C0335

ELECTRONIC DESIGN 2

1984

VOICE COMMUNICATIONS OVER ETHERNET

BY : STEPHEN STELLA 8101781J

PROPOSER : DR GREG EGAN.

SUPERVISOR : DR GREG EGAN.

ACKNOWLEDGMENTS.

I would like to sincerly thank the following people for the help and assistance they have given me during this project.

My Parents.

Dr G.Egan.

Mr J.Hulskamp.

Mr A.Young.

Mr R.Alessio.

CONTENTS.

INTRODUCTION.

This design project is largely a practical demonstration of of any packet speech system regardless of the packet network it runs on.

The principle of packet speech is quite simple. An analog signal,such as speech, is converted to a digital format using a A/D converter. The digital information is then stored in a buffer until a "packet" of information has been built. This information is then transferred at high speed over some high bandwidth packet system to a receiver station. ETHERNET for example, transfers the data at 10 meg/bits per second. At the receiving station,the information is dumped in another buffer and then "unpacked" at the rate it was originally "packed at. The unpack information is next sent to a D/A. The analog signal from the D/A is then filtered, amplified and becomes available to the receiver.

While there is a slight delay, the receiver should not be able to notice any interuption in the signal he receives.

Packet speech offers several potential advantages over traditional telecommunication systems. For example, human speech generally occures in bursts and packet systems handle this type of traffic both efficiently and effectively. Packet speech makes more efficient use of available channel bandwidth compared to most systems in use. Information routing is simplified and greatly reduces the hardware and wiring used by conventional systems today. Hence we eliminate the need to dedicate a physical link for the duration of a "call" and that link my be used for other purposes (eg workstation communications or file

transfer).

Since the information is in a digital form it is possible to obtain better quality audio signals with reduced noise degradation. It is possible to secure the information by scrambling or other means and it is possible to provide new services such as "voice mail".

With proper planning it is relatively easy to interface a packet speech system to the national carrier (Telecom) AUSPAC and MIDAS services for long haul communications.

One practical application for this technology is the replacement of traditional PABX/PDBX systems that are in use today.

ETHERNET, for example, is a superior digital communications network which is finding increasing use by many organizations today. Such organizations however, would also require the services of a PABX to provide their voice communication needs. This represents a duplication of resources that ETHERNET is singly capably of supplying. A PABX represents a considerable investment of finances in equipment and cable laying when advantage could be taken of the equipment and routing provided by ETHERNET.

Such a system would reduce the exchange requirements to that required to interface to Telecom. All internal routing would be handled by ETHERNET.

A voice communications system utilizing ETHERNET could provide all the services provided by traditional PABX/PDBX systems plus new features which would incorporate the advantages

of ETHERNET. Services such as voice mail or teleconferencing with similtaniously displays or printouts of documents and diagrams would become realizable.

Hence, any organization that requires the services of an ETHERNET system could benefit from the introduction of a packet speech system.

Packet speech will become more important in the future as demand for resources such as satellites increases and maximum utilization must be made of these expensive data channels.

RESEARCH.

Several factors had to be traded of in this project in order for it to be realizable and of practical use. A discussion of these factors, and the design considerations involved with each, follows.

SPEECH QUALITY.

The first criterion to be considered is the quality of the signal that we expect from a packet speech system.

The main criterion that I used to quantify speech quality is the bandwidth of the transmitted signal. This is because the signals bandwidth determine the amount of data that must later be transferred. This is a factor which should be minimised in order to maintain efficiency. Other factors such as signal to noise ratio or distortion are not effected by the digital side of the project and , since they are not traded off against other operating parameters, are not considered here.

I based the bandwidth requirements of the system around a standard that has already been established and commonly accepted. That standard is the CCITT recommendation for digital telephone systems.

This means that the system has the performance of the telephone service that we are all use to and is typical of the types of applications that I believe this system could be used for.

The CCITT recommend that the bandwidth of such a system be 300 Hz to 3.3 kHz and that the signal be sampled at 8 kHz. The A/D-D/A converters should use A-LAW companding and have a

resolution of 8 bits. This results in 64kbits or 8 kbytes of data to be transferred each second. (Half Duplex).

Conveniently, several manufactures have developed chip sets which perform the companding A/D,D/A conversion. Such chips are refered to as CODEC'S and the one used in this design is a Motorola device designated as the MC14407. It is part of a family of chips which provide the various functions required to build telephone systems.

HUMAN AUDITORY PERCEPTION.

The next factor I considered was that of human auditory perception. This involves the maximum delay and interuption that is allowed in the signal before the user notices ,or worse, becomes annoyed. The delay in the received signal is due to the time it takes to build a packet of data, transfer that data to the receiving station and then "unpacking" it. Signal interuptions are due to packets of data being discarded due to collisions when trying to transfer the data or not being able to gain access to the transmission medium. These are situations that are likely to occur in an ETHERNET environment.

The books I refered to suggested that humans are capable of detecting interuptions and delays in signals only when they are greater than 25 ms in duration. Researchers in America who have been working on packet speech for several years now operate their systems with packets which are up to 50 ms in duration. I am accepting packets of 16 ms duration in this system for the reasons that follow.

PACKET SIZE.

It is the size of the data packets that determines the amount of delay in the received signal and the interuption time when a packet is lost. ETHERNET strictly defines the size of its packets to be between 46 and 1500 bytes long. We would like the packets to be as large as possible to minimize packet overheads and thus make efficient use of ETHERNET. Larger packets, however, require larger and more expensive fifos for storing the packets in. The compromise I decided on was a packet size of 128 bytes. This represents a time slice of 16 ms when sampling at 8 kHz. This is a small packet and is inefficient to send over ETHERNET. However RMIT's ETHERNET is, at this time, under utilized and efficiency is not really a problem. Smaller packets will also allow more subtle investigation of signal degradation due to varing percentages of deliberate packet loss. One last factor is that 128 bytes is a convienient size for the fifos required on the system.

SYSTEM PERFORMANCE.

Consider two 10 megbit channels which are to be used for telephone communications, One being an ETHERNET system and the other being an traditional PCM multiplexed system.

The PCM system could support 156.25 (10 megbit/64kbit) similtanious telephone channels. (Some channels would have to be used for control).

The ETHERNET system with a data packet size of 128 bytes has each channel sending 77 kbits each second. (Includes packet overheads). Each packet will contain 1232 bits and assuming that

a packet from each channel is sent somewhere within a 16 ms interval, (ie before the next packet of each channel is ready to be sent) then the time to send each packet takes on average 133.2 us. (1232 bits * 100 ns per bit + 10 us delay between packets). Each channel must send 62.5 packets per second, so each channel has control of ETHERNET for 8.325 ms. Thus if we run ETHERNET at say 90 % efficiency we can support 105 channels similtaniously. This appears to be worse than the PCM system but ETHERNET can easily suppress the sending of empty or "silent" packets which occur when people are listening or thinking. Assuming people talk for 50 % of the time then the effective capacity of the ETHERNET system doubles to about 210 voice channels. That represents a 25 % increase in effective bandwidth and it is this improvement in performance that may see the introduction packet voice systems in the near future.

DESIGN.

This section outlines the design of the physical hardware required to demonstrate packet speech. The circuit diagrams should be refered to when reading this section.

DIGITAL SYSTEM.

Note that the reset pins on all the chips have been connected and are reset on power up by the reset circuitry. This feature is extremely useful for forcing the hardware to a known state for hardware debugging and testing. Also note that where ever CMOS drives TTL then a 10 kohm pull up resistor has been included to ensure correct TTL levels are maintained.

S100 INTERFACE.

The hardware is desisned around the S100 buss so that it is easily interfaced to RMIT work stations which are Z80 S100 systems. The S100 card is a convenient size and is available for wire wrapping which is invaluable when developing prototype circuits.

The design was originally going to appears to the mother board as two i/o locations, one for writting data to the output fifo and the other for reading data from the input fifo. These two i/o locations now exist at the same place in the i/o map since the act of reading or writing data will determine which port is being addressed. Hence the board operates through a single i/o location. That i/o location was arbitrarily selected to be 182 (decimal) as it appears to be "out of the way" of the rest of the systems i/o and is reasonably easy to decode. Reading data off the board simply requires enabling a data

buffer (74LS244) connecting the output fifo to the data in lines on the S100 buss. The buffer is enabled when both the address decoding signals and sINP on the s100 buss are present. The negative edge of this signal is used to shift out the next byte of data from the output fifos ,ie after the micro has read current byte.

The output of data to the board is only a little more difficult and involves latching data off the S100 buss and then shifting it into the fifos. Data is loaded into the latch (74LS273) when the address decode, sOUT and pWR* signals are all present. The negative edge of this signal is inverted and used to shift in the current data byte from the latch to the fifos.

FIFOS. (3341)

The fifos used are the Fairchild 3341 which are 4 bit by 64 word devices. Each buffer consists of four 3341's configured to provide the 128 byte deep array required for the design. These devices are designed to interface together to make larger fifo arrays. The only problem that I encountered with this section was that the spec sheets I was working from were for bipolar equivalents of the 3341 and hence failed to mention the - 5 volt rail that the 3341's require for correct operation.

CODEC. (MC14407)

It is worth discussing the codec devices at this stage so that the interfacing between the fifos and the codec is more easily understood.

The first consideration is that the codec chip transfers

digital data in serial bit streams because it was designed to work in PCM multiplexed systems. Hence shift registers are required to transfer data in and out of the chip.
The other main consideration is the fact that the codec chip operates on a 12 volt supply. This complicates the design somewhat since the remainder of the circuit runs off a 5 volt rail.

Refering to the spec sheets, the codec has four main digital input signals and one digital output. The input signals are RDD which is the serial receive data input, CCI which is used as a convert clock and serial data clock rate. TDE and RCE which control the time period during which the 8 bits of data are clocked in and out of the chip during each conversion cycle and MSI which is a synchronizing signal. An open collector 74LS gate with a 1 kOhm pull up resistor is used to convert each signal from a 5 volt TTL level to a 12 volt CMOS level. AND gates (74LS09) were used because I could not obtain a hex buffer package such as a 74LS06 or 74LS07.

The digital output from the codec is the serial transmit data line. This signal is converted to TTl levels with a simple zenor diode arrangement.

Getting the data in and out of the codec required the use of a SIPO shift register (74LS164) and a PISO shift register (74LS166). Some combinational logic was required to generate the timing signals required to synchronize the loading and shifting of data in these devices. The clock circuitry and its operation can be best explained by the timing diagram and the circuit diagram.

The only complication with this part of the circuit was that the fifos tended to shift in or out several bytes of data instead of a single byte each time they transferred data to or from the shift registers. The fifos appear to be level triggered instead of edge triggered and they raced for the duration of the LFIFO (Load Fifos) pulse. The solution to this problem is the addition of the flip flops to limit each operation to a single byte transfer.

That completes the digital hardware involved with the design. The next section will discuss the analog hardware and its design.

ANALOG SYSTEM.

The analog system of the design involves the microphone preamp, output driver and the bandlimiting filters.

CODEC.

The codec takes care of the A/D, D/A conversions required by the system. One unusual feature of the codec that required some attention is that the analog earth of the codec is NOT the same as the digital ground. The codec generates a seperate analog ground at one half of its supply voltage, ie with a 12 volt supply, the codec generates an earth rail which is 6 volts above the digital earth. This was quite useful when interfacing to the the op amps used for the amplifiers and filters but is a bit of a problem when interfacing to external equipment which is mains earthed.

The analog earth requires a pull up resistor if it is to be used elsewhere in the circuit and should be ac coupled to the

system earth and 12 volt supply rail to improve the systems noise performance.

The codec is designed to interface to an external PAM sampled data filter on both its input and output. This external device is part of the codec chip family and is refered to as the MC14413.

Unfortunately this chip is not available in Australia. The MC14414 provides 5th order bandpass filtering on the input signal to the codec but more importantly provided output filtering and sinx/x correction for the 50 % duty cycle PCM output from the codec. The output circuitry used and its problems is discussed later.

INPUT STAGE.

The input stage is reasonably simple and acts as an amplifier and bandpass filter to the input signal thus reducing aliasing problems. The MC14413 acts as a 5th order elliptic low pass filter and a 3rd order Chebyshev high pass filter.

My design settles for 4th order band pass filtering since it could be implimented with one quad op amp and would also act as the preamplifier.

OUTPUT STAGE.

This stage was a problem. I originally had a 4th order Butterworth filter on the output from the codec. This worked but as I discovered later was of no use with PAM signals because of the non linear time delay introduced by the Butterworth filter.

This resulted in quite spectacular phase distortion on the output signal.

The situation was improved by first passing the output pulses through a 2nd order biquad Bessel's filter with a variable transmission zero. This not only improves the phase response of the system, it also improves the "tone" of the output signal.

SOFTWARE.

Software is an integral part of this project and has been involved since construction of the boards began. The following discussion deals with the development of these programs. (Refer to attached program listings).

The hardware was designed so that the software interface would be easy to impliment and efficient. Hence there are no status ports to be checked or timing constraints that need to be considered by the software.

The first program is called codec and is a simple diagnostic package which proved invaluable during the construction and debugging of the hardware.

The S100 interfaces were tested by reading or writting a known byte continuously to or from the i/o ports on the boards. The fifos and shift registers were tested by connecting the two shift registers together so that data could be passed through one to the other via the fifos and read back 256 bytes later. These tests were carried out using "Codec" to drive the hardware.

(NOTE : The analog system was tested similarly by connecting the output from the codec back into its input and using the chip in a "play back" mode).

The second program is called CD1 and was developed as the first board was being completed. It allows the transfer of packets to and from the one board in real time so that the board could again be used in a "play back" mode. This program also demonstrated that the project was infact going to work as intended !

CD1 operates by initilizing a CTC to cause an interupt every 16 ms. The interupt routine then takes 128 bytes from the output fifo and transfer each byte to the input fifo, ie it sends a packet to itself.

This program can easily be used to investigate the effects of packet size.

The next program written is called CD2 and is a full duplex ETHERNET simulator between two work stations. It will detect packet collisions and delays the sending of its next packet when they occur. In this way the sending of packets will eventually synchronized between the two stations so that they don't overlap. Any packet involved in a collision is re-sent but the following packet is lost since the fifos still hold the "collision" packet and the following packet cannot be saved.

This program has been successfully run at half duplex. It has not been tested at full duplex because the program requires two PIO's for full duplex operation and the work stations only have one free PIO each.

The time taken by this program to send each packet is just under 8 ms so it will be able to send two packets (full duplex operation) per 16 ms but only just !

Infact the software would have been much more efficient if it had not been interupt driven. This is because it has to operate as a state machine and evaluate its status each time it is interupted to send or receive another byte of a packet. It is easier to only interupt at the start of the packet and then send the packet without the use of further interupts. This requires the micro to control the hand shaking to the other machine but

since these operations occur concurrently between the two machines, then less time will be wasted waiting for each byte to be accepted. Unfortunately the PIO's can not be used with its inbuilt handshaking unless it is interupt driven. However the four i/o ports on the work station could easily support a non interupt drive program, given the appropriate connectors.

The only problem with this arrangement is that the detection of collisions is much more difficult and might require a few more control lines between the two machines or assigning one machine as the master and the other as the slave.

IMPROVEMENTS.

There are several areas where future improvements could be made to this design.

The first is the audio section. The quality of the voice signals is largely dependent on this section and the performance of the system would improve greatly with the addition of the MC14413 filter chip to the input and output of the Codec chip.

Since the fifos appear to be level triggerred instead of edge triggerred then flip flops should be added to the fifo input and output to the S100 interface to ensure correct loading of data.

Lastly the software could be written so that the data ports are used for communications between two work stations instead of PIO's in interupt mode. This would improve the efficiency of the software since there would be less overheads to proccess.

BIBLIOGRAPHY.

Digital Processing of Speech Signals.
L.R.Rabiner and R.W Schafer.
Prentice Hall 1978.

Foundations of Modern Auditory Theory. Vol 1.
Edited by J.V.Tobias
Academic Press 1970.

Visual and Auditory Perception.
G.M.Murch
Bobbs Merill Co. Inc. 1973.

Analog Delay Lines.
R.Marston.
Murray Publishers. ETI Feb 1983.

Local Telecommunications.
Edited by J.M.Griffiths.
Short Run Press Ltd. 1983.

Introduction to Local Area Networks.
Digital Equipment Corp. 1982.

Australian Electronics Directory.
Technical Indexes Pty ltd. 1983-1984.

The S-100 BUS Handbook.
D.Bursky
Haden Book Co. Inc.1980.

Introduction to Filter Theory.
D.E.Johnson.
Prentice Hall. 1976.

Linear Data Book.
National Semiconductor. 1980.

Programming the Z80.
R.Zaks.
Sybex 1980.

Analog/Digital Conversion Handbook.
Analog Devices Inc. 1972.

APPENDIX A

PARTS LIST.

INTERGRATED CIRCUITS.

IC1	MC14407
IC2 - IC9	F3341
IC10	74LS164
IC11	74LS166
IC12, IC13	74LS161
IC14 - IC16	74LS21
IC17	74LS09
IC18	74LS04
IC19	74LS273
IC20	74LS244
IC21, IC22	74LS21
IC23	74LS08
IC24	74LS04
IC25	74LS74
IC26	7812
IC27	7805
IC28	7905
IC29 - IC31	LM348

1	3.1V ZENOR DIODE.
1	5.1V ZENOR DIODE.

RESISTORS.

2	10 KOHM TRIMPOTS.
1	100 KOHM TRIMPOT.
2	10 KOHM SIP RESISTOR PACKS.

1	10	OHM
15	1	KOHM
4	2.2	KOHM
1	2.7	KOHM
15	10	KOHM
1	12	KOHM
2	22	KOHM
1	33	KOHM
2	47	KOHM
2	100	KOHM
1	150	KOHM

CAPACITORS.

