL PACKAGES	
	DAC700/702	DAC701/703
1	Bit 1 (MSB)	Bit 1 (MSB)
2	Bit 2	Bit 2
3	Bit 3	Bit 3
4	Bit 4	Bit 4
5	Bit 5	Bit 5
6	Bit 6	Bit 6
7	Bit 7	Bit 7
8	Bit 8	Bit 8
9	Bit 9	Bit 9
10	Bit 10	Bit 10
11	Bit 11	Bit 11
12	Bit 12	Bit 12
13	Bit 13	Bit 13
14	Bit 14	Bit 14
15	Bit 15	Bit 15
16	Bit 16 (LSB)	Bit 16 (LSB)
17	$R_{FEEDBACK}$	V_{OUT}
18	V_{DD}	V_{DD}
19	$-V_{CC}$	$-V_{CC}$
20	Common	Common
21	I_{OUT}	Summing Junction (Zero Adjust)
22	Gain Adjust	Gain Adjust
23	$+V_{CC}$	$+V_{CC}$
24	+6.3V Reference Output	+6.3V Reference Output

PACKAGE INFORMATION[1]

MODEL	PACKAGE	PACKAGE DRAWING NUMBER
DAC702JP	24-Pin Plastic DIP	167
DAC703JP	24-Pin Plastic DIP	167
DAC702KP	24-Pin Plastic DIP	167
DAC703KP	24-Pin Plastic DIP	167
DAC700KH	24-Pin Ceramic DIP	165
DAC701KH	24-Pin Ceramic DIP	165
DAC702KH	24-Pin Ceramic DIP	165
DAC703KH	24-Pin Ceramic DIP	165
DAC700BH	24-Pin Ceramic DIP	165
DAC701BH	24-Pin Ceramic DIP	165
DAC702BH	24-Pin Ceramic DIP	165
DAC703BH	24-Pin Ceramic DIP	165
DAC700LH	24-Pin Ceramic DIP	165
DAC701LH	24-Pin Ceramic DIP	165
DAC702LH	24-Pin Ceramic DIP	165
DAC703LH	24-Pin Ceramic DIP	165
DAC700CH	24-Pin Ceramic DIP	165
DAC701CH	24-Pin Ceramic DIP	165
DAC702CH	24-Pin Ceramic DIP	165
DAC703CH	24-Pin Ceramic DIP	165
DAC702SH	24-Pin Ceramic DIP	165
DAC703SH	24-Pin Ceramic DIP	165
DAC703JU	24-Pin SOIC	239
DAC703KU	24-Pin SOIC	239

NOTE: (1) For detailed drawing and dimension table, please see end of data sheet, or Appendix D of Burr-Brown IC Data Book.

Current Output

Settling times are specified to ±0.003% of FSR for a full-scale range change for two output load conditions: one for 10Ω to 100Ω and one for 1000Ω. It is specified this way because the output RC time constant becomes the dominant factor in determining settling time for large resistive loads.

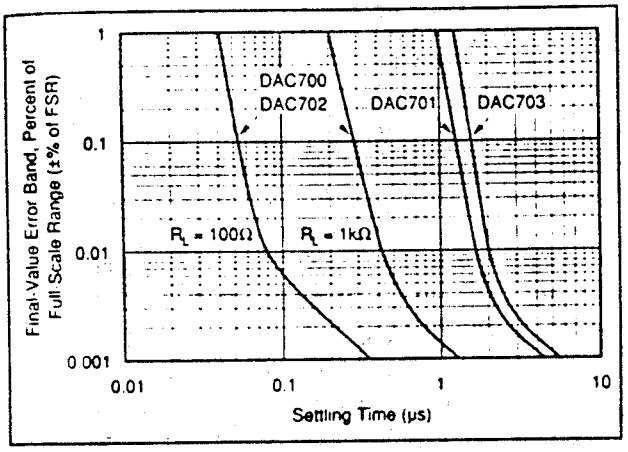


FIGURE 1. Final-Value Error Band vs Full-Scale Range Settling Time.

COMPLIANCE VOLTAGE

Compliance voltage applies only to current output models. It is the maximum voltage swing allowed on the output current pin while still being able to maintain specified accuracy.

POWER SUPPLY SENSITIVITY

Power supply sensitivity is a measure of the effect of a change in a power supply voltage on the D/A converter output. It is defined as a percent of FSR change in the output per percent of change in either the positive supply ($+V_{CC}$), negative supply ($-V_{CC}$) or logic supply (V_{DD}) about the nominal power supply voltages (see Figure 2).

It is specified for DC or low frequency changes. The typical performance curve in Figure 2 shows the effect of high frequency changes in power supply voltages.

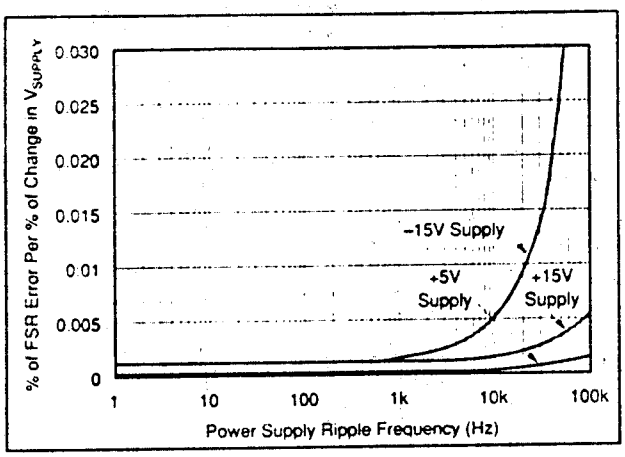


FIGURE 2. Power Supply Rejection vs Power Supply Ripple

REFERENCE SUPPLY

All models have an internal low-noise +6.3V reference voltage derived from an on-chip buried zener diode. This reference voltage, available to the user, has a tolerance of ±5% (KH models) and ±1% (BH models). A minimum of 1.5mA is available for external loads. Since the output impedance of the reference output is typically 1W, the external load should remain constant.

If a varying load is to be driven by the reference supply, an external buffer amplifier is recommended to drive the load in order to isolate the bipolar offset (connected internally to the reference) from load variations.

OPERATING INSTRUCTIONS

POWER SUPPLY CONNECTIONS

For optimum performance and noise rejection, power supply decoupling capacitors should be added as shown in the Connection Diagram. 1µF tantalum capacitors should be located close to the D/A converter.

EXTERNAL ZERO AND GAIN ADJUSTMENT

Zero and gain may be trimmed by installing external zero and gain potentiometers. Connect these potentiometers as shown in the Connection Diagram and adjust as described below. TCR of the potentiometers should be 100ppm/°C or less. The 3.9MΩ and 270kΩ resistors (±20% carbon or better) should be located close to the D/A converter to prevent noise pickup. If it is not convenient to use these high-value resistors, an equivalent "T" network, as shown in Figure 3, may be substituted in place of the 3.9MΩ part. A 0.001µF to 0.01µF ceramic capacitor should be connected from Gain Adjust to Common to prevent noise pickup. Refer to Figures 4 and 5 for the relationship of zero and gain adjustments to unipolar and bipolar D/A converters.