2	1500	pF	CERAMIC.
2	2200	pF	"
1	4700	pF	"
4	4.7	nF	"
4	22	nF	"
30	100	nF	" TTL DECOUPLING.
15	10	uF	ELECTRO. (16 V).
1	100	uF	"

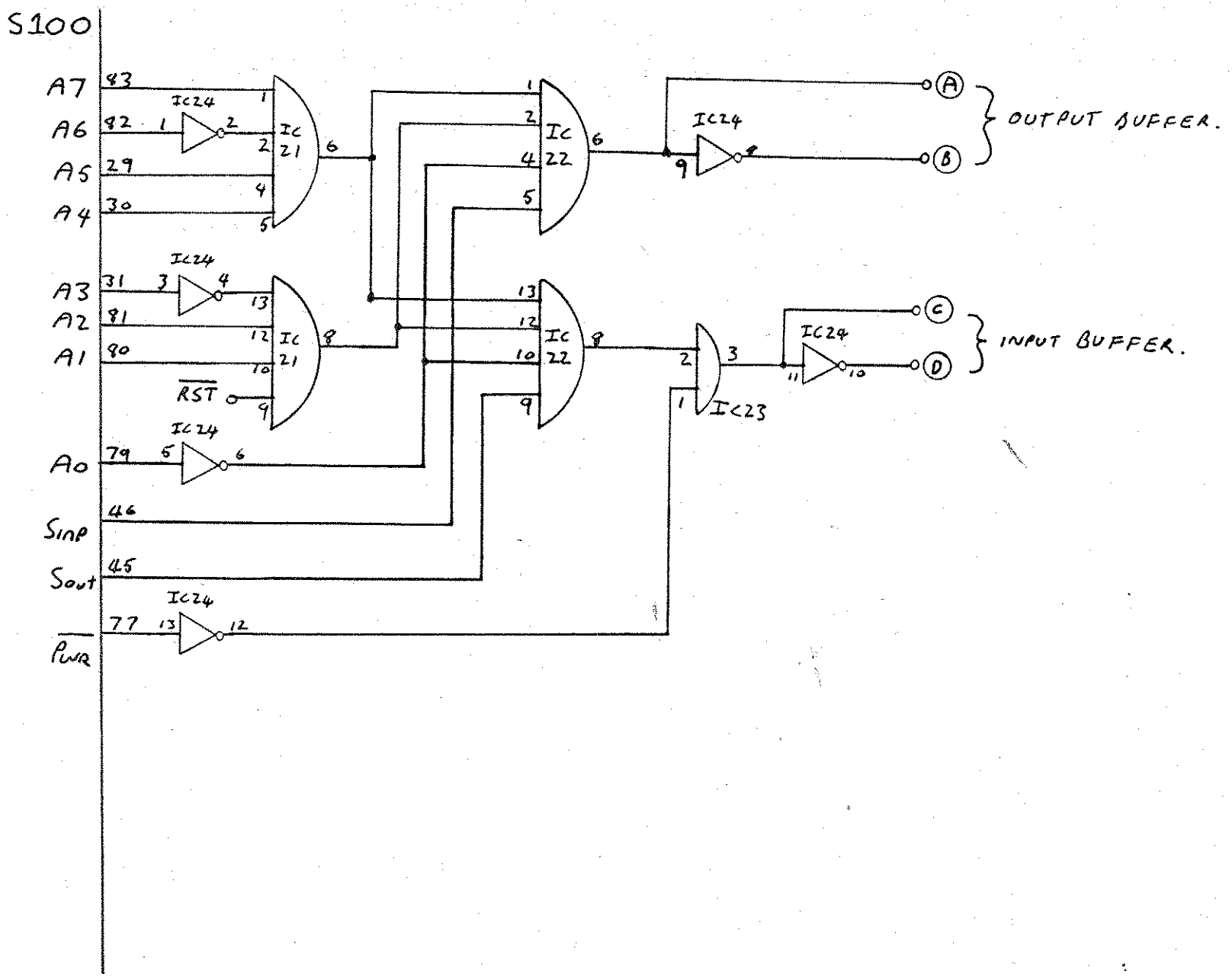
WIRE WRAP SOCKETS.

14	14 PIN.
11	16 PIN.
2	20 PIN.
1	24 PIN.

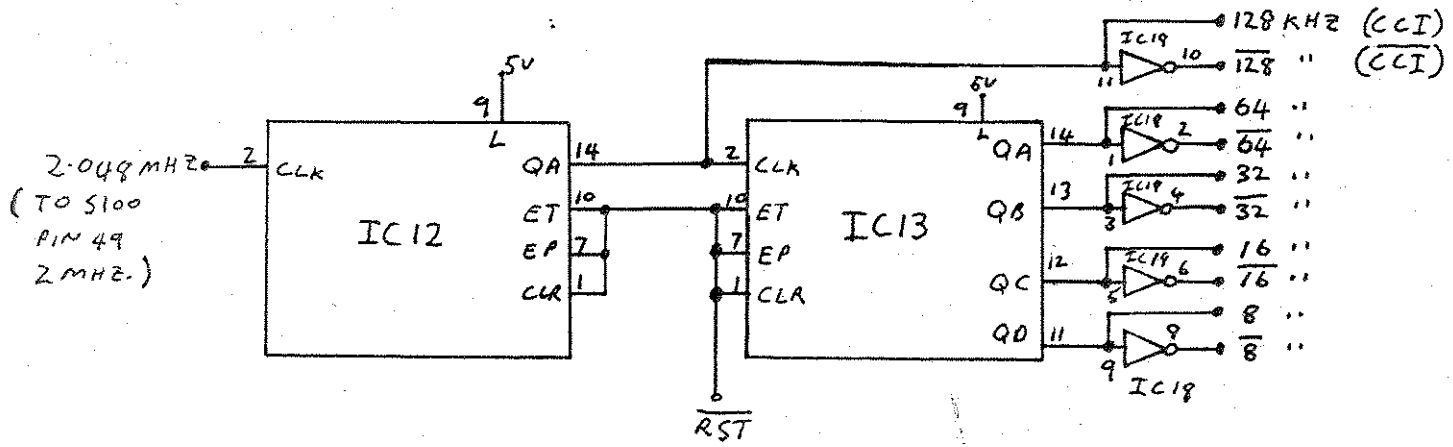
HARDWARE.

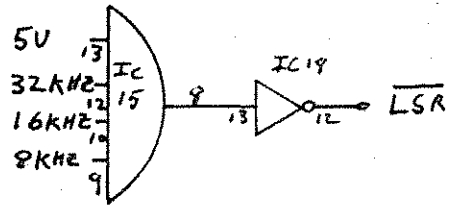
1	S100 WIRE WRAP CARD.
2	6.5 MM PHONO PLUGS.
30	WIRE WRAP PINS.
1	1K:1K AUDIO TRANSFORMER.
2	B4 NUTS, BOLTS AND WASHERS.