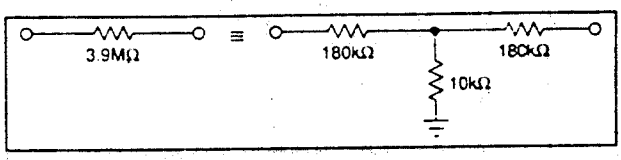
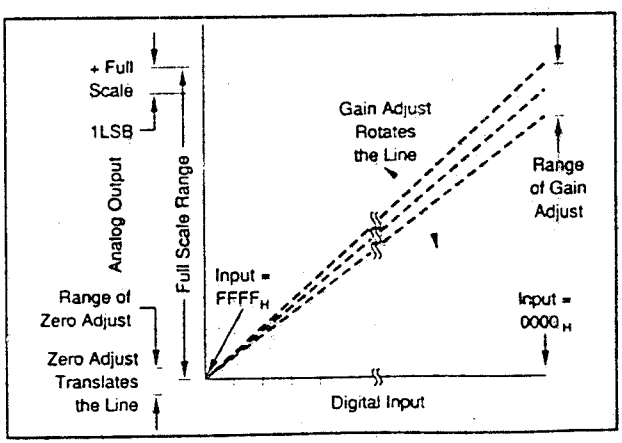


FIGURE 3. Equivalent Resistances.



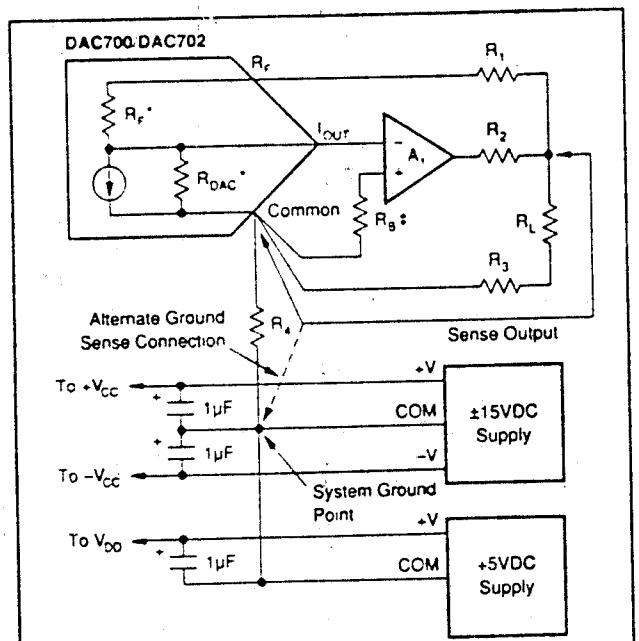
VOLTAGE OUTPUT MODELS							
		ANALOG OUTPUT					
		DAC701 UNIPOLAR			DAC703 BIPOLAR		
DIGITAL INPUT CODE		16-BIT	15-BIT	14-BIT	16-BIT	15-BIT	14-BIT
1LSB	(μV)	153	305	610	305	610	1224
0000$_H$	(V)	+9.99985	+9.99969	+9.99939	+9.99960	+9.99939	+9.99878
FFFF$_H$	(V)	0	0	0	−10.0000	−10.0000	−10.0000

ANALOG OUTPUT MODELS							
		ANALOG OUTPUT					
		DAC700 UNIPOLAR			DAC702 BIPOLAR		
DIGITAL INPUT CODE		16-BIT	15-BIT	14-BIT	16-BIT	15-BIT	14-BIT
1LSB	(μA)	0.031	0.061	0.122	0.031	0.061	0.122
0000$_H$	(mA)	−1.99997	−1.99994	−1.99988	−0.99997	−0.99994	−0.99988
FFFF$_H$	(mA)	0	0	0	+1.00000	+1.00000	+1.00000

TABLE II. Digital Input and Analog Output Relationships.

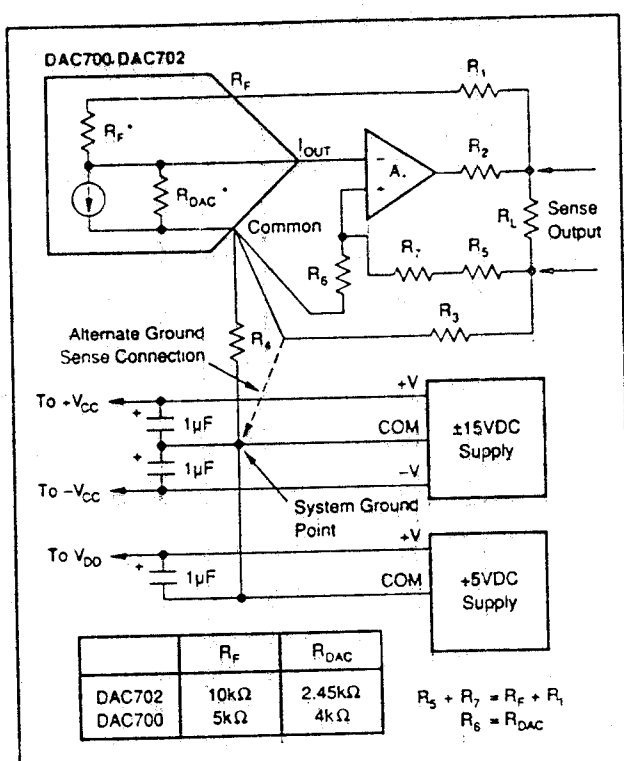
If the output cannot be sensed at Common or the sytem ground point as mentioned above, the differential output circuit shown in Figure 8 is recommended. In this circuit the output voltage is sensed at the load common and not at the D/A converter common as in the previous circuits. The value of R_4 and R_3 must be adjusted for maximum common-mode rejection at R_L. Note that if R_3 is negligible, the circuit of Figure 8 can be reduced to the one shown in Figure 7. Again the effect of R_4 is negligible.

The D/A converter and the wiring to its connectors should be located to provide optimum isolation from sources of RFI and EMI. The key concept in elimination of RF radiation or pickup is loop area; therefore, signal leads and their return conductors should be kept close together. This reduces the external magnetic field along with any radiation. Also, if a single lead and its return conductor are wired close together, they present a small flux-capture cross section for any external field. This reduces radiation pickup in the circuit.



	R_B	R_F	R_{DAC}
DAC702	2.45kΩ	10kΩ	2.45kΩ
DAC700	4kΩ	5kΩ	4kΩ

‡ R_B should be equal to the output impedance at the current output to compensate for the bias current drift of A_1. Use standard 10%, 1/4W carbon composition or equivalent resistors.