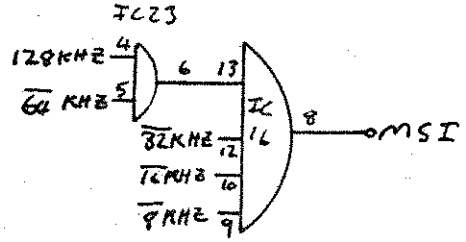
APPENDIX B

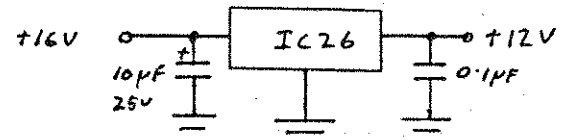


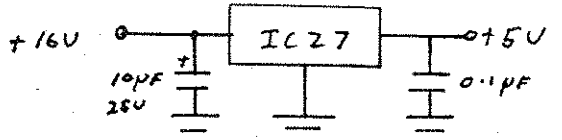
S100 ADDRESS DECODING

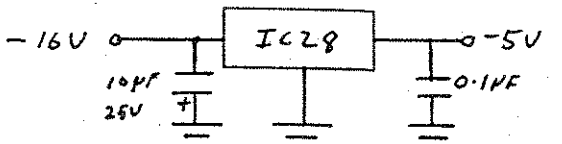


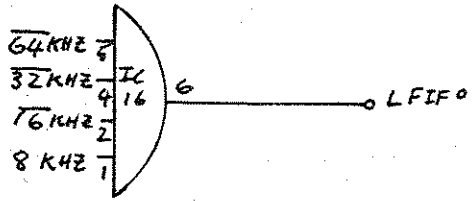


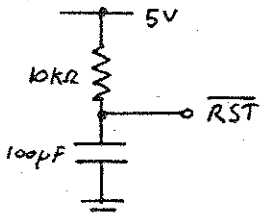




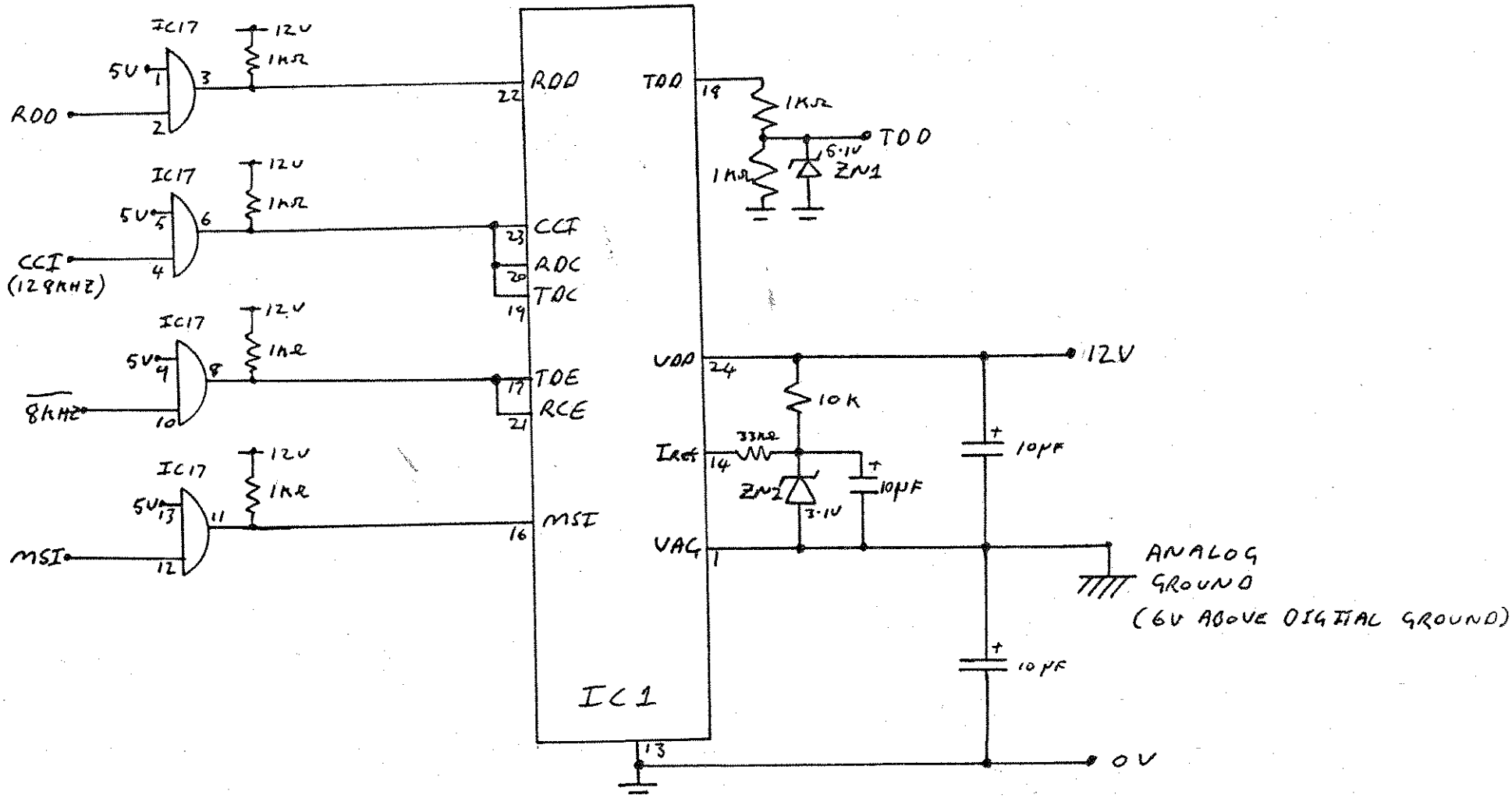




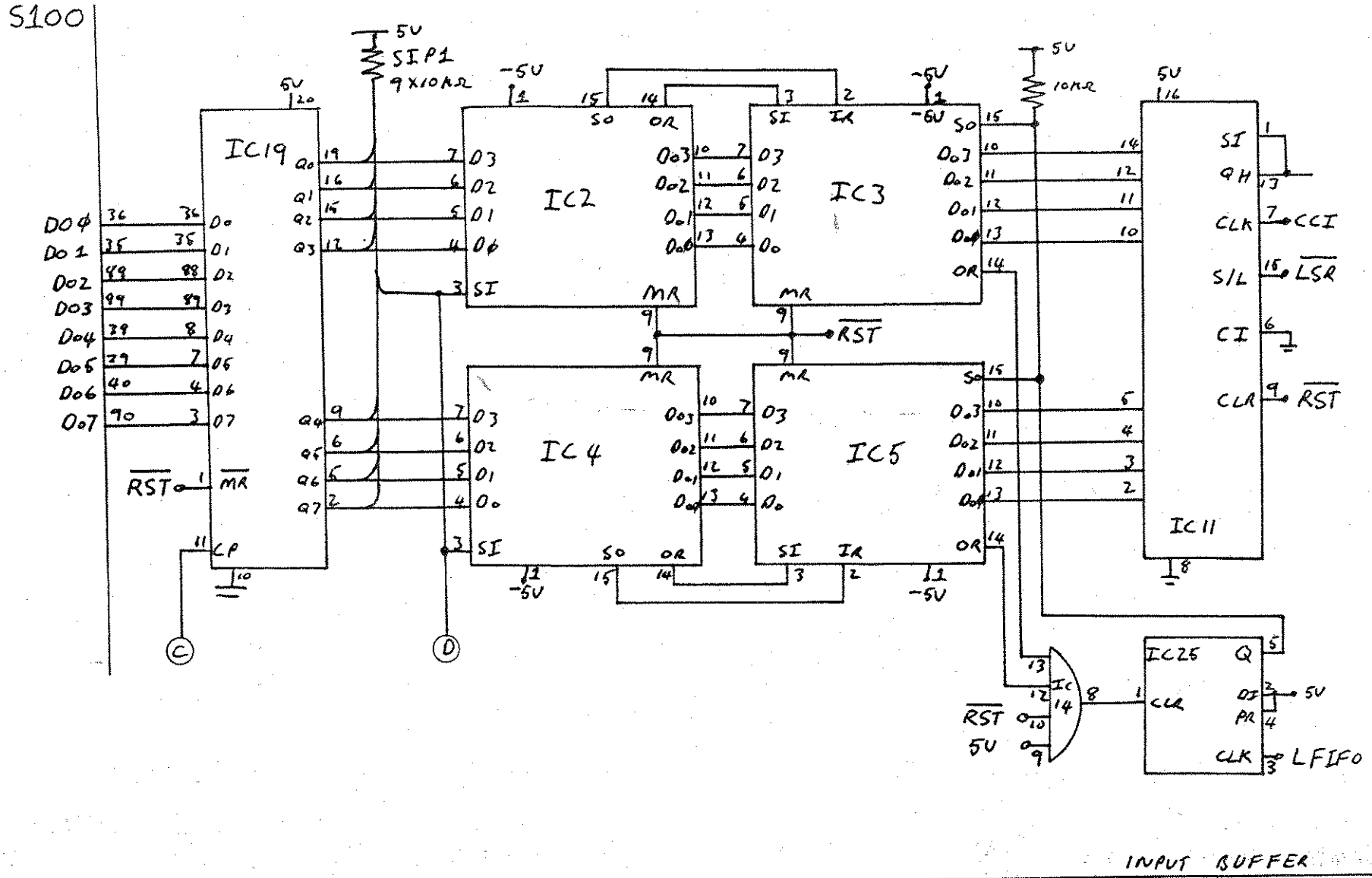


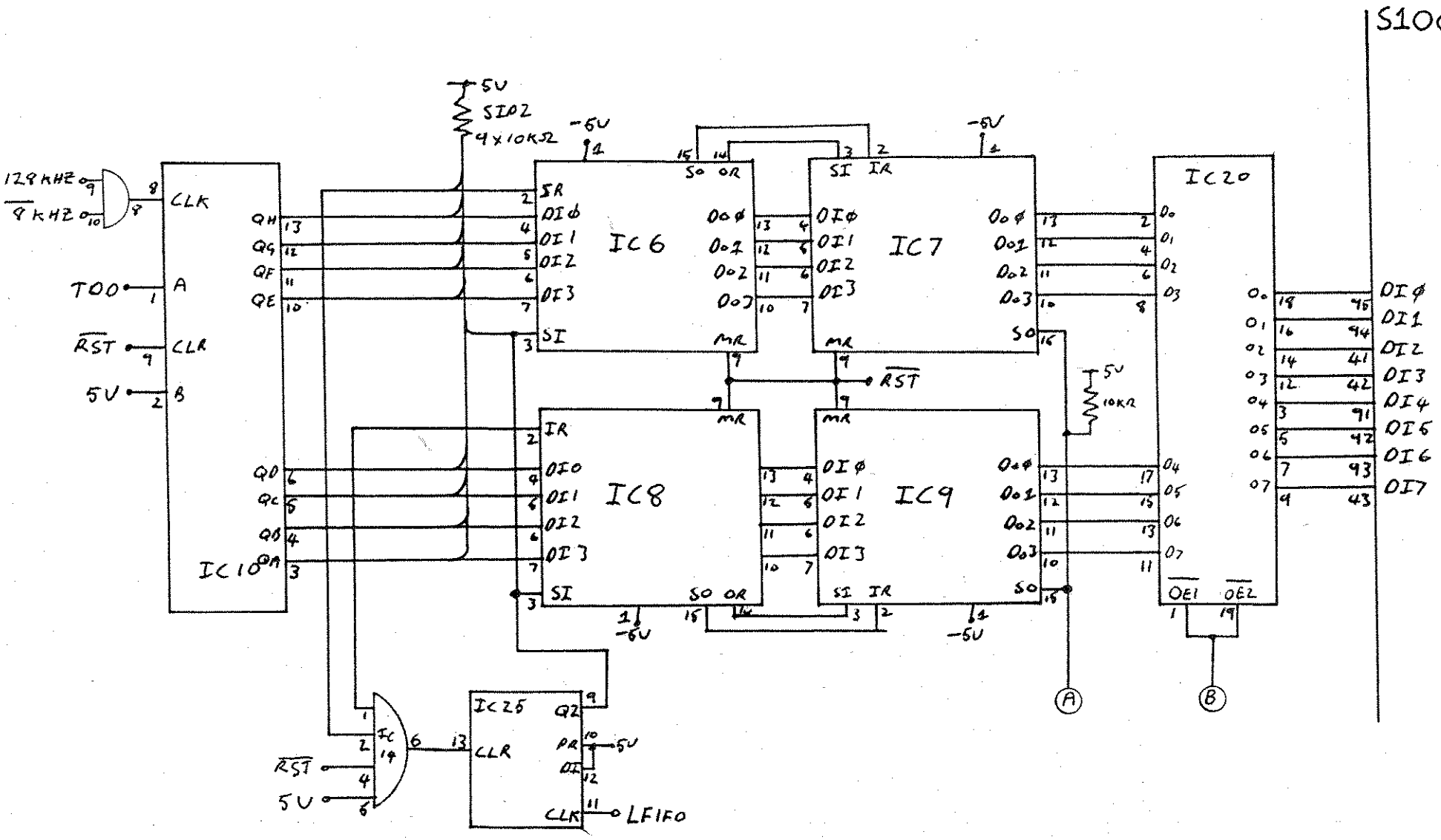


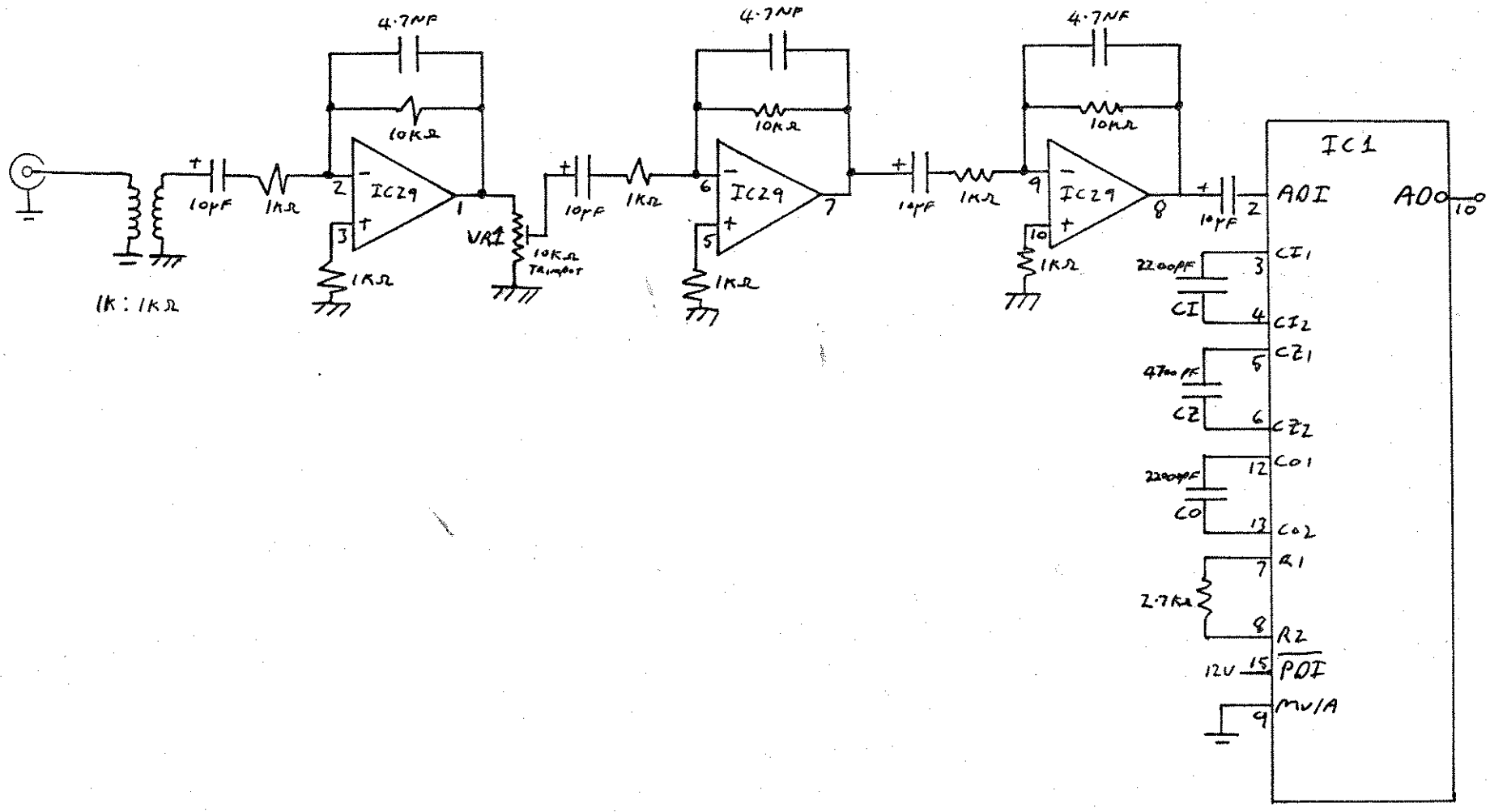
POWER, CLOCK AND RESET CIRCUITRY



CODEC DIGITAL INTERFACE

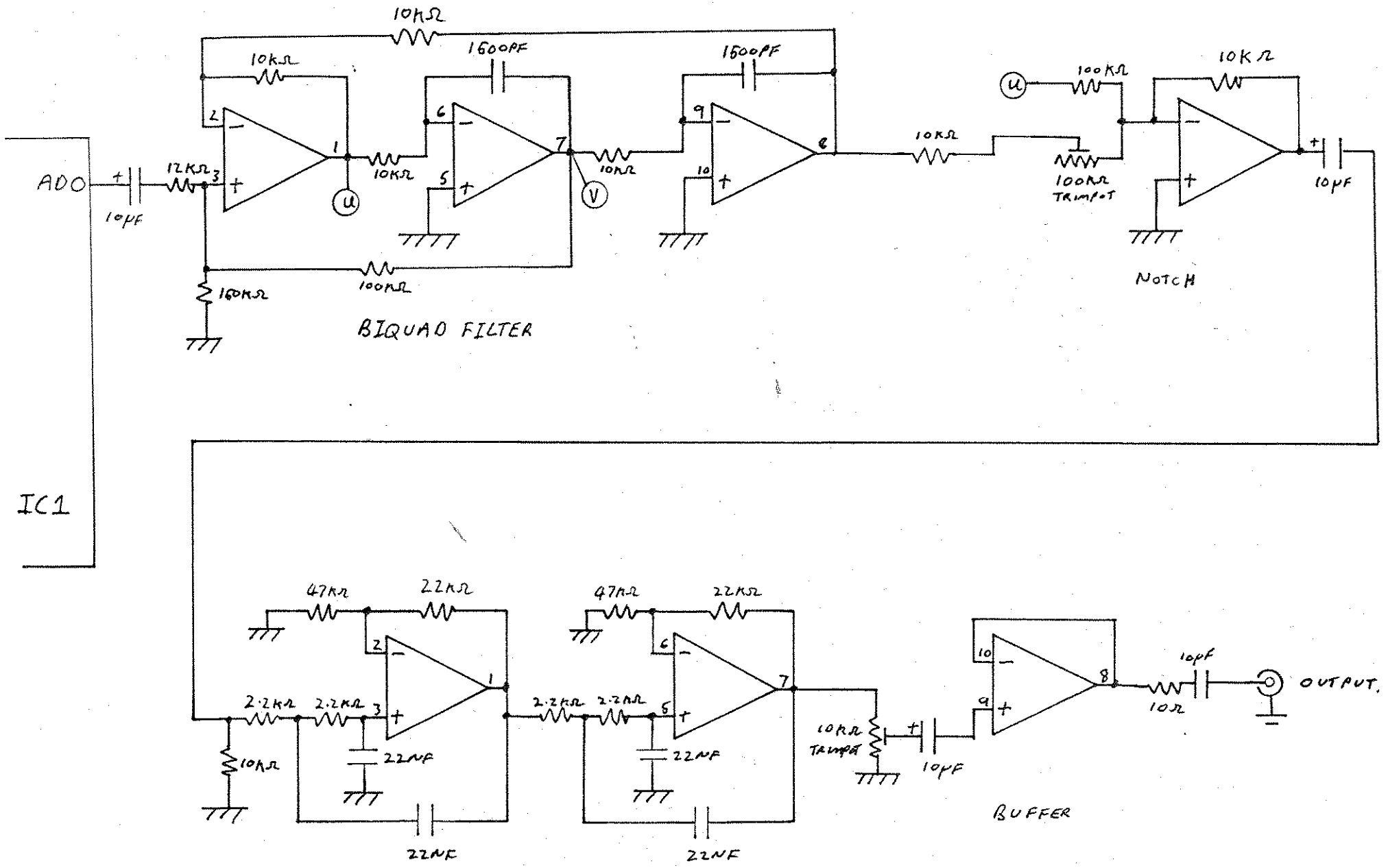

S100
5V
SIP1
9X10KΩ
IC19
IC2
IC3
IC4
IC5
IC11
IC25
-5V
10KΩ
DO0
DO1
DO2
DO3
DO4
DO5
DO6
DO7
RST
MR
CP
SI
SO
OR
IR
CLK
S/L
CI
CLR
CCI
LSR
LFIFO
C
D
INPUT BUFFER


S100
5V
SIP2
4x10KΩ
-6V
128 KHZ
9 KHZ
CLK
A
CLR
B
TOO
RST
5V
QH
QG
QF
QE
QD
QC
QB
QA
IC10
IC6
IC7
IC8
IC9
IC20
SR
DI0
DI1
DI2
DI3
SI
SO
OR
IR
MR
DO0
DO1
DO2
DO3
10KΩ
OE1
OE2
IC25
Q2
PR
CLR
LFIFO
IC 14
OUTPUT BUFFER



1K : 1KΩ

ANALOG INPUT


IC1
AD0
10µF
12KΩ
10KΩ
1600PF
10KΩ
1600PF
10KΩ
100KΩ
160KΩ
BIQUAD FILTER
10KΩ
100KΩ
TRIMPOT
10KΩ
10µF
NOTCH
47KΩ
22KΩ
2·2KΩ
2·2KΩ
10KΩ
22NF
22NF
47KΩ
22KΩ
2·2KΩ
2·2KΩ
22NF
22NF
10KΩ
TRIMPOT
10µF
BUFFER
10pf
10Ω
OUTPUT
4TH ORDER BUTTERWORTH
L.P. FILTER
ANALOG OUTPUT

APPENDIX C

```
program codec;

{This is a diagnostic program used to test and debug the hardware boards}
{during their development. It is written to run under Turbo pascal .     }

const fifo=182;

var   a,b,d,c,e,f:integer;
      u,v,w:char;

procedure packet;
begin
    writeln('packet size ?');
    readln(a);
    writeln('delay');
    readln(b);
    repeat
       for c:=0 to a do
       begin
          a:=port^182^;
          port^182^:=a;
       end;
      for c:=0 to b do
      begin
          a:=port^2^;
          port^2^:=a;
      end;
    until keypressed;
end;

procedure write;
begin
   writeln('Value to output to fifo ?');
   readln(a);
   repeat
       port^fifo^:=a;
   until keypressed;
end;

procedure read;
begin
   repeat
      a:=port^fifo^;
      writeln(a);
   until keypressed;
end;

procedure copy;
begin
   repeat
     a:=port^fifo^;
     port^fifo^:=a;
   until keypressed;
end;
```

```
procedure real;
{This procedure uses the onboard ctc (8253) to send packets in real time}

var a,b,c,d,e,f:integer;

begin
port^35^:=182;
port^34^:=232;
port^34^:=3;
c:=128;
  repeat
     repeat
        port^35^:=128;
        a:=port^34^;
        b:=port^34^;
     until a=0;
     for b:=0 to c do
     begin
       a:=port^182^;
       port^182^:=a;
     end;
     repeat
       port^35^:=128;
       a:=port^34^;
       b:=port^34^;
     until a<>0;
until keypressed;
end;

begin { main }
  repeat
      writeln('(P)acket  (W)rite   (R)ead   (C)opy   (Q)uit   (T)est  ?');
      readln(u);
      case u of
       'p':real;
       'w':write;
       'r':read;
       'c':copy;
       't':packet;
       'q':
       end; {case}
  until u='q';
end.
```

```
;Codec Test Program 1.
;By Stephen Stella OCT 1984
 For DESIGN 2 C0335.
 This  program is interupt driven and simply transfers packets of
;data to and from a single codec board.

        org     100h
delay   equ     250             ;sets timer to interupt every 16 ms.
packet  equ     125             ;size of packets.
;tc     equ     20h             ;i/o address of ctc. (8ah for onboard ctc)
;tcvect equ     80h             ;ctc interupt vector.  (0f4h for onboard ctc)
fifo    equ     182             ;i/o address of codec board.
      ;sets timer to interupt every 16 ms.
 007D                   packet  equ     125             ;size of packets.
 0020                   ctc     equ     20h             ;i/o address of ctc. (8ah for onboard ctc)
 0080                   ctcvect equ     80h             ;ctc interupt vector.  (0f4h for onboard ctc)
 00B6                   fifo    equ     182             ;i/o address of codec board.

 0100'                  start:  ;initilize timer.

 0100'  ED 57                   ld      a,i             ;get interupt vector offset.
 0102'  67                      ld      h,a             ;save as msbyte of hl.
 0103'  2E 80                   ld      l,ctcvect       ;ctc vector offset.
 0105'  11 011E'                ld      de,txint        ;address of routine to put in vector.
 0108'  73                      ld      (hl),e          ;save lsbyte of address in vector.
 0109'  23                      inc     hl
 010A'  72                      ld      (hl),d          ;save msbyte ect.

 010B'  3E 80                   ld      a,ctcvect       ;put vector offset in ctc.
 010D'  D3 20                   out     (ctc),a

 010F'  3E A7                   ld      a,10100111b     ;control word for ctc.
 0111'  D3 20                   out     (ctc),a

 0113'  3E FA                   ld      a,delay         ;time constant for ctc.
 0115'  D3 20                   out     (ctc),a

                        ; this completes the initilization.

 0117'  ED 5E           main:   im 2                    ;set mode 2 interupts.
 0119'  FB                      ei
 011A'  00              crap:   nop
 011B'  C3 011A'                jp      crap            ;sit around and wait for the fireworks.

 011E'  F3              txint:  di                      ;this is the start of the interupt routine.
 011F'  F5                      push    af
 0120'  C5                      push    bc              ;save regs

 0121'  06 7D                   ld      b,packet        ;set up a counter to send 'packet_size'
                                                        ;bytes of data.

 0123'  DB B6           again:  in      a,(fifo)        ;this bit 'sends' packet.
 0125'  D3 B6                   out     (fifo),a
 0127'  10 FA                   djnz    again


                        1
```

```
0129'   C1                      pop     bC
012A'   F1                      pop     af
012B'   FB                      ei
012C'   ED 4D                   reti
                                END
```

Macros:

Symbols:

0123'	AGAIN	011A'	CRAP	0020	CTC
0080	CTCVECT	00FA	DELAY	00B6	FIFO
0117'	MAIN	007D	PACKET	0100'	START
011E'	TXINT				

No Fatal error(s)

```
;
;Stephen Stella   Oct 1984.
;For DESIGN 2 C0335.
;
;This program is fully interupt driven using a ctc and two pio's
;and will allow both half and full duplex communications between
;two boards on different machines.
; In fact its a reasonable simulation of ETHERNET and copes,in a
;crude way, with "collisions" and tries to syncronize the sending
;of packets to avoid such nasties by displacing the ctc tx timer
;in time each time a collision occures.
;For half duplex don't enable the ctc interupts on the receive
;board and for safety inhibit the input pio interupts on the
;transmitting board.


        .Z80

;EQUATES.....................................................................

txlength equ    127             ;tx packet size.
rxlenght equ    127             ;rx packet size.
txpiod   equ    84h             ;i/o address of output port.
txpios   equ    85h             ;status for above.
txpiov   equ    82h             ;offset for interupt vector.
txpiocw  equ    00001111b       ;control byte for mode 0 pio.(output).
rxpiod   equ    84h             ;i/o address of input port.
rxpios   equ    85h             ;status for above.
rxpiov   equ    84h             ;offset for interupt vector
rxpiocw  equ    01001111b       ;control byte for mode 1 pio.(input).
ctc      equ    20h             ;i/o address of ctc.
ctcv     equ    80h             ;interupt vector offset.
ctccw    equ    10100111b       ;control word for ctc.
fifo     equ    182             ;i/o address of fifo's.
inuse    equ    0ffh            ;flag for "packet IN use".
notinuse equ    00h             ;flag for "packet NOT in use".
epio     equ    10000011b       ; enable's pio interupts.
dpio     equ    00000011b       ;disables pio interupts.
delay    equ    250             ;time constant for ctc.
displace equ    0ffh            ;time delay constant.

;END OF EQUATES.............................................

        org     100h
        di                      ;While changing the vectors.
txuse:  db      00              ;flag for tx packet in progress.
rxuse:  db      00              ;same for rx.
txpack: db      00              ;no of bytes left to send.
rxpack: db      00              ;no of bytes left to receive.
```

```
;INITILIZATION ROUTINES.....................................................

init:                           ;initilize the ctc.
        ld      a,i             ;this bit builds the address of
        ld      h,a             ;interupt vector for the ctc and
        ld      l,ctcv          ;puts the address of the transmit
        ld      de,txint        ;routine in it.
        ld      (hl),e
        inc     hl
        ld      (hl),d

        ld      a,ctcv          ;set ctc vector offset.
        out     (ctc),a

        ld      a,ctccw         ;control word for ctc.
        out     (ctc),a

        ld      a,delay         ;time constant
        out     (ctc),a
                                ;next set up tx pio (output pio).
        ld      a,i             ;set up address in txpio
        ld      h,a             ;interupt vector
        ld      l,txpiov
        ld      de,txint
        ld      (hl),e
        inc     hl
        ld      (hl),d

        ld      a,txpiov        ;load txpio int vect offset
        out     (txpios),a

        ld      a,txpiocw       ;set op mode (0-output)
        out     (txpios),a

        ld      a,dpio          ;disable txpio int
        out     (txpios),a

                                ;now set up rx pio (input pio).
        ld      a,i             ;set up rx interupt vector and
        ld      h,a             ;put address of rx routine in it.
        ld      l,rxpiov
        ld      de,rxint
        ld      (hl),e
        inc     hl
        ld      (hl),d

        ld      a,rxpiov        ;load vector offset
        out     (rxpios),a

        ld      a,rxpiocw       ;load control mode (1-input).
        out     (rxpios),a
        ld      a,epio          ;enable rxpio interupts.
        out     (rxpios),a

;END OF INITILIZATION.....................................................
```

```
main:
        im      2
        ei
again:  nop
        jp      again           ;wait for the fireworks.


;TRANSMIT INTERUPT ROUTINE.....................................................

txint:
        di
        push    af

        ld      a,(rxuse)       ;check if we're currently
        cp      notinuse        ;receiving a packet.
        jp      z,ok1
abort:  ;we have a collision here.
        ld      a,dpio          ;diable tx pio interupts.
        out     (txpios),a

        ld      a,notinuse      ;clear tx in progress flag.
        ld      (txuse),a

        ld      a,displace      ;kill some time
kill:   nop
        dec     a
        jr      nz,kill
        call    restart         ;by reseting timer here the next
                                ;tx packet will be later and after
                                ;enough smash's the packets will
                                ;displace in time and aviod each other.
        jr      retn1           ;only have one exit point.

ok1:    ld      a,(txuse)       ;check if we're sent any bytes yet.
        cp      inuse
        jp      z,send1

first1: ;first byte of packet so set flags ect.
        ld      a,txlength      ;set packet lenght.
        ld      (txpack),a

        ld      a,inuse         ;set transmitting flag.
        ld      (txuse),a

        ld      a,epio          ;enable tx pio to interupt
        out     (txpios),a      ;so rest of packet is sent.

send1:  in      a,(fifo)        ;get data.
        out     (txpiod),a      ;send it.

        ld      a,(txpack)      ;dec no of bytes to send.
        dec     a
        ld      (txpack),a
        jp      nz,retn1
```

```
finish: ;if here we're finished sending this packet.
        ld      a,notinuse      ;clear tx flag.
        ld      (txuse),a

        ld      a,dpio          ;disable tx pio interupts until
        out     (txpios),a      ;next packet.

retn1:  pop     af
        ei
        reti

restart:        ;subroutine to restart the ctc timer.
                ;used to offset the timing of packet sending
                ;so as to not collide with the other station.
        ld      a,ctccw
        out     (ctc),a
        ld      a,delay
        out     (ctc),a
        ret

;RECEIVE INTERUPT ROUTINE.....................................................

rxint:
        di
        push    af

        ld      a,(txuse)       ;check if we're transmitting.
        cp      notinuse
        jp      z,ok2

stop:                           ;stop sending and receive packet instead.
                                ;this routine shouldn't ever be called !!!
        ld      a,dpio          ;stop tx pio from interupting.
        out     (txpios),a

        ld      a,notinuse      ;clear tx flag.
        ld      (txuse),a

ok2:    ld      a,(rxuse)       ;check if in middle of packet
        cp      inuse
        jp      z,send2

first2:                         ;if here then this is first byte of pack.
                                ;set flags ect
        ld      a,inuse         ;set rx in progress flag.
        ld      (rxuse),a

        ld      a,txlength      ;set rx packet.
        ld      (rxpack),a

send2:  in      a,(rxpiod)      ;get data
        out     (fifo),a        ;put it away

        ld      a,(rxpack)      ;check if packet is finished.
```

```
        dec     a
        ld      (rxpack),a
        jp      nz,return2

finish2:                        ;here if we're received last byte of packet.
        ld      a,notinuse      ;clear rx flag.
        ld      (rxuse),a

return2: pop    af
        ei
        reti
                                ;go back to main and wait for another interupt.

        end

                                ;that's all folk's.
```

APPENDIX D

MOTOROLA

MC14404
MC14406
MC14407

PULSE CODE MODULATION (PCM) CODEC

The MC14406 and MC14407/4 per channel PCM codecs are designed for 8000-samples-per-second, 8-bit-per-sample voice coding and decoding. These devices are full duplex. The MC14406 provides Mu and the MC14407/4 provides both Mu and A companding laws.

The transmit and receive data rates are independently selectable up to 3.088 MHz allowing direct interface to 24, 30, 32, and 48 channel digital frames.

Both codecs are fabricated using CMOS technology for reliable low power performance. The MC14406 is the full feature device in a 28 pin package. The MC14407/4 provides a 24-pin package without signaling capabilities.

- Per Channel Full Duplex Capability
- Low Power Operation – 80 mW Typ
- Power Down Input (1.0 mW Max in Power Down Mode)
- Pin Selection of A-law and Mu-law Companding (MC14407, MC14404)
- Single Power Supply Operation – 10 to 16 Volts
- Zero Code Supression (MC14406, MC14407)
- Transmit and Receive Signaling Available (MC14406)
- Independent Transmit and Receive Clocks to 3.088 MHz
- Externally Selectable Full Scale
- On-Chip Auto Zero
- Mu-Law Digital Format MC14406, MC14407
- A-Law CCITT Digital Format MC14404

CMOS LSI

(LOW POWER COMPLEMENTARY MOS)

FULL DUPLEX 8-BIT COMPANDED PCM CODEC

24-PIN PACKAGE
CASE 709 - PLASTIC CASE 623 - CERAMIC

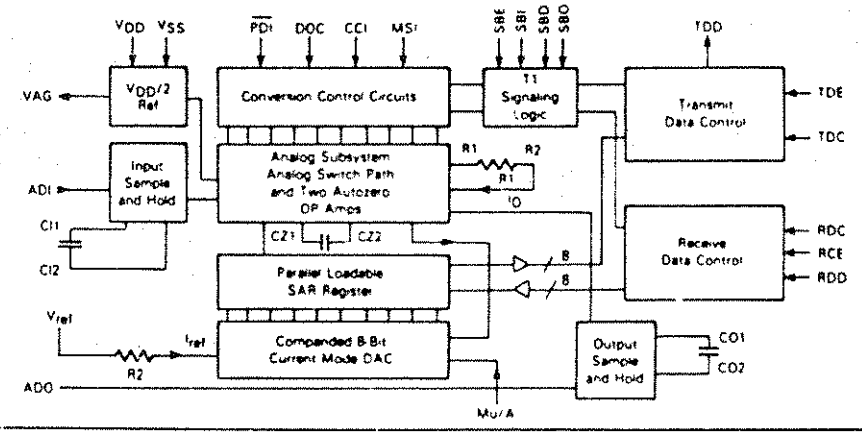
28-PIN PACKAGE
CASE 710 - PLASTIC CASE 733 - CERDIP

L SUFFIX
LEADLESS CERAMIC PACKAGE

ORDERING INFORMATION

MC14XXX — Suffix Denotes
- L Ceramic Package
- P Plastic Package
- Z Chip Carrier

BLOCK DIAGRAM



MAXIMUM RATINGS (Voltages Referenced to V_{SS})

	Symbol	Value	Unit
DC Supply Voltage	V_{DD} V_{SS}	- 0.5 to + 18	V
Voltage, Any Pin to V_{SS}	V	- 0.5 to V_{DD} + 0.5	V
DC Current Drain per Pin (Excluding V_{DD})	I	10	mA
Operating Temperature Range	T_A	- 40 to + 85	°C
Storage Temperature Range	T_{stg}	- 65 to + 150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{DD}$.

Unused inputs must always be tied to an appropriate logic voltage level (e.g., either V_{SS} or V_{DD}).

RECOMMENDED OPERATING CONDITIONS

Parameter	Symbol	Min	Typ	Max	Unit
DC Supply Voltage	V_{DD} V_{SS}	10	12	16	V
Convert Clock Frequency	CCI	128	128	280	kHz
Transmit Data Clock Frequency	TDC	128	2048	3088	kHz
Receive Data Clock Frequency	RDC	128	2048	3088	kHz
Input Sample Capacitor	CI1, CI2		2000		pF
Output Sample Capacitor	CO1, CO2		4700		pF
Unit Step Size = Full Scale/4096*	ADO	0.36	0.85	1.2	mV
Full Scale Voltage (V_{DD} = 10 V)	ADO	2.0		2.5	Vp
Full Scale Voltage (V_{DD} = 12 V)	ADO	2.0		3.5	Vp
Full Scale Voltage (V_{DD} = 15 V)	ADO	2.0	5.0	5.0	Vp
Reference Current Range	I_{ref}	65	80	125	μA
Bypass Capacitor (VAG to V_{DD})	VAG		0.1		μF
Auto Zero Capacitor	CZ1, CZ2		2000		pF

SYSTEM PERFORMANCE A/D THROUGH D/A (R_1 = 30 kΩ, R_2 = 3 kΩ)

Characteristic		Test Figure	0°C Min	0°C Max	25°C Min	25°C Typ	25°C Max	85°C Min	85°C Max	Unit
Gain Repeatability (Part to Part) (ref 1.02 kHz 0 dBm)		ADO	- 0.05	0.55	- 0.15	- 0.3	- 0.45	0.05	0.55	dB
Gain Tracking Error (ref = 0 dBm0 @ 1.02 kHz)	+ 3 to - 37	1, 2, 3	- 0.4	0.4	- 0.3	± 0.1	+ 0.3	- 0.4	0.4	dB
	Input Level (dBm0): - 37 to - 40		- 0.5	0.5	- 0.4	± 0.2	+ 0.4	- 0.5	0.5	
	- 40 to - 50		- 1.0	1.0	- 0.9	± 0.4	+ 0.9	- 1.0	1.0	
	- 50 to - 55		- 3.0	3.0	- 2.5	± 0.8	+ 2.5	- 3.0	3.0	
Signal to Noise Ratio	Input Level (dBm0): + 3 to - 30	1, 2, 3	34		35	38		34		dB
	- 35		31		32	36		31		
	- 40		27		28	30		27		
	- 45		22		24	27		22		
Idle Channel Noise	ADI = VAG	1, 2		12		2	9		12	dBrnC0
Quiet Code Noise	RDO = Zero Code	1, 2		9		2	6		9	dBrnC0
Single Frequency Distortion (Input Level = - 10 dBm0)	1.02 kHz	1, 2, 3		- 40			- 40		40	dBm0
	Out of Band 0 to 12 kHz			- 28			- 28		- 28	
Power Supply Rejection (V_{DD} = 12 V + 0.1 VRMS @ 1 kHz)		1, 2, 3	40		40	50		40		dB

DIGITAL ELECTRICAL CHARACTERISTICS (V_{SS} = 0 V)

Characteristic	Symbol	V_{DD} Vdc	0°C Min	0°C Max	+25°C Min	+25°C Typ	+25°C Max	+85°C Min	+85°C Max	Unit
Operating Current, I_{ref} = 80 µA	I	12	–	9.0	–	8.5	8.0	–	9.0	mA
Power-Down Current (PDI = V_{SS})	I_{PD}	12	–	100	–	20	80	–	100	µA
Input Current										
CCI, RDD, RCE, RDC, TDC, PDI	I_{in}	12	–	±1.0	–	±0.00001	±0.3	–	±1.0	µA
DOC, SBD, MSI, SBI, SBE	I_{inS}		–	+100	–	+30	+50	–	+100	
(Internal Pull-Down Resistors)	I_{inD}		–	±1.0	–	−0.00001	−0.3	–	±1.0	
TDE, Mu/A-Law	I_{inS}		–	±1.0	–	+0.00001	+0.3	–	±1.0	
(Internal Pull-Up Resistors)	I_{inD}		–	−100	–	−30	−50	–	−100	
Input Voltage "0" Level	V_{IL}	12	–	3.6	–	5.25	3.60	–	3.6	V
		15	–	4.0	–	6.75	4.0	–	4.0	
"1" Level	V_{IH}	12	8.4	–	8.4	6.75	–	8.4	–	V
		15	11.0	–	11	8.25	–	11.0	–	
Input Capacitance	C_{in}	12	–	–	–	5.0	7.5	–	–	pF
Output Drive Current SBO										
V_{OH} = 11	I_{OH}	12	−2.0	–	−2.0	−4.0	–	−2.0	–	mA
V_{OH} = 13.5		15	−3.0	–	−3.0	−8.8	–	−3.0	–	
V_{OL} = 1.0	I_{OL}	12	2.0	–	2.0	4.0	–	2.0	–	mA
V_{OL} = 1.5		15	3.0	–	3.0	8.8	–	3.0	–	
Output Drive Current TDO										
V_{OH} = 11	I_{OH}	12	−4.0	–	−4.0	−8.0	–	−4.0	–	mA
V_{OH} = 13.5		15	−8.0	–	−8.0	−17.6	–	−8.0	–	
V_{OL} = 1.0	I_{OL}	12	4.0	–	4.0	8.0	–	4.0	–	mA
V_{OL} = 1.5		15	6.0	–	6.0	17.6	–	6.0	–	

ANALOG ELECTRICAL CHARACTERISTICS (V_{DD} = 12 V)

Characteristic		Symbol	0°C Min	0°C Max	+25°C Min	+25°C Typ	+25°C Max	+85°C Min	+85°C Max	Unit
Analog Ground Voltage		VAG	5.95	6.35	6.0	6.1	6.3	5.95	6.35	V
Source Current	V_{OH} = 5.9 Vdc		100	–	100	200	–	100	–	µA
Sink Current	V_{OL} = 6.4 Vdc		2.5	–	2.5	5.0	–	2.5	–	mA
Output Impedance	AC (1.02 kHz – 30 dBm)		–	–	–	50	–	–	–	Ω
Input Offset Voltage		I_{ref}	–	±50	–	–	±50	–	±60	mV
Offset Voltage Drift			–	–	–	±30	±100	–	–	µV/°C
Input Impedance dc		ADI	–	–	1.0	10	–	–	–	MΩ
AC Impedance CI = 2000 pF, 1 kHz			–	–	–	80	–	–	–	kΩ
Lower Common Mode Range			–	–	–	0.5	1.0	–	–	V
Upper Common Mode Range			–	–	9.8	10.3	–	–	–	V
Sample Duty Cycle	Full Duplex	ADO	–	–	–	50	–	–	–	%
Sample Duty Cycle	Decode only 14406		–	–	–	12.5	–	–	–	
Neutral Offset (From VAG)		ADO	−50.0	+70.0	−50.0	±25	+70.0	−50.0	+70.0	mV
Neutral Offset Drift			–	–	–	±30	±100	–	–	µV/°C
Source Current	V_{OH} = 9.5 V		500.0	–	500.0	200	–	500.0	–	µA
Sink Current	V_{OL} = 2.5 V		10.0	–	1.0	3.0	–	1.0	–	mA
Lower Common Mode Range			–	–	–	0.5	1.0	–	–	V
Upper Common Mode Range			–	–	9.8	10.3	–	–	–	V
Output Impedance	AC (1.02 kHz – 30 dBm)		–	–	–	100	–	–	–	Ω
Setting Time to 3.0 V R and F			–	–	–	0.3	–	–	–	µs

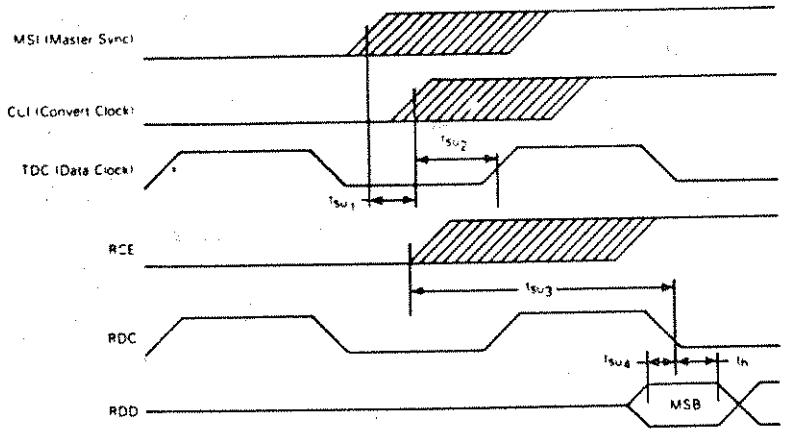
C-7

SWITCHING CHARACTERISTICS (V_{DD} = 12 V, C_L = 50 pF)

Characteristic		Symbol	0-70°C Min	0-70°C Typ	0-70°C Max	Unit
Output Rise Time Output Fall Time	TDD, SBO	t_{TLH} t_{THL}	–	30	80	ns
Input Rise Time Input Fall Time	CCI, RCE, RDC, TDC, MSI	t_{TLH} t_{THL}	–	–	4	µs
Pulse Width	CCI, RCE, RDC, TDC, MSI	t_{WH}	100	50	–	ns
Clock Pulse Frequency	TDC, RDC CCI	f_{CL}	64 100	– –	3088 280	kHz
Propagation Delay Time	TDC to TDD TDE to TDD	t_{PLH} t_{PHL}	– –	90 45	180 90	ns
Setup Time MSI Rising Edge to CCI Rising Edge	(TDD, RDC, CCI = 128 kHz)*	t_{su1}	– 3	–	+ 3	µs
CCI Rising Edge to TDC Rising Edge	(TDC = 2.048 MHz)* (TDC = 1.544 MHz)*	t_{su2}	– 125 – 200	– –	+ 225 + 300	ns
RCE Rising Edge to RDC Falling Edge	(RDC = 2.048 MHz) (RDC = 1.544 MHz)	t_{su3}	120 120	– –	400 560	ns
RDD to RDC Falling Edge		t_{su4}	80	40	–	ns
Hold Time	RDD to RDC Falling Edge	t_h	120	60	–	ns

*Specifications assume use of 50% duty cycle for clocks

TIMING DIAGRAM



NOTE Internal circuitry in the MC14406/7 is designed to tolerate clock skewing between CCI, TDC and MSI. The MSI and CCI edges are synchronized to the following TDC edge.