	R_F	R_{DAC}
DAC702	10kΩ	2.45kΩ
DAC700	5kΩ	4kΩ

$R_5 + R_7 = R_F + R_1$
$R_6 = R_{DAC}$

FIGURE 8. Differential Sensing Output Op Amp Configura

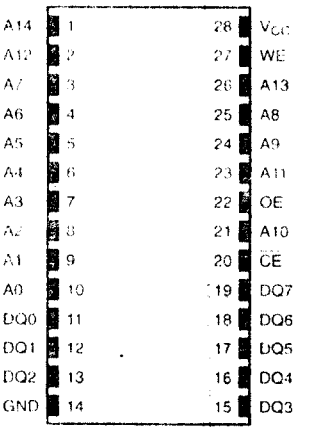
DALLAS SEMICONDUCTOR

DS1230Y/AB
256K Nonvolatile SRAM

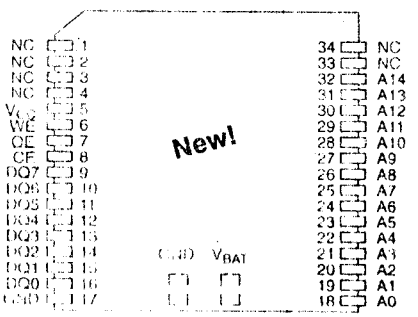
FEATURES

- 10 years minimum data retention in the absence of external power
- Data is automatically protected during power loss
- Replaces 32K x 8 volatile static RAM, EEPROM or Flash memory
- Unlimited write cycles
- Low–power CMOS
- Read and write access times as fast as 70 ns
- Lithium energy source is electrically disconnected to retain freshness until power is applied for the first time
- Full ±10% V_{CC} operating range (DS1230Y)
- Optional ±5% V_{CC} operating range (DS1230AB)
- Optional industrial temperature range of –40°C to +85°C, designated IND
- JEDEC standard 28–pin DIP package
- New PowerCap Module (PCM) package
  - Directly surface–mountable module
  - Replaceable snap–on PowerCap provides lithium backup battery
  - Standardized pinout for all nonvolatile SRAM products
  - Detachment feature on PowerCap allows easy removal using a regular screwdriver

PIN ASSIGNMENT



28–PIN ENCAPSULATED PACKAGE
740 MIL EXTENDED



34–PIN POWERCAP MODULE (PCM)
(USES DS9034PC POWERCAP)

PIN DESCRIPTION

A0 – A14	–	Address Inputs
DQ0 – DQ7	–	Data In/Data Out
$\overline{CE}$	–	Chip Enable
$\overline{WE}$	–	Write Enable
$\overline{OE}$	–	Output Enable
V_{CC}	–	Power (+5V)
GND	–	Ground
NC	–	No Connect

IDUCTOR
DATA

UAD SINGLE SUPPLY COMPARATORS

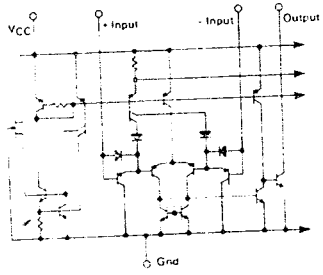
nparators are designed for use in level detection, low-
g and memory applications in Consumer Automotive
al electronic applications.

- Split Supply Operation
- t Bias Current — 25 nA (Typ)
- t Offset Current — ±5.0 nA (Typ)
- t Offset Voltage — ±1.0 mV (Typ LM139A Series)
- mmon-Mode Voltage Range to Gnd
- put Saturation Voltage — 130 mV (Typ) @ 4.0 mA
- CMOS Compatible

TINGS

Rating	Symbol	Value	Unit
oltage LM139, A/LM239, A LM339A/LM2901 MC3302	V_{CC}	+ 36 or ± 18 + 30 or ± 15	Vdc
l Voltage Range A/LM239, A LM339, A/LM2901 MC3302	V_{IDR}	36 30	Vdc
Mode Voltage Range	V_{ICR}	− 0.3 to V_{CC}	Vdc
rcuit to Gnd (Note 1)	I_{SC}	Continuous	
V_{in} < − 0.3 Vdc) (Note 2)	I_{in}	50	mA
on @ T_A = 25°C age ve 25°C ge ve 25°C	P_D	1.0 8.0 1.0 8.0	Watts mW/°C Watts mW/°C
erature letal Package ge	T_J	175 150	°C
ient Temperature Range	T_A	− 55 to + 125 − 25 to + 85 − 40 to + 85 − 40 to + 105 0 to + 70	°C
erature Range	T_{stg}	− 65 to + 150	°C

1 — CIRCUIT SCHEMATIC (Diagram shown is for 1 comparator)



LM139, A
LM239, A LM2901
LM339, A MC3302

QUAD COMPARATORS

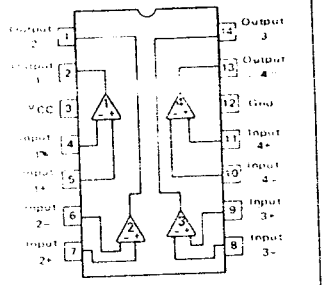
SILICON MONOLITHIC
INTEGRATED CIRCUIT

N, P SUFFIX
PLASTIC PACKAGE
CASE 646

J, L SUFFIX
CERAMIC PACKAGE
CASE 632

D SUFFIX
PLASTIC PACKAGE
CASE 751A
(SO-14)

PIN CONNECTIONS



(Top View)

ORDERING INFORMATION

Device	Temperature Range	Package
LM139J, AJ	− 55°C to + 125°C	Ceramic DIP
LM239D, AD LM239J, AJ LM239N, AN	− 25°C to + 85°C	SO-14 Ceramic DIP Plastic DIP
LM339D, AD LM339J, AJ LM339N, AN	0°C to + 70°C	SO-14 Ceramic DIP Plastic DIP
LM2901D LM2901N	− 40°C to + 105°C	SO-14 Plastic DIP
MC3302L MC3302P	− 40°C to + 85°C	Ceramic DIP Plastic DIP

ELECTRICAL CHARACTERISTICS (V_{CC} = +5.0 Vdc, T_A = 25°C, unless otherwise noted)

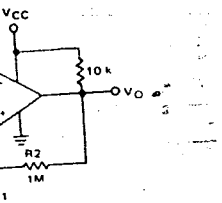
Characteristic	Symbol	LM139A Min	LM139A Typ	LM139A Max	LM239A/339A Min	LM239A/339A Typ	LM239A/339A Max	LM139 Min	LM139 Typ	LM139 Max	LM239/339 Min	LM239/339 Typ	LM239/339 Max	LM2901 Min	LM2901 Typ	LM2901 Max	MC3302 Min	MC3302 Typ	MC3302 Max	Unit
Input Offset Voltage (Note 4) V_O @ 1.4 Vdc, R_S = 0 Ω, 5.0 Vdc ≤ V_{CC} ≤ 30 Vdc, 0 ≤ V_{ICR} ≤ V_{CC} − 1.5 Vdc	V_{IO}		±1.0	±2.0		±1.0	±2.0		1.3	±5.0		±2.0	±5.0		±2.0	±7.0		±3.0	±20	mVdc
Input Bias Current (Notes 4, 5) (Output in Linear Range)	I_{IB}		25	100		25	250		25	100		25	250		25	250		25	500	nA
Input Offset Current (Note 4) V_O @ 1.4 Vdc	I_{IO}		±3.0	±25		±5.0	±50		±3.0	±25		±5.0	±50		±5.0	±50		±3.0	±100	nA
Input Common-Mode Voltage Range V_O @ 1.4 Vdc, 5.0 Vdc ≤ V_{CC} ≤ 30 Vdc	V_{ICR}	0		V_{CC} − 1.5	0		V_{CC} − 1.5	0		V_{CC} − 1.5	0		V_{CC} − 1.5	0		V_{CC} − 1.5	0		V_{CC} − 1.5	Vdc
Supply Current R_L = ∞ (For All Comparators) V_{CC} = 30 Vdc V_{CC} = 5.0 Vdc	I_{CC}		0.8 0.8	2.0 2.0		0.8 0.8	2.0 2.0		0.8 0.8	2.0 2.0		0.8 0.8	2.0 2.0		0.8 0.8	2.0 2.0		0.8 0.8	2.0 2.0	mA
Voltage Gain R_L ≥ 15 kΩ, V_{CC} = 15 Vdc	A_V	50	200		50	200			200			200			100			30		V/mV
Large Signal Response Time V_I = TTL Logic Swing, V_{ref} = 1.4 Vdc, V_{RL} = 5.0 Vdc, R_L = 5.1 kΩ			300			300			300			300			300			300		ns
Response Time (Note 6) V_{RL} = 5.0 Vdc, R_L = 5.1 kΩ			1.3			1.3			1.3			1.3			1.3			1.3		μs
Output Sink Current $V_I(-) \geq 1.0$ Vdc, $V_I(+) = 0$, $V_O \leq 1.5$ Vdc	I_{sink}	6.0	16		6.0	16		6.0	16		6.0	16		6.0	16		6.0	16		mA
Saturation Voltage $V_I(-) \geq 1.0$ Vdc, $V_I(+) = 0$, $I_{sink} \leq 4.0$ mA	V_{sat}		130	400		130	400		130	400		130	400		130	400		130	600	mV
Output Leakage Current $V_I(+) \geq 1.0$ Vdc, $V_I(-) = 0$, V_O = 5.0 Vdc	I_{OL}		0.1			0.1			0.1			0.1			0.1			0.1		nA

PERFORMANCE CHARACTERISTICS (V_{CC} = +5.0 Vdc, T_A = T_{low} to T_{high} [Note 3])

Characteristic	Symbol	LM139A Min	LM139A Typ	LM139A Max	LM239A/339A Min	LM239A/339A Typ	LM239A/339A Max	LM139 Min	LM139 Typ	LM139 Max	LM239/339 Min	LM239/339 Typ	LM239/339 Max	LM2901 Min	LM2901 Typ	LM2901 Max	MC3302 Min	MC3302 Typ	MC3302 Max	Unit
Input Offset Voltage (Note 4) V_O @ 1.4 Vdc, R_S = 0 Ω, 5.0 Vdc ≤ V_{CC} ≤ 30 Vdc, 0 ≤ V_{ICR} ≤ V_{CC} − 1.5 Vdc	V_{IO}						±4.0			±9.0			±9.0			±15			±40	mVdc
Input Bias Current (Notes 4, 5) (Output in Linear Range)	I_{IB}						400		300			400			500			1000		nA
Input Offset Current (Note 4) V_O @ 1.4 Vdc	I_{IO}			±100			±150			±100			±150			±200			±300	nA
Input Common-Mode Voltage Range V_O @ 1.4 Vdc, 5.0 Vdc ≤ V_{CC} ≤ 30 Vdc	V_{ICR}	0		V_{CC} − 2.0	0		V_{CC} − 2.0	0		V_{CC} − 2.0	0		V_{CC} − 2.0	0		V_{CC} − 2.0	0		V_{CC} − 2.0	V
Saturation Voltage $V_I(-) \geq 1.0$ Vdc, $V_I(+) = 0$, $I_{sink} \leq 4.0$ mA	V_{sat}			700			700			700			700			700			700	mV
Output Leakage Current $V_I(+) \geq 1.0$ Vdc, $V_I(-) = 0$, V_O = 30 Vdc	I_{OL}		1.0			1.0			1.0			1.0			1.0			1.0		μA
Differential Input Voltage All $V_I \geq 0$ Vdc	V_{ID}		V_{CC}			V_{CC}			V_{CC}			V_{CC}			V_{CC}			V_{CC}		Vdc

NOTES:
1. The maximum output current may be as high as 20 mA, independent of the magnitude of V_{CC}. Output short circuits to V_{CC} can cause excessive heating and eventual destruction.
2. This magnitude of input current will only occur if the leads are driven more negative than ground or the negative supply voltage. This is due to the input PNP collector comparators to go to the V_{CC} voltage level (or ground if overdriven in the negative direction). The maximum input current is independent of the magnitude of V_{CC}. In the single supply mode, the inputs can be driven negative by as much as the max rating and normal output states will recover when the inputs become > 0 Vdc. Remember that an input is driven negative. This will not destroy the device when limited to the max rating.
3. LM139/139A — T_{low} = − 55°C, T_{high} = + 125°C
   LM239/239A — T_{low} = − 25°C, T_{high} = + 85°C
   LM339/339A — T_{low} = 0°C, T_{high} = + 70°C
   MC3302 — T_{low} = − 40°C, T_{high} = + 85°C
   LM2901 — T_{low} = − 40°C, T_{high} = + 105°C
4. At the output switch point, V_O = 1.4 Vdc, R_S = 100 Ω, 5.0 Vdc ≤ V_{CC} ≤ 30 Vdc, with the inputs over the full common-mode range (0 Vdc to V_{CC} − 1.5 Vdc).
5. The input current flows out of the inputs due to the PNP input stage. This current is virtually constant, independent of the output state.
6. The response time specified is for a 100 mV input step with 5.0 mV overdrive. For larger signals, 300 ns is typical.

ING COMPARATOR WITH HYSTERESIS



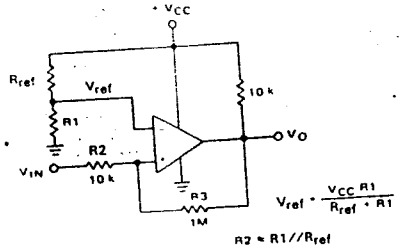
$$V_{ref} \approx \frac{V_{CC}\,R1}{R_{ref} + R1}$$

$R3 \approx R1 // R_{ref}//R2$

$$V_H \approx \frac{R1 // R_{ref}}{R1//R_{ref} + R2}\,(V_{Omax} - V_{Omin})$$

$R2 >> R_{ref}//R1$

FIGURE 3 — NON-INVERTING COMPARATOR WITH HYSTERESIS



$$V_{ref} \approx \frac{V_{CC}\,R1}{R_{ref} + R1}$$

$R2 \approx R1//R_{ref}$

Amount of Hysteresis V_H

$$V_H \approx \frac{R2}{R2 + R3}\,(V_{Omax} - V_{Omin})$$

TYPICAL CHARACTERISTICS

(V_{CC} = +15 Vdc, T_A = +25°C (each comparator) unless otherwise noted.)