FIGURE 1 – TEST CIRCUIT
MC14408 SYSTEM PERFORMANCE DATA

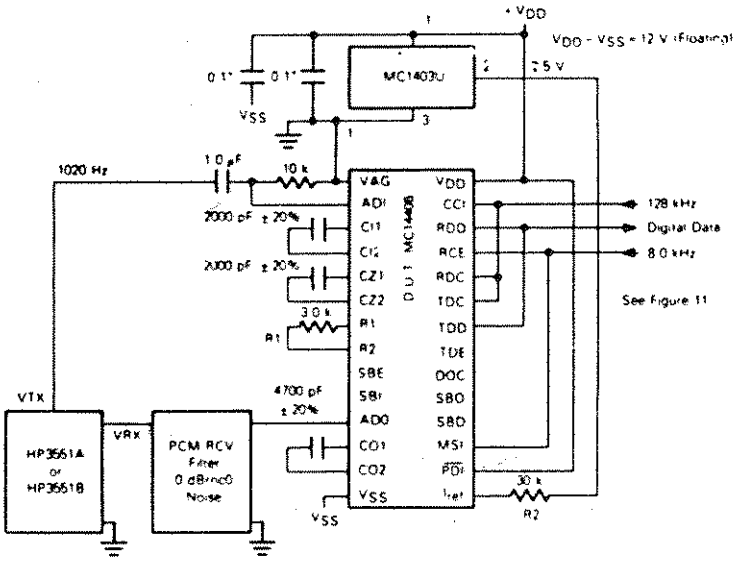
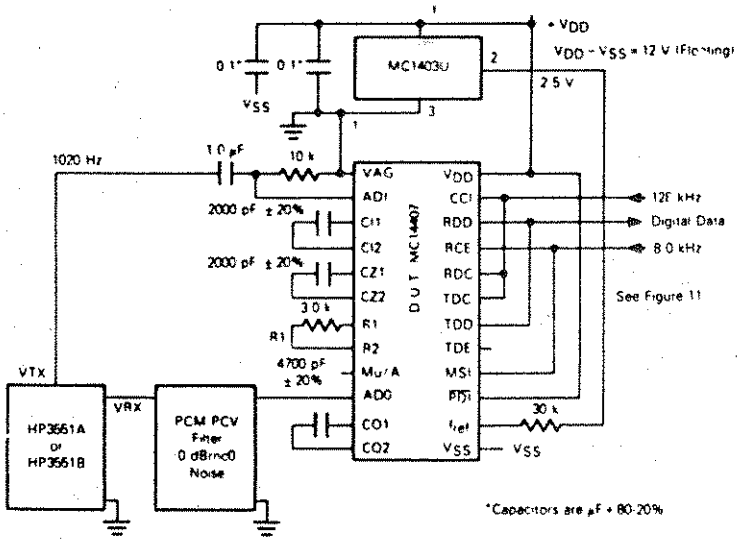


FIGURE 2 – TEST CIRCUIT
MC14407 SYSTEM PERFORMANCE DATA



*Capacitors are μF ± 80-20%

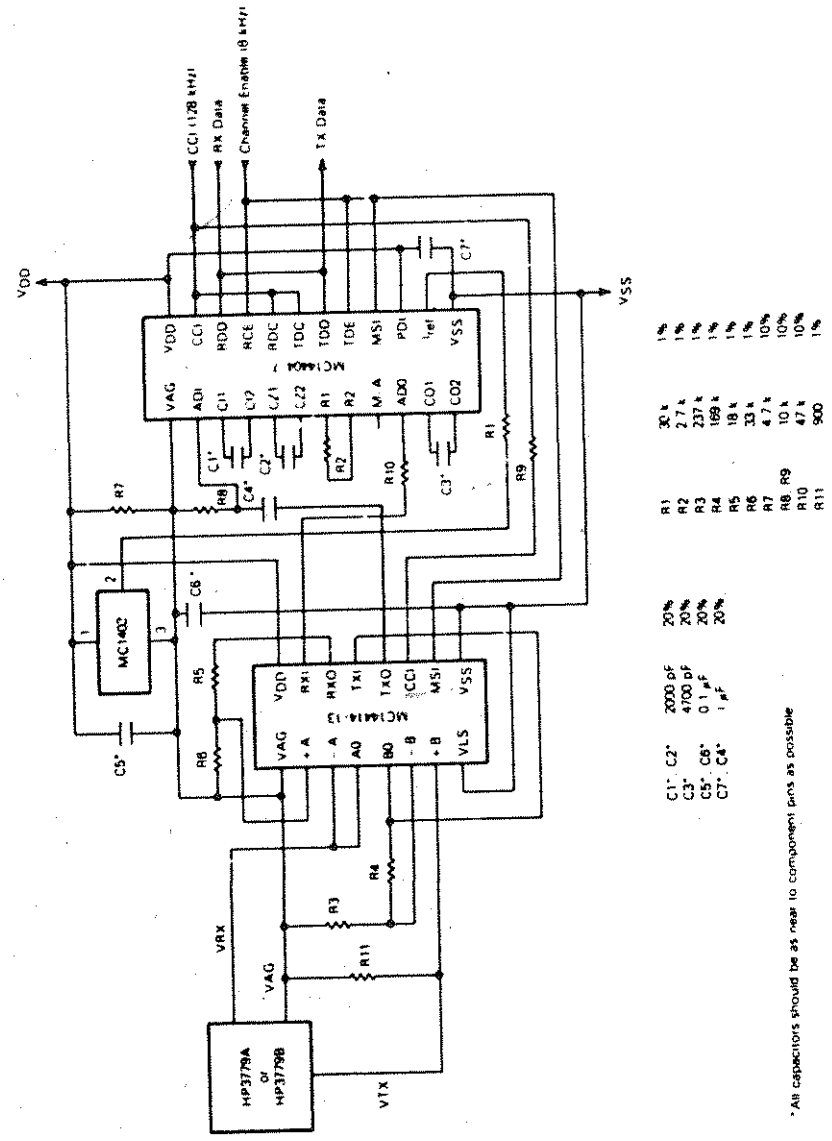
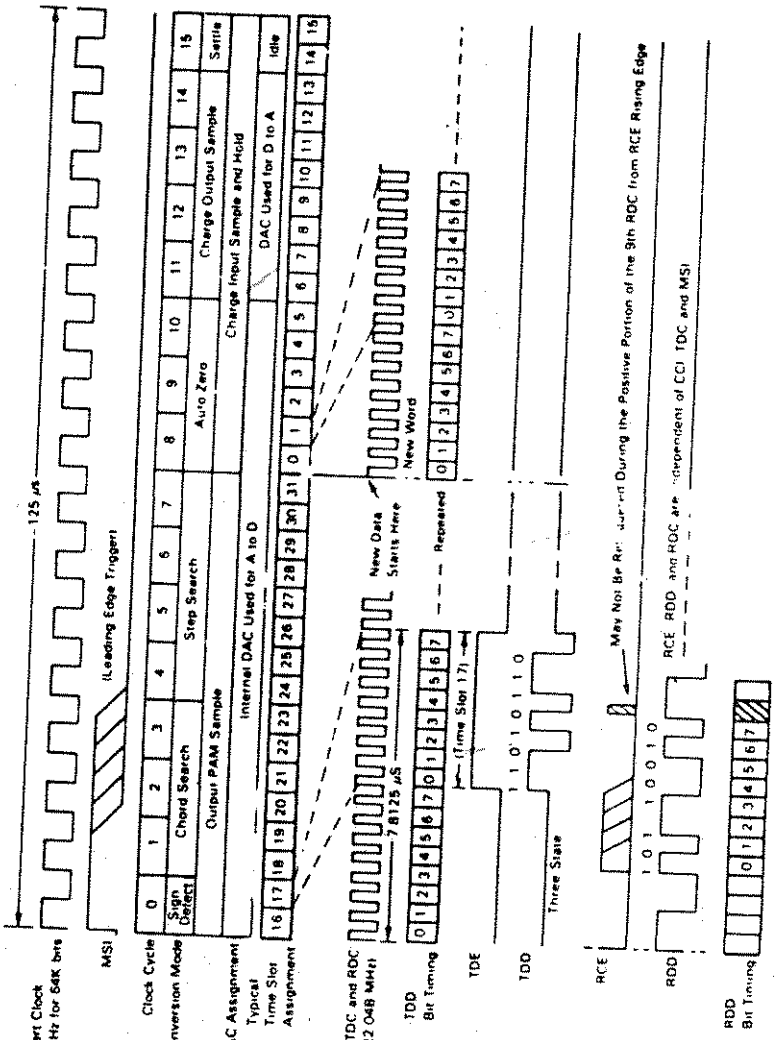


FIGURE 3 – ANALOG TRANSMISSION TEST CIRCUIT FOR MC14414/13 PCM FILTER AND MC14407/4 PCM CODEC

4

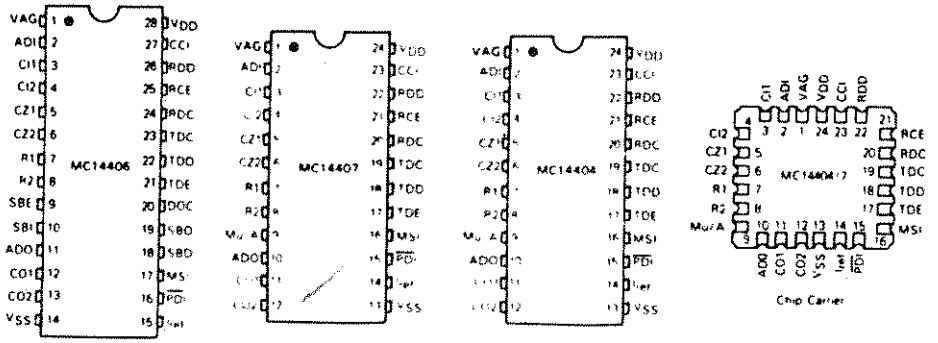
FIGURE 4 — CONVERSION AND DATA TIMING FOR 64K BIT CHANNEL IN 2.048 MEGABIT FRAME



MOTOROLA

MC14408

MC14404 • MC14406 • MC14407



VAG — Analog Ground Output (V_{DD}-V_{SS})/2. VAG is the output of the internal (V_{DD}-V_{SS})/2 voltage reference and is controlled by $\overline{PDI}$. A pull-up load resistor and bypass capacitor may be required if this output is to be used elsewhere in the system. VAG outputs may be wire ORed together with up to 8 other codecs. This pin serves as the analog ground for the codec.

ADI — Analog Data Input. The band limiting input filter between the channel unit hybrid and the codec drive pin 2. Driver source impedance should be 600 Ω or less. This is the voice input to the codec and will be sampled at 8.0 kHz. The sample aperture ends with MSI.

CI1, CI2 — Input Sample Capacitor.* The input sample capacitor is connected to these pins. A 2000 pF capacitor is recommended. X7R ceramic capacitors are satisfactory.

CZ1, CZ2 — Auto Zero Capacitor.* The auto zero circuit requires a 2000 pF capacitor between these pins. An X7R ceramic capacitor is satisfactory.

R1, R2 — Gain Resistor. The V to I conversion resistor is connected between these pins. Gain is established by the ratio of this resistor and the resistor at pin I_{ref}.

Mu/A — Mu-Law/A-Law Select (Internal Pull-Up). Selection of A-law or Mu-law companding is provided. The MC14407 has an internal pull-up device to select Mu-Law while the MC14404 has an internal pull-up to select A-law. Both can be altered by a logic level at this input. The MC14406 is fixed using Mu-law. See TDD description for digital format.

SBE — Signal Bit Enable (Internal Pull-Down). SBE controls the insertion of transmit signaling bits into the transmit data register. When taken high, the next transmit word will contain the SBI pin level in the LSB position rather than the last bit of the PCM word. If kept high, the SB data will be inserted in succeeding conversions until one conversion after it is brought low. It can be used for two-hook signaling or A and B signaling in D3 banks.

SBI — Signal Bit Input (Internal Pull-Down). SBI is the data input for transmitting signaling bits. The level of SBI will be latched on the leading edge of SBE, or by the internal latch data pulse if SBE is held high. If SBE is pulsed, the leading edge of SBE will latch SBI and load it in to the next transmit word.

*Keep capacitors very near the pins.

ADO — Analog Data Output. ADO outputs the received PAM sample for eight convert clock cycles beginning with MSI. It then returns to the output neutral voltage which may be a few millivolts different from VAG.

CO1, CO2 — Output Sample Capacitor.* The output sample capacitor is connected between these pins. Ceramic Capacitor of 3900 pF is recommended. X7R ceramic capacitors are satisfactory.

V_{SS} — Most Negative Supply. This is the most negative supply pin, and the digital ground. All digital inputs and outputs will swing the full supply voltage.

I_{ref} — Current Reference Input. A reference current of 65 to 125 μA sets the full scale DAC current. An 83 μA input current corresponds to 1.20 mA full scale DAC current pin R2.

$\overline{PDI}$ — Power-Down Input. A low level at this input deactivates the codec. The analog circuitry is based off and the digital clock inputs are disabled. The power dissipation in this mode is less than 0.5 mW typ.

The codec will be totally powered down after the MSI pulse that follows the $\overline{PDI}$ negative transition. When $\overline{PDI}$ is taken high, the codec will be fully operational following the second MSI pulse (between 125 and 256 μs if MSI is @ 8 kHz). In the powered down mode, VAG is disabled and pulled up to V_{DD} which can be used to disable all loads connected between VAG and V_{DD}.

MSI — Master Sync Input (Internal Pull-Down). The MSI leading edge resets the entire chip to the initial cycle (0000). The chip continues operation on the next leading edge of data and convert clock. The eighth leading edge of convert clock (defined from MSI) resets the output data multiplexer to the transmit word sign bit.

SBO — Signal Bit Output. SBO outputs the LSB of the receive data register. The LSB may be sampled externally during the first three convert clocks after MSI or used as a trigger pulse in off hook applications.

SBD — Signal Bit Decode (Internal Pull-Up). Signal bit decode allows the control of the ½ LSB or LSB centering required in sample decode cycles. It is loaded by the RCE edge in each cycle. If a zero is entered, the next decode cycle adds ½ LSB to the 8-bit received word to center the quantization error of the 8-bit received sample when the output sample capacitor is charged. If a 1 is loaded by RCE, the ½ LSB is

C-12

not added, and the LSB is forced to a 1 to form a centered error 7-bit output, and the LSB is ignored and assumed to be signaling information. (For codes 1111111X or 0111111X, the LSB is disabled for SBD high.)

DOC – Decode Only Control (Internal Pull-Down). DOC is normally tied low. When high, it configures the device to skip the eight convert clocks used for A-to-D conversion. The operating cycle is then two clocks for PAM output at ADO and six clocks for charging the next output sample. The device may be used to do as many as eight decode cycles in a 125 μs period with an external transmission gate selector. (MC14406 only)

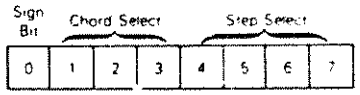
Received Data is loaded into the transmit data register at MSI when in DOC mode, so DOC can be used as a digital loop back control.

TDE – Transmit Data Enable (Internal Pull-Up). The Transmit Data Enable is a three-state control for the transmit digital output. A number of codec outputs can be interleaved into a serial stream by connecting the outputs and controlling TDE with a 1 of N decoder. It will provide switching characteristics capable of 3.088 MHz operation.

TDD – Transmit Digital Data. The Transmit Digital Data rate is controlled by the data clock input, and is frame aligned with the Master Sync input. If data clock is at 1.544 MHz and convert clock is at 128 kHz, the new sign bit will be output beginning with the leading edge of convert clock 8, and the 8-bit word repeated throughout the convert cycle at the data clock rate.

Two digital data formats are used. The MC14406 and MC14407 use Mu-law format while the MC14404 uses A-law CCITT format. Conversion data is shown in the table below.

Code	Sign/ Magnitude		Mu-Law		A-Law (CCITT)	
+ full scale	1111	1111	1000	0000	1010	1010
+ zero	1000	0000	1111	1111	1101	0101
– zero	0000	0000	0111	1111	0101	0101
– full scale	0111	1111	0000	0010	0010	1010



NOTE: Starting from sign magnitude, to change format:

To Mu Law –
- MSB is unchanged (sign)
- invert remaining seven bits
- if code is 0000 0000, change to 0000 0010 (for zero code suppression)

To A Law –
- MSB is unchanged (sign)
- invert odd numbered bits
- ignore zero code suppression

TDC – Transmit Data Clock. TDC sets the digital data rate of the codec. The transmit data stream will provide a continuous repetition of the current transmit data word at the TDC bit rate, beginning with MSB first and synchronized with the last MSI. The new data word is loaded and serially output at mid-cycle or on the leading edge of convert clock 8.