ORMALIZED INPUT OFFSET VOLTAGE

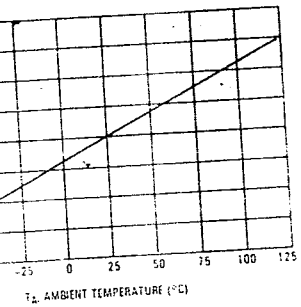


FIGURE 5 — INPUT BIAS CURRENT

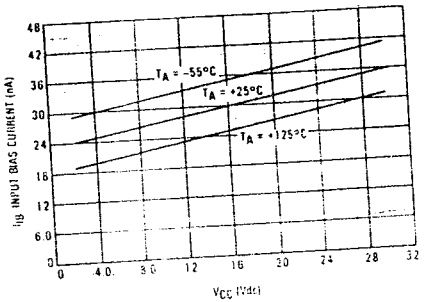


FIGURE 6 — OUTPUT SINK CURRENT versus OUTPUT SATURATION VOLTAGE

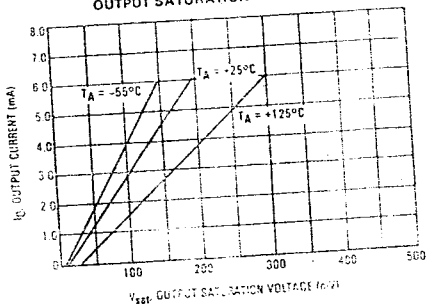
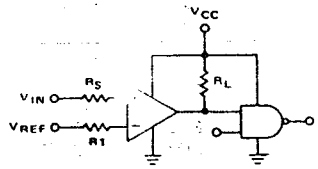


FIGURE 7 — DRIVING LOGIC

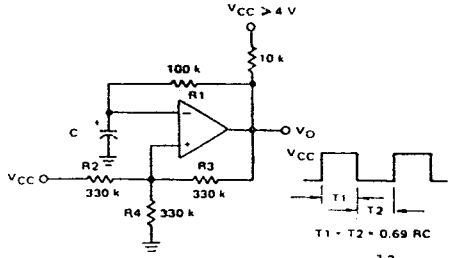


R_S = Source Resistance

$R1 \approx R_S$

LOGIC	DEVICE	V_{CC} Volts	R_L kΩ
CMOS	1/4 MC14001	+15	100
TTL	1/4 MC7400	+5	10

FIGURE 8 — SQUAREWAVE OSCILLATOR



T1 = T2 = 0.69 RC

$f \approx \frac{7.2}{C(\mu F)}$

R2 = R3 = R4

R1 ≈ R2//R3//R4

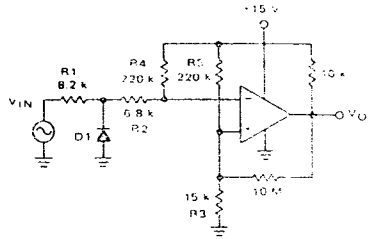
APPLICATIONS INFORMATION

These quad comparators feature high gain, wide bandwidth characteristics. This gives the device oscillation tendencies if the outputs are capacitively coupled to the inputs via stray capacitance. This oscillation manifests itself during output transitions (V_{OL} to V_{OH}). To alleviate this situation input resistors < 10 kΩ should be used. The addition of positive feedback (<10 mV) is also recommended.

It is good design practice to ground all unused input pins.

Differential input voltages may be larger than supply voltages without damaging the comparator's inputs. Voltages more negative than −300 mV should not be used.

FIGURE 9 — ZERO CROSSING DETECTOR (Single Supply)

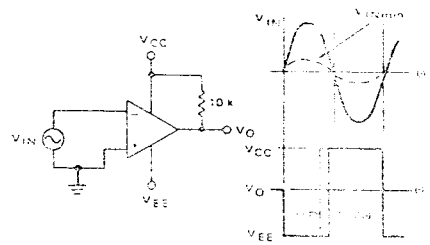


D1 prevents input from going negative by more than 0.6 V

R1 = R2 = R3

$R3 < \frac{R5}{10}$ for small error in zero crossing

FIGURE 10 — ZERO CROSSING DETECTOR (Split Supplies)



A
ONDUCTOR
L DATA

LM124, LM224, LM324, LM324A LM2902

QUAD DIFFERENTIAL INPUT OPERATIONAL AMPLIFIERS

SILICON MONOLITHIC INTEGRATED CIRCUIT

J SUFFIX
CERAMIC PACKAGE
CASE 632

N SUFFIX
PLASTIC PACKAGE
CASE 646
(LM224, LM324, LM2902 Only)

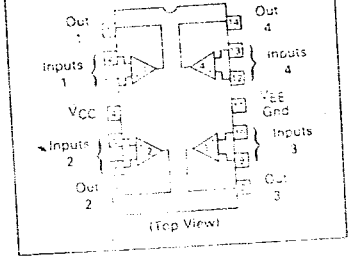
D SUFFIX
PLASTIC PACKAGE
CASE 751A
(SO-14)

UAD LOW POWER OPERATIONAL AMPLIFIERS

LM124 Series are low-cost, quad operational amplifiers
ue differential inputs. These have several distinct advan-
ver standard operational amplifier types in single supply
tions. The quad amplifier can operate at supply voltages
as 3.0 Volts or as high as 32 Volts with quiescent currents
one fifth of those associated with the MC1741 (on a per
er basis). The common mode input range includes the neg-
upply, thereby eliminating the necessity for external biasing
onents in many applications. The output voltage range also
es the negative power supply voltage.

- rt Circuited Protected Outputs
- e Differential Input Stage
- gle Supply Operation: 3.0 to 32 Volts
- w Input Bias Currents: 100 nA Max (LM324A)
- ur Amplifiers Per Package
- ernally Compensated
- mmon Mode Range Extends to Negative Supply
- dustry Standard Pinouts

PIN CONNECTIONS



M RATINGS (T_A = +25°C unless otherwise noted)

Rating	Symbol	LM124 LM224 LM324,A	LM2902	Unit
pply Voltages Supply pplies	V_{CC} V_{CC}, V_{EE}	32 ±16	26 ±13	Vdc
erential Voltage Range (1)	V_{IDR}	32	±26	Vdc
mmon Mode Voltage Range	V_{ICR}	−0.3 to 32	−0.3 to 26	Vdc
ward Current (2) −0.3 V)	I_{IF}	50	—	mA
hort Circuit Duration	t_S	Continuous		
Temperature ic Package c Packages	T_J	175 150		°C
Temperature Range ic Package c Packages	T_{stg}	−65 to +150 −55 to +125		°C
ng Ambient Temperature e 24 24 24A 902	T_A	−55 to +125 −25 to +85 0 to +70 0 to +70 —	— — — — −40 to +105	°C

: Power Supplies.
s input current will only exist when the voltage is negative at any of the input leads
mal output states will reestablish when the input voltage returns to a voltage greater
n −0.3 V.