RDC – Receive Data Clock. RDC controls the receive data register. It clocks the receive data register on the trailing edge under the control RCE. It is often connected to TDC.

RCE – Receive Clock Enable. The rising edge of RCE triggers the receive data register to accept a new data input. After the rising edge of RCE, the data on RDD is loaded into the Receive Data Register on the next eight trailing edges of RDC. The ninth clock transfers the new 8 bit word to an internal intermediate register and frees the Receive Data Register for a new RCE.

RDD – Receive Digital Data. RDD is the input to the receive data shift register. It is controlled by RCE. The register is clocked on the trailing edge of RDC. The data format is the same as that of TDD. See data timing.

CCI – Convert Clock Input. CCI controls the conversion sequence. A 128 kHz clock will produce 64K bits/s full duplex operation and a 256K bits/s clock will produce 64K bit operation for two channels. Sixteen clocks represent one chip cycle from MSI to MSI.

V_{DD} – Most Positive Supply. V_{DD} is typically 12 V with an operation range of 10 V to 16 V. All logic outputs swing the full supply voltage.

PCM CODEC DESCRIPTION

The MC14406 and MC14407 4-PCM Codecs are designed for use in digital voice transmission and switching applications. They are compatible with both telecommunication standards. The MC14406 provides Mu-law companding with LSB signaling while the MC14407-4 allows A-law or Mu-law companding by pin selection. The MC14407-4 does not offer LSB signaling. Both are fabricated on a monolithic CMOS circuit including both linear and digital elements.

Transmit and receive data controls handle digital I/O interface using independent transmit and receive data clocks. The remainder of the circuit is controlled by a 128 kHz convert clock (CCI). The input and output sample and hold circuits, constructed with CMOS transmission gates, require non-precise external capacitors.

The analog subsystem provides autozeroing and signal path control to and from the 8-bit companded DAC. An 8-bit parallel loadable successive approximation register manipulates the DAC bits during both transmit and receive. Additional logic is provided for signaling bits, power down, and digital loop back.

DIGITAL-TO-ANALOG CONVERTER

An 8-bit companded DAC is required to perform Mu- or A-law pulse code modulation. In the MC14406/7 an array of 256 devices is arranged in groups binarily weighted from 128 to 1. A current mirror connection is used such that all the gates and sources of the n-channel array are common. A precise reference current is supplied to the I_{ref} pin by an external voltage reference and resistor R (see test circuits). This current magnitude is ratioed into currents proportional to the SAR register contents. This is the current that is allowed to flow into the R2 pin for both encode and decode functions. The companded chord offset voltages are produced by switching currents of the 256-unit array into the R2 output.

A second DAC array of 64 devices is used to establish the steps within the range of a single chord. A 3-bit DAC allows ½ LSB centering. Prior to performing a conversion, all weighted drains are ON with all currents switched into an in-

ternal "dump" bus. During the three clock cycles for chord search, the unneeded chord DAC currents are turned off. The crucial switch changes thus redirect currents that are completely settled.

One major advantage of a current steering DAC of this type is that it may be statically tested with 8 bit words applied through the receive data logic of the codecs (see static DAC test). The output current of the DAC is available at the R2 pin of the codec and the DAC's static accuracy is testable after product assembly. An externally I to V converter at P2 bypasses the zeroing circuitry allowing fully static testing (see Figure 10).

THE ANALOG SUBSYSTEM

The MC14406/7 PCM Codecs utilize the DAC network for both encoding and decoding functions because of the relatively large die area involved in the DAC structure. Thus a signal path from the analog input to the DAC and to the analog output from the DAC must be created. Two absolute value sample and hold circuits, an op amp and a comparator accomplish these functions. Analog switches are used extensively to reorganize the analog components into the various required modes of operation (see Figure 6).

Initially the input is tracked and then sampled and held using AG1 and AG2. In Figure 6a, resistor R1 converts the held voltage to a current which flows into the DAC. An eight cycle successive approximation beginning with sign detection produces the PCM word. The junction between the DAC and resistor is sensed by the comparator to make each coding decision. At mid frame the new 8 bit word is transferred to an output latch and the analog system is reconfigured for autozero (Figure 6b). In this mode, both the comparator and the op amp function in unity gain to charge the zero capacitor with the sum of their offsets. This zeroing voltage is then added in series with the two amplifiers in the other configurations. This zeroing method appears static externally and avoids idle noise problems associated with the sign bit averaging schemes used in earlier codec designs. Signals ac coupled to the codec exhibit no zeroing error and encounter no bias line dither.

Next, the received 8 bit word is loaded into the DAC and the output is charged to the proportional (positive) voltage (Figure 6c). A1 is then configured as an I to V converter and A2 serves as a unity gain follower. The sign is restored in the output sample and held in a similar manner as the input switches AG1-5. The capacitance at the input of A1 and A3 is minimized to eliminate charge loss on C1 and C0 when the polarity of these capacitors is reversed. The ratio of external R1 and R2 determines full scale by:

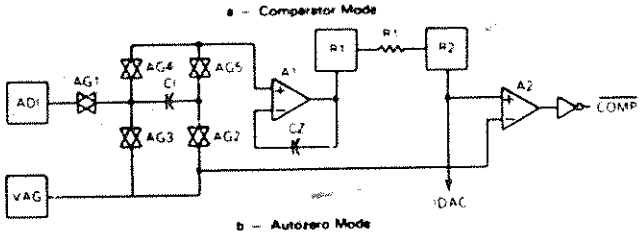
14.00 V_{ref} R1/R2 = Full Scale Voltage in Mu Law

14.05 V_{ref} R1/R2 = Full Scale Voltage in A Law

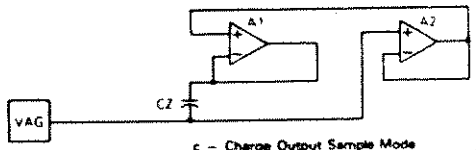
This analog organization minimizes the amplifiers required and minimizes the clock frequencies required for control. A 64K bit full duplex operation is achieved with a 128 kHz convert clock that segments the frame into sixteen 7.8 µs cycles. The entire codec can function at 64K bits full duplex with a single 128 kHz clock or can use 3.088 MHz data clocks for high speed serial data streams.

FIGURE 6 - ANALOG SUBSYSTEM MODES

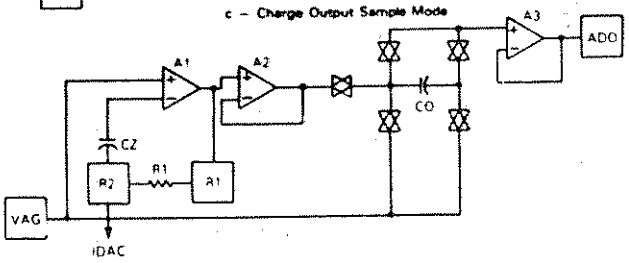
a – Comparator Mode



b – Autozero Mode



c – Charge Output Sample Mode



C-14

APPLICATIONS SECTION

Figures 7 through 12 highlight performance and provide application information for Motorola's semiconductors for digital switching. Figure 7a is a performance table for end to end tests on the MC14407 (codec) and MC14413 (filter). Figure 7b contains the test circuit used to measure this performance. In Figure 8 are plotted the key parameters of gain tracking, quantizing distortion and frequency response. Figures 9a and 9b are the actual layout and component placement for the test circuit used to make the performance measurements. Figure 10 is the test fixture for making the static test of the DAC on the MC14404/06/07 codecs. To help simplify generation of clocks for the codec, clock circuits are shown in Figure 11. Finally, in Figure 12 is shown a schematic of the first implementation with monolithic LSI of a complete telephone line circuit.

A **End-to-End Performance.** The Motorola codec (MC14404/06/07) and filter (MC14413) performance data is shown in Figure 7a. This table contains the Bell System (PUB 43801) and CCITT (G.712) specifications for 4-wire voice frequency parameters. In the typical column is the measured performance of the Motorola codec/filter pair. This data was taken using the Motorola demo board whose schematic is shown in Figure 7b. All of these measurements were taken "End to End." For example, to make a gain measurement for gain tracking, insert a 0 dBm signal with a frequency of 1 kHz into the transmit port of channel A and measure a signal at the receive port of channel B. The gain is then the difference in levels between the transmit and receive ports.

An end to end measurement of this type exercises 2 codecs and 2 filters in 2 separate channels and requires consistent part to part variations to obtain the performance shown in Figure 7a. As can be verified by the table, these components meet the industry standard specified performance.

In Figure 8, the key analog performance parameters are graphically demonstrated against the requirements of the two industry standard specifications. These measurements were also made with the circuit of Figure 7b. The graphs show compliance with these key parameters.

To facilitate customer design using the Motorola components, the layout and parts location for this test circuit is shown in Figure 9. A customer can save time by using this layout as a model for his design as this component placement has been used for these measurements. This codec filter cell was not optimized for component density as it is a test fixture. An actual production line card codec/filter cell can be 1/4 the size of the demo card.

B **Static Testing of the DAC for the MC14404/06/07 Codecs.** Motorola's codec is designed to allow analog parametric tests at probe, and final testing with static techniques. What this means is that this component's DAC performance can be measured using industry standard digital test equipment. Figure 10 shows the test fixture required to measure this performance.

The component is tested by tying CCI, RDC and TDC to a single clock (128 kHz typical). Data is then inserted into RDD pin. This data corresponds to a PCM data word, for example, 1111 1111 for Mu-Law + zero. The rising edge of CCI, the RCE (chip enable) is typically an 8 kHz signal. Every frame, a new PCM word is sequentially clocked in. At the R2 output, the current corresponding to the PCM word is then measured and compared to a standard theoretical DAC. This type of test is very beneficial from both a cost and technical point of view. Statically verifiable accuracy to a fixed standard inherently assures half channel performance and DAC monotonicity.

C **Simplified Clock Circuits.** In Figure 11 two circuits which generate the clocks required by most telephone systems are shown. Figure 11a generates the 128 kHz CCI and the 8 kHz MSI signal that the Motorola codec filter system utilized for timing from 1.544 MHz system clock. In many cases MSI and FST (Frame Sync) are the same, however, the configuration is a simple method to generate the 1.536 MHz clock which is required in some systems. The most economical method of implementing this clock's circuit is to share the function over many line circuits. This is best accomplished by performing the function in the common control board.

In Figure 11b the circuit for generating CCI and MSI from a 2.048 MHz system clock is shown. As with the 1.544 MHz case, the MSI and FST could be tied together, but the configuration shown reduces group delay.

D **Motorola's Subscriber Channel Unit.**

1. The schematic shown in Figure 12 is the industry's first monolithic LSI subscriber channel unit. This circuit contains the following components and functions:
   a) The MC14407 (codec) performs voice coding and decoding.
   b) The MC14413 (filter) provides 60 Hz rejection, transmit anti-aliasing, and receive restoration filter functions. In addition, two free op-amps on the MC14413 are used for transmit gain adjustment and ring trip.
   c) The 2-4 wire conversions, battery feed, secondary lightning protection, line fault protection and signal balance are performed by the SLIC (subscriber loop interface circuit) which includes the MC3419, MJE270/271 and MDA220.
   d) The controller for the line circuit is the MC14418 which provides the capability for variable time slot assignment and line circuit supervision under external microprocessor control.

   The circuit in Figure 12 provides all of the basic functions required for a single party telephone line.
2. Since these devices were all designed to function together, this system offers the user various unique features.

D-1

a) **Signal Level and Transhybrid Rejection.** The system design provides for independent gain adjust for the receive and transmit paths. The resistor RVTX and the op-amp "A" perform the transmit gain adjust and the I to V conversion necessary with the MC3419. The receive gain is set by RX. Adjustment of these two resistors compensate for all component gain variations in the channel unit.

The transhybrid rejection is controlled by a "balance" passive element (RB in Figure 12). For resistive loop impedances, a single resistor provides for balance and for reactive loop impedance, a reactive element can be used. Perfect balance occurs for only one loop impedance, however, a selection of one or two balance impedances is adequate to compensate the normal range of subscriber loops. (See MC3419 data sheet for details.)

b) **Power Considerations.** The system is designed to power down when the phone is on-hook. Since the phone is on-hook 95% of the time in most central office applications, this is a very key feature. In the power down mode, the entire circuit draws less than 10 mW. The power down control logic is supplied from the MC14418 controller (pin Q0 in Figure 12). This logic level is routed to the MC3419 and MC14407 power down pins, however, it is not required to route a write to the MC14413 filter as the codec powers down the filter when the codec itself powers down.

All three devices are fabricated in standard metal gate CMOS, therefore in the power up mode, the MC14407, MC14413, and MC14418 draw only 140 mW. The total circuit dissipation is controlled by the loop length. In a short loop condition 5.5 watts are required to power the telephone (if loop current limiting is not used). The two power transistors dissipate 5 watts of this power and the MC3419 only dissipates 500 mW. This partitioning takes advantage of standard low cost packages which results in the best technical and economical solution.

c) **Ring Trip/Ring Enable.** The combination of the MC3419, MC14418 and MC14413 devices together provide these two supervision features for the line circuit. Under processor control, the MC14418 sets the ringing relay to ring the phone. This is done through the output Q2. When the subscriber goes off hook, the DC current change in the tip lead is sensed at the TS pin on the MC3419. Using one of MC14413 op-amps, a ring trip signal is fed back to the MC14418 which toggles R2 to reset (pin Q2). Thus the ring trip function is implemented without interrupting the control processor.

d) **Tone Plant.** The control of supervision tones like dial tone is again handled by the MC14418 controller. A bit set by the MC14418 (Q1 in Figure 12) selects between the receive highway and the tone plant bus through a transmission gate.

e) **Backplane Design.** The Motorola system concept allows for a completely parallel backplane. The circuit in Figure 12 achieves this. Every wire on the backplane is common to every other channel in the group. This concept greatly reduces assembly cost and backplane wiring. The design is also quite frugal with the number of contact pins in the edge connector. A quad line card constructed using the Motorola devices requires only 22 pins.

f) **Density.** Extraneous gates, op-amps, and external components are avoided for the line circuit due to the system design of the components. This results in a dense pc board layout. Four copies of the schematic shown in Figure 12 have been laid out on a 7" x 7" PC card. The user can also take advantage of improved stacking density in his system due to the low profile of the SLIC. Traditional approaches to the SLIC problem using transformers resulted in board spacing of 1½". With the Motorola SLIC, board to board spacing of ¾" are achievable, thus the user can double the number of lines in his cabinet.

4

D-2

FIGURE 7a — TYPICAL END TO END PERFORMANCE OF MOTOROLA CODEC AND FILTER

All measurements made using HP3779B PCM Test Set

Specifications	Typical Performance of MC14407/4 Codec and MC14413 Filter	Bell System D4 Voice Freq. Requirements PUB 43801	CCITT G7.12 Voice Freq. Requirements
Channel Saturation	+ 3 dBm0	+ 3 dBm0	+ 3 dBm0
Gain Tracking with 1 kHz Tone + 3 to − 40 dBm0 − 40 to − 50 dBm0 − 55 dBm0	 ± 0.2 dB ± 0.3 dB ± 0.5 dB	 ≤ ± 0.5 dB ≤ ± 1.0 dB ≤ ± 3.0 dB	 ≤ ± 0.5 dB ≤ ± 1.0 dB ≤ ± 3.0 dB
Quantizing Distortion @ 1 kHz + 3 to − 30 dBm0 − 35 dBm0 − 40 dBm0 − 45 dBm0	 37 dB 34 dB 31 dB 26 dB	 ≥ 33 dB ≥ 30 dB ≥ 27 dB ≥ 22 dB	 > 33 dB ≥ 30 dB ≥ 27 dB ≥ 22 dB
Idle Channel Noise with VTX = VAG Quiet Code Noise (all 1's at decoder (RDD) input) Selective Response @ multipleS of 8 kHz	16 dBrnc0 10 dBrnc0 − 6[illegible] dBm0	≤ 23 dBrnc0 ≤ 15 dBrnc0 See Frequency Response	≤ − 65 dBm0P ≤ − 75 dBm0P ≤ − 50 dBm0
Frequency Response @ 0 dBm0 Input relative to 1.02 kHz or 0.8 × 20 kHz 50 Hz gain 60 Hz gain 200 to 300 Hz Ripple 3400 Hz gain 4000 Hz gain ≥ 4600 Hz gain	 − 28 dB − 24 dB ± 0.20 dB − 1.0 dB − 32 dB < − 62 dB	 – ≤ − 20 dB ≤ ± 0.3 dB ≥ − 3.0 dB ≤ − 28 dB ≤ − 60 dB	 ≤ − 24 dB – ≤ ± 0.5 dB ≥ − 1.8 dB ≤ − 28 dB ≤ − 60 dB
Single Frequency Spurious Response In band with input 1 kHz @ 0 dBm Out of band with input 0 to 12 kHz @ 0 dBm	 ≤ − 44 dB ≤ − 32.5 dB	 ≤ − 40 dB ≤ − 28 dB	 ≤ − 40 dB ≤ − 25 dB
Differential Delay Distortion 1150 to 2300 1000 to 2500 900 to 2700	 56 μs 72 μs 91 μs	 ≤ 60 μs ≤ 100 μs ≤ 200 μs	 – – –