ORDERING INFORMATION

Device	Temperature Range	Package
LM124J	−55 to +125°C	Ceramic DIP
LM2902D	−40 to +105°C	SO-14
LM2902N	−40 to +105°C	Plastic DIP
LM2902J	−40 to +105°C	Ceramic DIP
LM224D	0 to +70°C	SO-14
LM224J	0 to +70°C	Ceramic DIP
LM224N	0 to +70°C	Plastic DIP
LM324AD	0 to +70°C	SO-14
LM324AN	0 to +70°C	Plastic DIP
LM324D	0 to +70°C	SO-14
LM324J	0 to +70°C	Ceramic DIP
LM324N	0 to +70°C	Plastic DIP

ELECTRICAL CHARACTERISTICS (V_{CC} = 5.0 V, V_{EE} = Gnd, T_A = 25°C unless otherwise noted)

Characteristic	Symbol	LM124/LM224 Min	LM124/LM224 Typ	LM124/LM224 Max	LM324A Min	LM324A Typ	LM324A Max	LM324 Min	LM324 Typ	LM324 Max	LM2902 Min	LM2902 Typ	LM2902 Max	Unit
Input Offset Voltage V_{CC} = 5.0 V to 30 V (26 V for LM2902), V_{CM} = 0 V to (V_{CC} − 1.7 V), V_O = 1.4 V, R_S = 0 Ω T_A = 25°C T_A = T_{high} to T_{low} (Note 4)	V_{IO}		2.0 —	5.0 7.0		2.0 —	3.0 5.0	— —	2.0 —	7.0 9.0	— —	2.0 —	7.0 10	mV
Average Temperature Coefficient of Input Offset Voltage T_A = T_{high} to T_{low} (Note 4)	$\Delta V_{IO}/\Delta T$	—	7.0	—	—	7.0	30	—	7.0	—	—	7.0	—	μV/°C
Input Offset Current T_A = T_{high} to T_{low} (Note 4)	I_{IO}	— —	3.0 —	30 100	— —	5.0 —	30 75	— —	5.0 —	50 150	— —	5.0 —	50 200	nA
Average Temperature Coefficient of Input Offset Current T_A = T_{high} to T_{low} (Note 4)	$\Delta I_{IO}/\Delta T$	—	10	—	—	10	300	—	10	—	—	10	—	pA/°C
Input Bias Current T_A = T_{high} to T_{low} (Note 4)	I_{IB}	— —	90 —	150 300	— —	45 —	100 200	— —	90 —	250 500	— —	90 —	250 500	nA
Input Common-Mode Voltage Range (Note 2) V_{CC} = 30 V (26 V for LM2902) V_{CC} = 30 V (26 V for LM2902), T_A = T_{high} to T_{low}	V_{ICR}	0 0	— —	28.3 28	0 0	— —	28.3 28	0 0	— —	28.3 28	0 0	— —	24.3 24	V
Differential Input Voltage Range	V_{IDR}	—	—	V_{CC}	—	—	V_{CC}	—	—	V_{CC}	—	—	V_{CC}	V
Large Signal Open Loop Voltage Gain R_L = 2.0 kΩ, V_{CC} = 15 V, for Large V_O Swing, T_A = T_{high} to T_{low} (Note 4)	A_{VOL}	50 25	100 —	— —	25 15	100 —	— —	25 15	100 —	— —	— —	100 —	— —	V/mV
Channel Separation 1.0 kHz ≤ f ≤ 20 kHz, Input Referenced	—	—	120	—	—	120	—	—	120	—	—	120	—	dB
Common-Mode Rejection Ratio R_S = 10 kΩ	CMRR	70	85	—	65	70	—	65	70	—	50	70	—	dB
Power Supply Rejection Ratio	PSRR	65	100	—	65	100	—	65	100	—	50	100	—	dB
Output Voltage Range R_L = 2 kΩ, V_{CC} = 30 V (26 V for LM2902), T_A = T_{high} to T_{low} (Note 4) R_L = 10 kΩ, V_{CC} = 30 V (26 V for LM2902), T_A = T_{high} to T_{low} (Note 4)	V_{OR}	0 —	— —	3.3 —	0 —	— —	3.3 —	0 —	— —	3.3 —	0 —	— —	3.3 —	V
Output Voltage — High Limit V_{CC} = 5.0 V, R_L = 10 kΩ, T_A = T_{high} to T_{low} (Note 4) V_{CC} = 30 V (26 V for LM2902), R_L = 2.0 kΩ, T_A = T_{high} to T_{low} (Note 4) V_{CC} = 30 V (26 V for LM2902), R_L = 10 kΩ	V_{OH}	26 27	— —	— —	26 27	— —	— —	26 27	— —	— —	22 23	— —	— —	V
Output Voltage — Low Limit V_{CC} = 5.0 V, R_L = 10 kΩ, T_A = T_{high} to T_{low} (Note 4)	V_{OL}	—	5.0	20	—	5.0	20	—	5.0	20	—	5.0	100	mV
Output Source Current V_{ID} = +1.0 V, V_{CC} = 15 V T_A = 25°C T_A = T_{high} to T_{low} (Note 4)	I_O +	20 10	40 20	—	20 10	40 20	—	20 10	40 20	—	20 10	40 20	—	mA
Output Sink Current V_{ID} = −1.0 V, V_{CC} = 15 V T_A = 25°C T_A = T_{high} to T_{low} (Note 4) V_{ID} = −1.0 V, V_O = 200 mV, T_A = 25°C	I_O −	10 5.0 12	20 8.0 50	— — —	10 5.0 12	20 8.0 50	— — —	10 5.0 12	20 8.0 50	— — —	10 5.0 —	20 8.0 —	— — —	mA
Output Short Circuit to Ground (Note 3)	I_{OS}	—	40	60	—	40	60	—	40	60	—	40	60	mA
Power Supply Current (T_A = T_{high} to T_{low}) (Note 5) V_{CC} = 30 V (26 V for LM2902), V_O = 0 V, R_L = ∞ V_{CC} = 5.0 V, V_O = 0 V, R_L = ∞	I_{CC}	— —	1.4 0.7	3.0 1.2	— —	1.4 0.7	3.0 1.2	— —	1.4 0.7	3.0 1.2	— —	1.4 0.7	3.0 1.2	mA

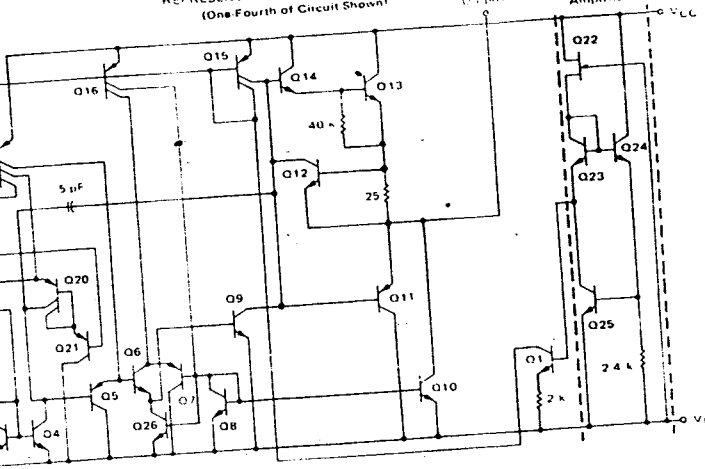
NOTES

(4) T_{low} = −55°C for LM124, T_{high} = +125°C for LM124
−40°C for LM2902 +85°C for LM2902
25°C for LM224 +70°C for LM224,A
0°C for LM324,A +125°C for LM2902

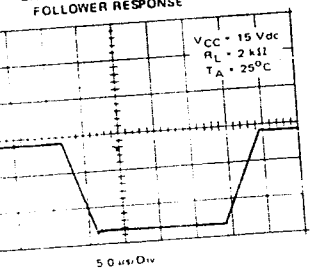
(1) This input common-mode voltage or either input signal voltage must not be allowed to go more than 0.3 V. The upper end of the common-mode voltage range is V_{CC} − 1.7 V.

(2) Short circuits from the output to V_{CC} can cause excessive heating and eventual destruction. Destructive dissipation can result from simultaneous shorts on all amplifiers.

REPRESENTATIVE CIRCUIT SCHEMATIC
(One Fourth of Circuit Shown)



LARGE SIGNAL VOLTAGE FOLLOWER RESPONSE

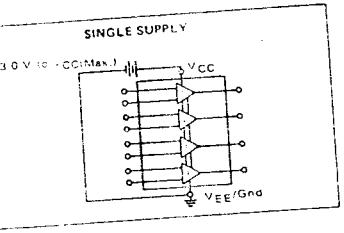


CIRCUIT DESCRIPTION

1124 Series is made using four internally com- two stage operational amplifiers. The first stage of each consists of differential input devices Q20 and Q18 with input buffer transistors Q21 and Q17 and the differential to single ended converter Q3 and Q4. The first stage performs not only the first stage gain function but also performs the level shifting and trans conductance reduction functions. By reducing the trans conductance a smaller compensation capacitor (only 5 pF) can be employed, thus saving chip area. The transcon ductance reduction is accomplished by splitting the col lectors of Q20 and Q18. Another feature of this input stage is that the input common-mode range can include the negative supply or ground, in single supply operation without saturating either the input devices or the dif ferential to single ended converter. The second stage con sists of a standard current source load amplifier stage.

Each amplifier is biased from an internal voltage regu lator which has a low temperature coefficient thus giving each amplifier good temperature characteristics as well as excellent power supply rejection.

SINGLE SUPPLY



SPLIT SUPPLIES

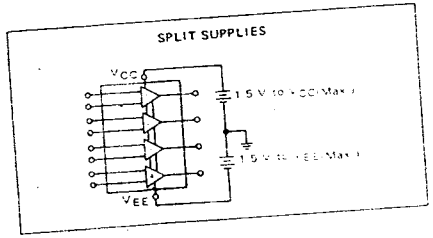


FIGURE 1 – INPUT VOLTAGE RANGE

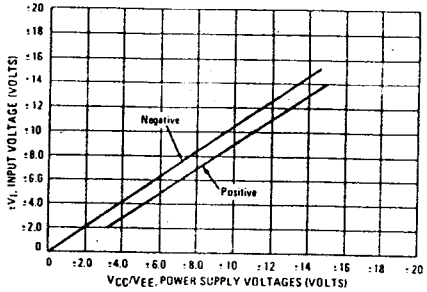


FIGURE 2 – OPEN LOOP FREQUENCY

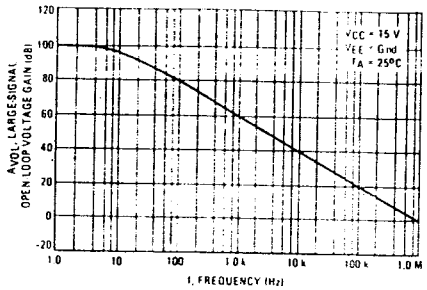


FIGURE 3 – LARGE-SIGNAL FREQUENCY RESPONSE

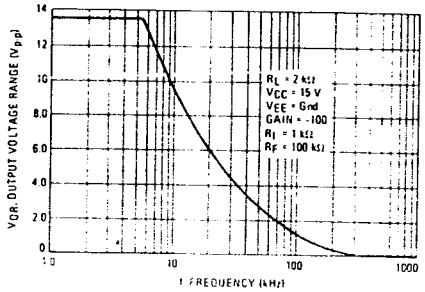


FIGURE 4 – SMALL-SIGNAL VOLTAGE FOLLOWER PULSE RESPONSE
(Non-Inverting)

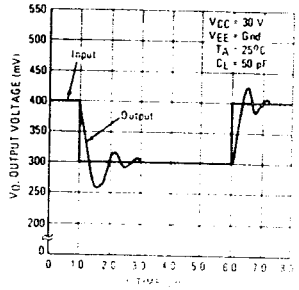


FIGURE 5 – POWER SUPPLY CURRENT versus POWER SUPPLY VOLTAGE

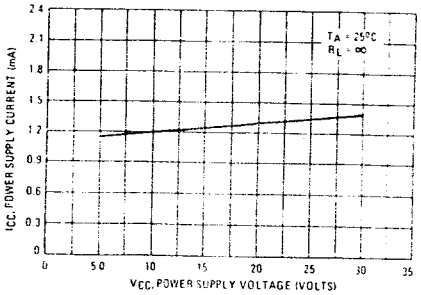
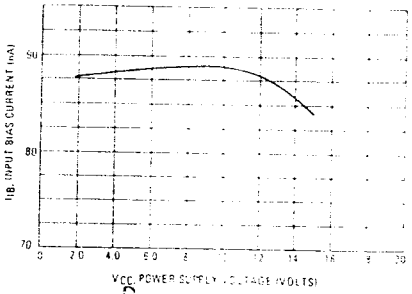
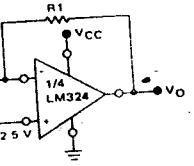


FIGURE 6 – INPUT BIAS CURRENT versus SUPPLY VOLTAGE



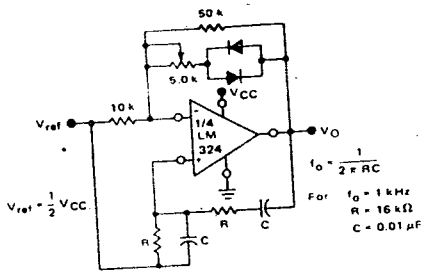
APPLICATIONS INFORMATION

OLTAGE REFERENCE



$V_O = 2.5\ V(1 + \frac{R1}{R2})$

FIGURE 8 – WIEN BRIDGE OSCILLATOR



PEDANCE DIFFERENTIAL AMPLIFIER

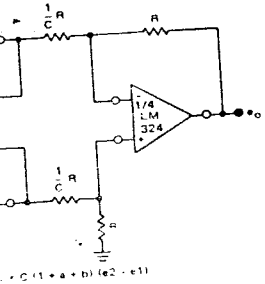
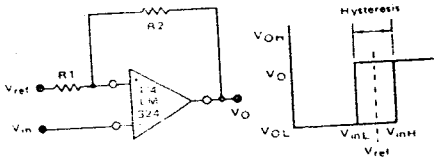