D-3

FIGURE 8 – KEYPAD TO PULSE DIALER FLOW DIAGRAM

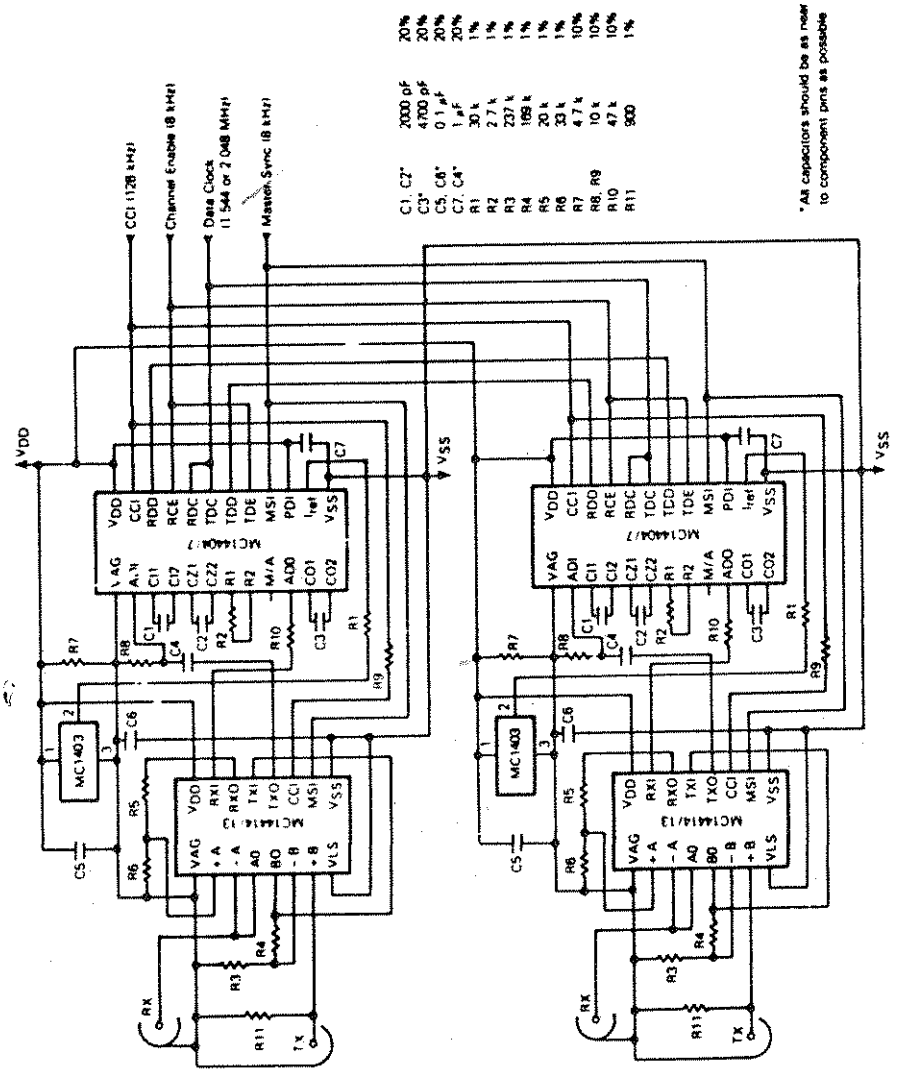


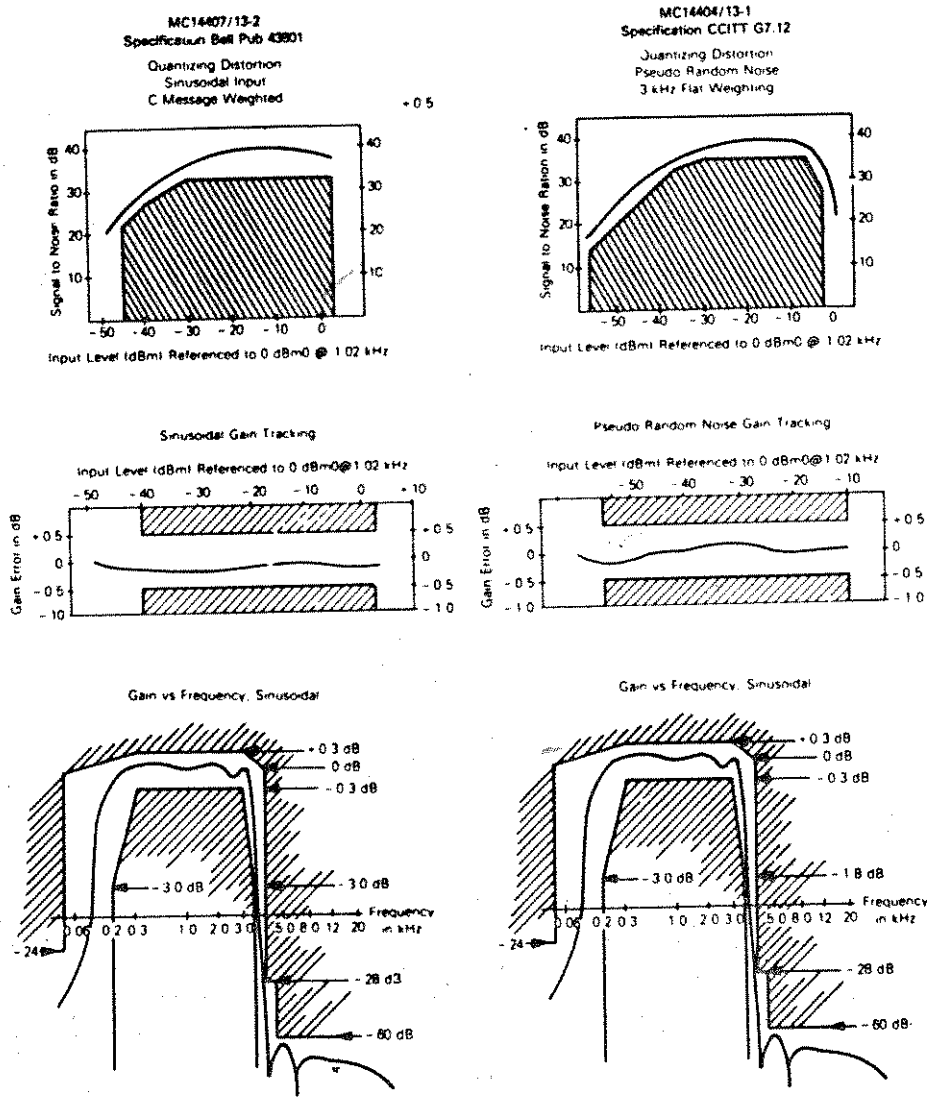
FIGURE 7b – MOTOROLA CODEC FILTER EVALUATION BOARD

D-4

FIGURE 8 – STANDARD K-500 TELEPHONE

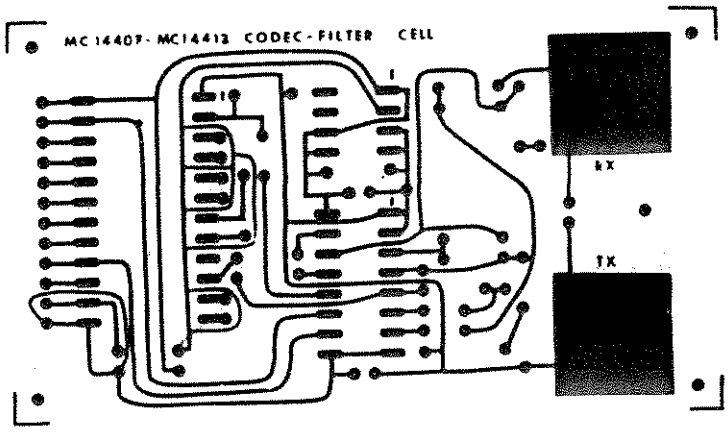
Dial Rotated

FIGURE 8 – TYPICAL END TO END CHANNEL PERFORMANCE FOR MOTOROLA MC14413/14-MC14404/7 CODEC AND FILTER

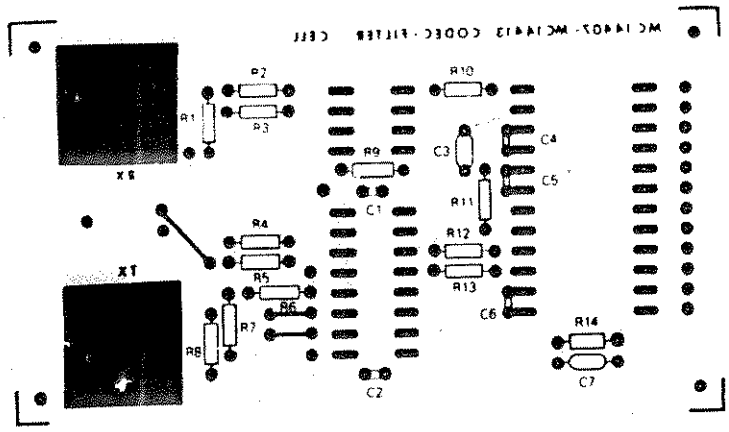


4

FIGURE 9 – MC14407-MC14413 CODEC-FILTER CELL



Bottom View



Top View

R1	909	1/8W	1%
R2	100 k	1/8W	1%
R3	180 k	1/8W	1%
R4	200 k	1/8W	1%
R5	37 k	1/8W	1%
R6	245 k	1/8W	1%
R7	150 k	1/8W	1%
R8	909	1/8W	1%
R9	4.7 k	1/8W	5%
R10	10 k	1/8W	5%
R11	2.664 k	1/8W	1%
R12	47 k	1/8W	5%
R13	2.2 k	1/8W	5%
R14	30.1 k	1/8W	1%

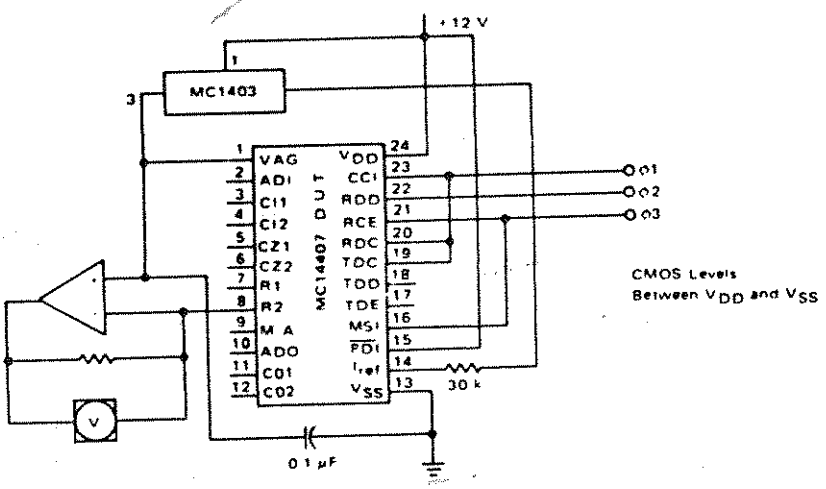
C1	0.1 μF
C2	0.1 μF
C3	1 μF
C4	2200 pF
C5	2200 pF
C6	3900 pF
C7	1 μF

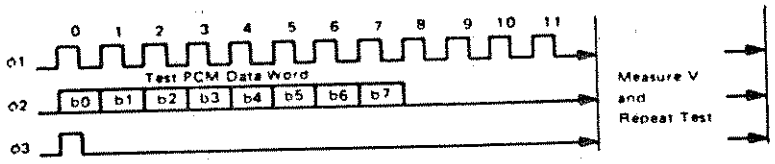
Note: Diagrams are 2X actual size

MAXIMUM RATINGS (Voltages referenced to V_{SS}, Pin 8.)

Rating	Symbol	Value	Unit

FIGURE 10 — STATIC DAC TEST CIRCUIT FOR MC14407 PCM CODEC

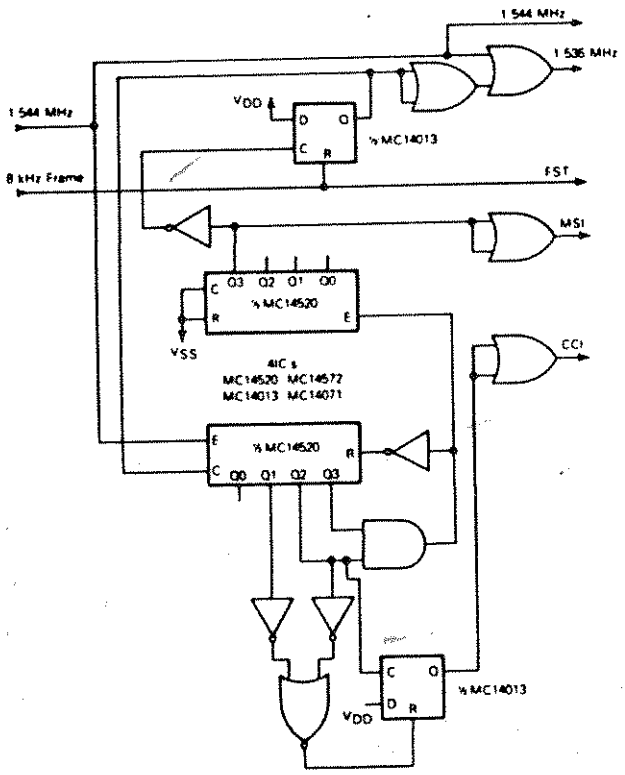




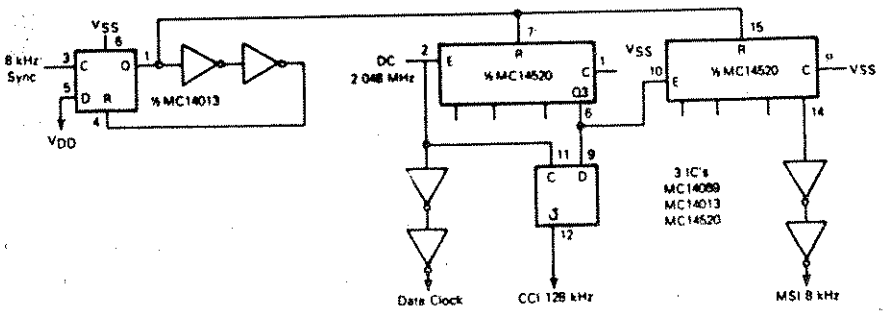
C-8

FIGURE 11

a — Clock Generator for 1.544 MHz System



b — Clock Generator for 2.048 MHz System



D-8

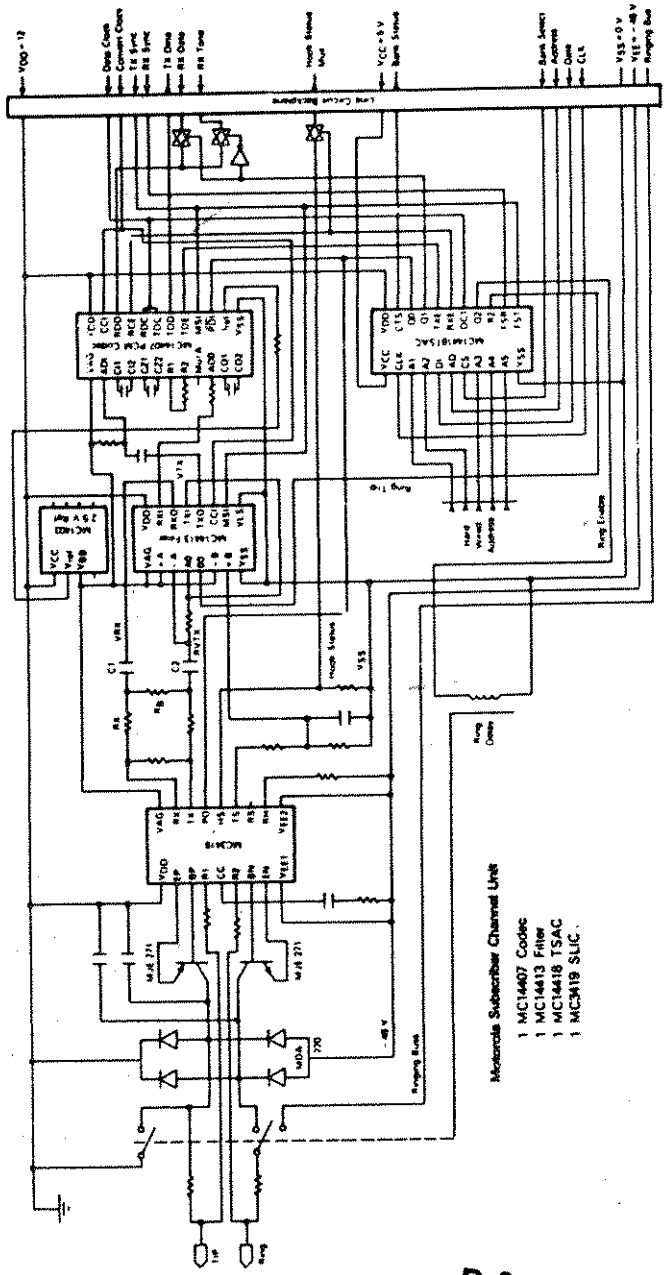
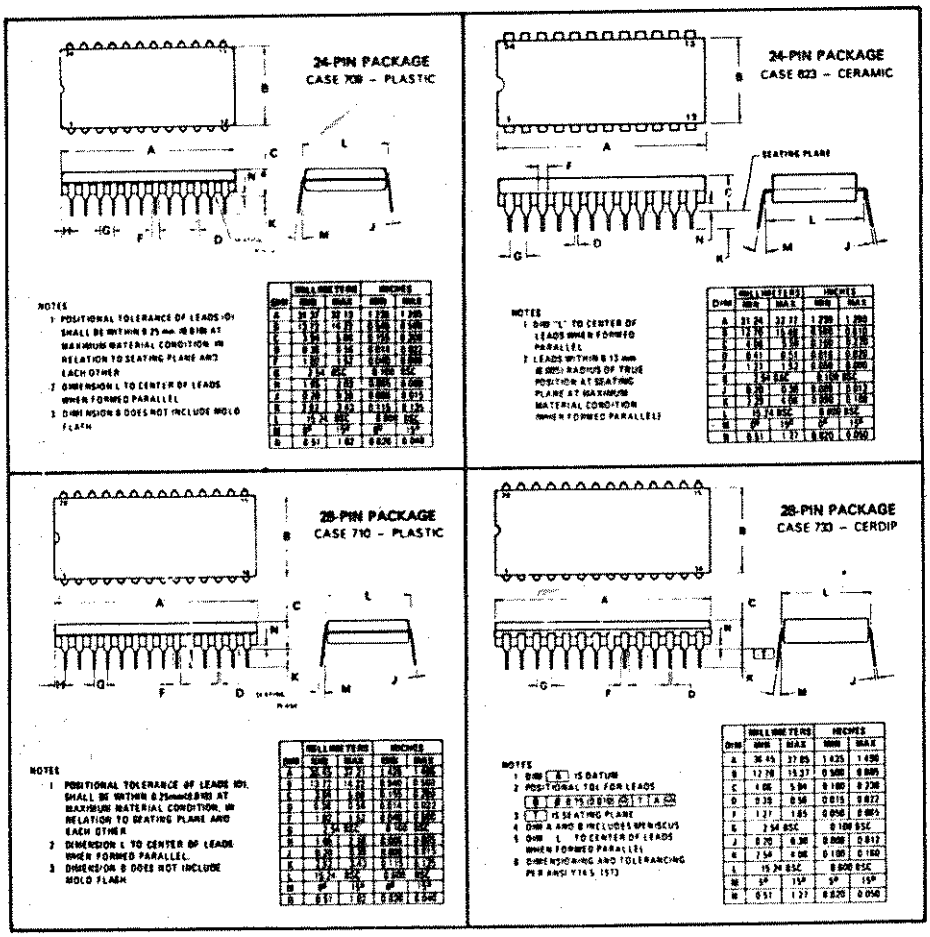