FIGURE 10 – COMPARATOR WITH HYSTERESIS

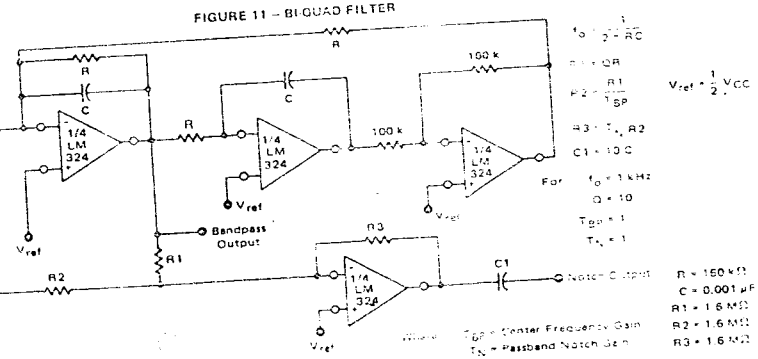


$V_{inL} = \frac{R1}{R1 + R2}(V_{OL} - V_{ref}) + V_{ref}$

$V_{inH} = \frac{R1}{R1 + R2}(V_{OH} - V_{ref}) + V_{ref}$

$H = \frac{R1}{R1 + R2}(V_{OH} - V_{OL})$

FIGURE 11 – BIQUAD FILTER



$f_o = \frac{1}{2\pi RC}$

$R1 = QR$

$R2 = \frac{R1}{2Q_{SP}}$

$R3 = T_N R2$

$C1 = 10\ C$

$V_{ref} = \frac{1}{2} V_{CC}$

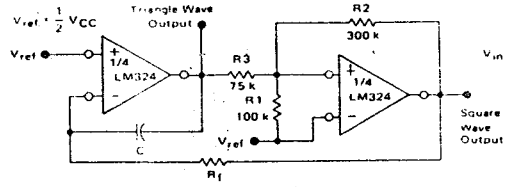
For $f_o = 1$ kHz
$Q = 10$
$T_{OP} = 1$
$T_N = 1$

$R = 160\ k\Omega$
$C = 0.001\ \mu F$
$R1 = 1.6\ M\Omega$
$R2 = 1.6\ M\Omega$
$R3 = 1.6\ M\Omega$

Where T_{OP} = Center Frequency Gain
T_N = Passband Notch Gain

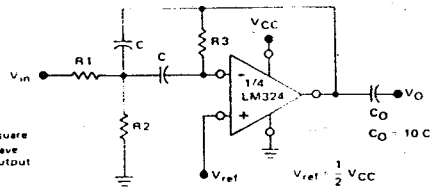
APPLICATIONS INFORMATION (continued)

FIGURE 12 – FUNCTION GENERATOR



$f = \frac{R1 + R_C}{4CR_f R1}$ if $R3 = \frac{R2\ R1}{R2 + R1}$

FIGURE 13 – MULTIPLE FEEDBACK BANDPASS FILTER



Given f_o = Center Frequency
$A(f_o)$ = Gain at Center Frequency

Choose Value f_o, C

Then

$R3 = \frac{Q}{\pi f_o C}$

$R1 = \frac{R3}{2A(f_o)}$

$R2 = \frac{R1\ R3}{4Q^2 R1 - R3}$

For less than 10% error from operational amplifier

$\frac{Q_o f_o}{BW} < 0.1$ Where f_o and BW are expressed in Hz.

If source impedance varies, filter may be preceded with voltage follower buffer to stabilize filter parameters.

MAX 232

RS 232 DRIVERS/RECEIVERS

FEATURES:

- Operates from single 5 volts supply.
- Meets all RS – 232C and V.28 specifications.
- Multiple drivers and receivers.
- On board DC – DC converter.
- +/- 9 volts output swing with 5 volts supply.
- Low power shut down - <1micro amps (typ).
- Three state TTL/CMOS receiver outputs.
- +/- 30 volts receiver input levels.

GENERAL DESCRIPTION:

MAX 232 belongs to the MAXIM family in which the line drivers/receivers are intended for all RS 232 and V.28/V.24 communications interface and in particular for those applications where +/-12 Volts is not available. Since nearly all RS 232 applications need both line drivers and receivers, the family includes both receivers and drivers in one package. Both the receivers and the line drivers (transmitters) meet all EIA RS 232 and CCIT V.28 specifications. The MAX 232 consists of three sections – the transmitters, the receivers and the charge pump DC – DC voltage converters.

DUAL CHARGE PUMP VOLTAGE CONVERTER:

The RS 232 drivers/receivers have on - board charge pump voltage converters which convert the +/- 5 volts input power to the +/- 10 volts needed to generate the RS 232 output levels. This + 5 volts to +/- 10 volts conversion is performed by two charge pump voltage converters. The first uses capacitor C1 to double the + 5 volts to + 10 volts, storing the + 10 volts on the V^+ output filter capacitor, C3. The second charge pump voltage converter uses capacitor C2 to invert the + 10 volts to – 10 volts, storing the – 10 volts on the V^- output filter capacitor, C4. A small amount of power may be drawn from the + 10 volts (V^+) and – 10 volts (V^-) outputs to power external circuitry.

DRIVER (TRANSMITTER) SECTION:

The transmitters or the line drivers are inverting level translators which convert the CMOS/TTL input levels to RS 232 or V.28 voltage levels. With + 5 volts Vcc , the typical output voltage swing is +/- 9 volts when loaded with a nominal 5kilo ohms input resistance of an RS 232 receiver. The output swing is guaranteed to meet the RS 232/V.28 specification of +/- 5 volts minimum output swing under the worst case conditions of all transmitters during the 3 kilo ohms minimum allowable load impedance Vcc = 4.5 volts and maximum operating ambient temperature. The open circuit output voltage swing is from (V^+ - 0.6 volts) to V^-.

The input thresholds are both CMOS/TTL compatible, with a logic threshold of about 25% of Vcc. The inputs of unused driver section can be left unconnected, an internal 400 kilo ohms input pull up resistor to Vcc will pull the

conditions. While or even a –3volts receiver threshold would not give proper indication on the control lines such a s DTR and DSR. The receiver on the other hand have a full 0.8V noise margin for detecting the power down or the cable connected states.

The receiver have a hysteresis of approximately of 0.5 Volts with a minimum guaranteed hysteresis 200 milli volts. This aids is obtaining clean output transitions even with slow rise and fall line input signals with moderate amount of noise and ringing, The propagation delays of the receivers are 350 nano seconds for negative going input signals and 650 nano seconds for positive going input signals.