Motorola Subscriber Channel Unit
1 MC14407 Codec
1 MC14413 Filter
1 MC14418 TSAC
1 MC3419 SLIC

FIGURE 12

MC14404 • MC14406 • MC14407

PACKAGE DIMENSIONS



4

D-10

Monolithic Memories

57401
67401
March 1977

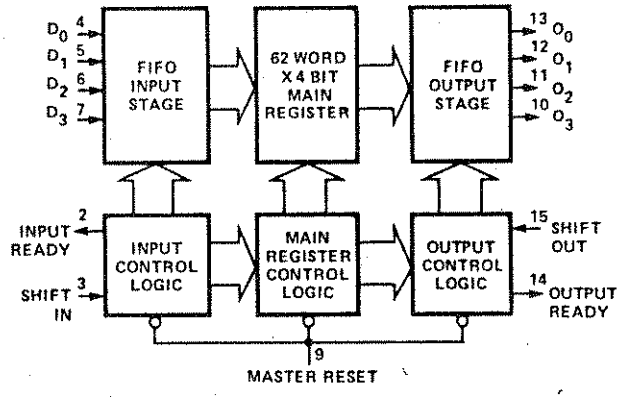
First In, First Out 64 x 4 Serial Memory - FIFO

PRODUCT FEATURES

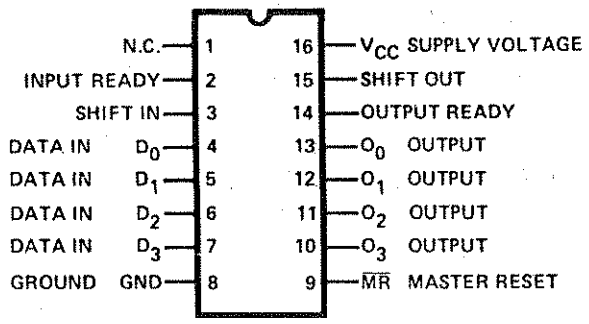
- 15 MHz Shift In, Shift Out Typical Rates
- Advanced Schottky Bipolar Processing
- TTL Inputs and Outputs
- Readily Expandable in Word and Bit Dimensions
- Asynchronous or Synchronous Operation
- Pin Compatible with Fairchild's F3341 MOS FIFO and Ten Times as Fast

PART NUMBER	PACKAGE	TEMPERATURE RANGE
57401	J16	Military
67401	J16	Commercial

LOGIC DIAGRAM

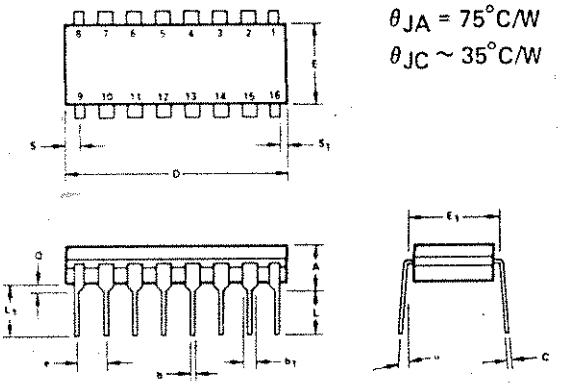


PIN CONFIGURATION



PACKAGE DRAWING J16

θ_{JA} = 75°C/W
θ_{JC} ~ 35°C/W



SYMBOL	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	–	.190	–	4.83
b	.016	.020	.41	.51
b_1	.055	.065	1.40	1.65
C	.008	.012	.20	.31
D	.755	.790	19.18	20.07
E	.265	.300	6.73	7.62
E_1	.290	.320	7.37	8.13
e	.090	.110	2.29	2.79
L	.125	.165	3.18	5.08
L_1	.150	–	3.81	–
Q	.015	.035	.38	.89
S	–	.070	–	1.78
S_1	.000	–	0.00	–
α	0°	15°	0°	15°

Monolithic Memories

1165 East Arques Avenue, Sunnyvale, CA 94086
Tel: (408) 739-3535 / TWX: 910-339-9229

ABSOLUTE MAXIMUM RATINGS

Supply Voltage, V_{CC} . . . . . -0.5V to +7.0V
Input Voltage . . . . . -1.5V to +5.5V
Input Current . . . . . -20 mA to 5 mA
Output Current . . . . . -100 mA to 100 mA
Storage Temperature Range . . . . . -65°C to +150°C

RECOMMENDED OPERATING CONDITIONS

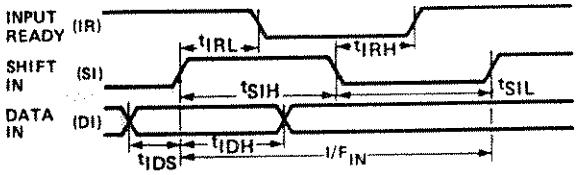
SYMBOL	PARAMETER	MILITARY			COMMERCIAL			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}	Supply Voltage	4.5	5	5.5	4.75	5	5.25	V
t_A	Operating Free Air Temperature	-55		125	0		70	°C

AC CHARACTERISTICS with Standard Load

SYMBOL		FIGURE	57401 – MILITARY 5 ±10%, -55°C to +125°C			67401 – COMMERCIAL 5 ±5%, 0°C to 70°C		
			MIN	MAX	UNITS	MIN	MAX	UNITS
f_{IN}	Shift In Clock Rate	1	7		MHz	10		MHz
t_{SIH}	Shift in High Time	1	45		ns	35		ns
t_{SIL}	Shift In Low Time	1	45		ns	35		ns
t_{IRL}	Input Ready ON Delay	1		60	ns		45	ns
t_{IRH}	Input Ready OFF Delay	1		60	ns		45	ns
t_{IDS}	Input Data Set Up	1	10		ns	5		ns
t_{IDH}	Input Data Hold Time	1	55		ns	45		ns
f_{OUT}	Shift Out Clock Rate	2	7		MHz	10		MHz
t_{SOH}	Shift Out High Time	2	45		ns	35		ns
t_{SOL}	Shift Out Low Time	2	45		ns	35		ns
t_{ORL}	Output Ready ON Delay	2		65	ns		55	ns
t_{ORH}	Output Ready OFF Delay	2		65	ns		55	ns
t_{ODH}	Output Data Hold Time	2	10		ns	20		ns
t_{ODS}	Output Data Delay	2		65	ns		55	ns
t_{PT}	Data Throughput Time	Note 1		4.0	μs		3.0	μs
t_{MRW}	Master Reset Pulse	Note 2	30		ns	25		ns
t_{MRORL}	Master Reset to OR Low	5		65	ns		55	ns
t_{MRIRH}	Master Reset to IR High	5		45	ns		35	ns
t_{MRS}	Master Reset to SI	5	45		ns	35		ns
t_{IPH}	Input Ready Pulse High	4	45		ns	35		ns
t_{IPL}	Input Ready Pulse Low	6	45		ns	35		ns
t_{OPH}	Output Ready Pulse High	3	45		ns	35		ns
t_{OPL}	Output Ready Pulse Low	7	45		ns	35		ns

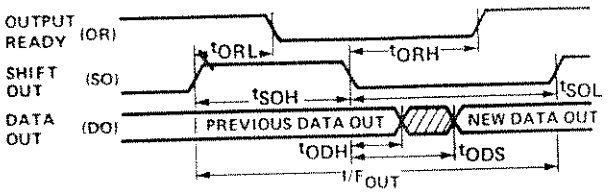
NOTES: 1. This parameter defines total time from the time data is present at D_0–D_3 to the time it is available at O_0–O_3 with FIFO initially empty and total time from the time data is extracted from O_0–O_3 to IR High with FIFO initially full.
2. Master Reset clears the 57401/67401 to the all cells empty state.

TIMING DIAGRAMS



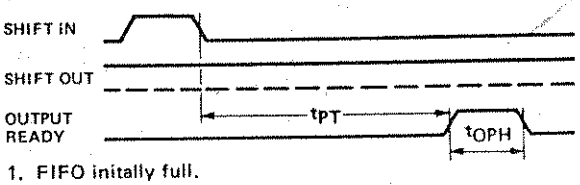
1. Input data must remain stable during t_{IDS} and t_{IDH}.
2. Input Ready HIGH indicates that space is available and a Shift In pulse may be applied. Input Ready LOW indicates that the FIFO is full or that a previous Shift In operation is not complete. Shift In pulses applied while Input Ready is LOW will be ignored.
3. The rise of Input Ready indicates that the data at the D0–D3 inputs has been accepted and that the input stage is empty.

Figure 1. FIFO Input Timing



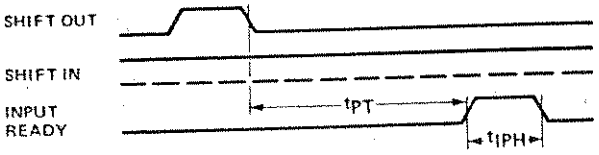
1. Output data will be stable at the rise of Output Ready.
2. Output Ready HIGH indicates that data is available and a Shift Out pulse may be applied. Output Ready LOW indicates that the FIFO is empty or that a previous Shift Out operation is still in progress. Shift Out pulses applied while Shift Out Ready is LOW will be ignored.
3. The rise of Output Ready indicates that new data has been loaded into the output stage.

Figure 2. FIFO Output Timing



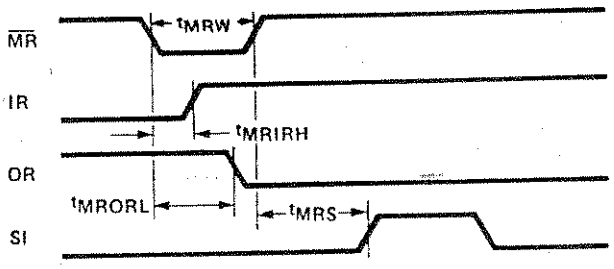
1. FIFO initally full.
2. Shift In held high (as shown).

Figure 3. t'_{OPH} Specification



1. FIFO initially empty.
2. Shift Out held high (as shown).

Figure 4. t_{IPH} Specification



1. FIFO initially full

Figure 5. FIFO Master Reset Timing

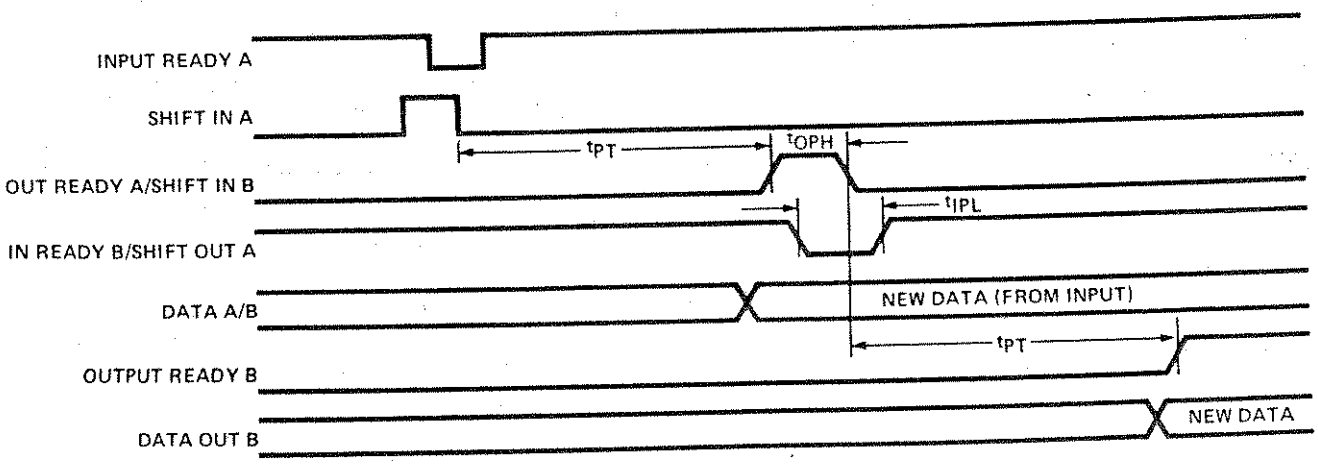


Figure 6. FIFO/FIFO Communication: Empty/Input Timing

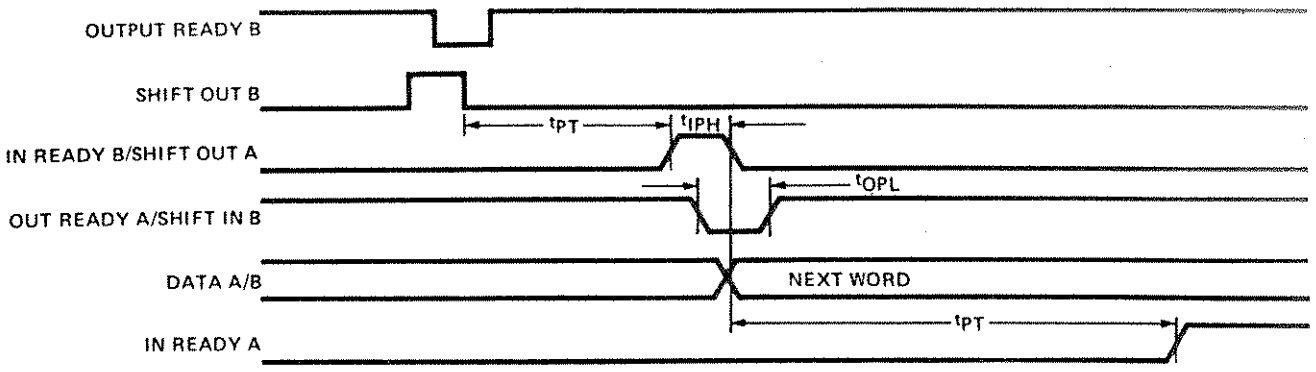
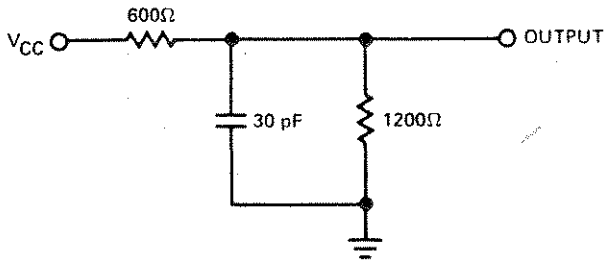


Figure 7. FIFO/FIFO Communication: Full/Output Timing

STANDARD TEST CIRCUIT



Input Pulse Amplitude = 2.5V
Input Rise and Fall Time = 5.0 ns from 1.0V to 2.0V
Measurements made at 1.50V

DC ELECTRICAL CHARACTERISTICS

OVER RECOMMENDED OPERATING FREE AIR TEMPERATURE RANGE UNLESS OTHERWISE NOTED

SYMBOL	PARAMETER	CONDITIONS	MIN	MAX	UNIT
V_{IL}	Low Level Input Voltage			0.8	V
V_{IH}	High Level Input Voltage		2		V
V_{IC}	Input Clamp Voltage	V_{CC} = Min, I_I = −18 mA		−1.5	V
I_{IL1}	Low Level Input Current Inputs D_0–D_3, MR	V_{CC} = Max, V_F = 0.45V		−0.8	mA
I_{IL2}	Low Level Input Current Inputs SI, SO	V_{CC} = Max, V_F = 0.45V		−1.6	mA
I_{IH}	High Level Input Current	V_{CC} = Max, V_I = 2.4V		50	μA
I_I	Maximum Input Current	V_{CC} = Max, V_I = 5.5		1	mA
I_{CC}	Supply Current	V_{CC} = Max, See Note 1		150	mA
V_{OL}	Low Level Output Voltage	V_{CC} = Min, I_{OL} = 8 mA		0.5	V
I_{OS}	Output Short Circuit Current	V_{CC} = Max, V_O = 0	−20	−90	mA
V_{OH}	High Level Output Voltage	V_{CC} = Min, I_{OH} = 0.9 mA	2.4		V

NOTE: 1. I_{CC} measured with worst case sequence; FIFO reset, one shift-in cycle with data inputs held low, measure I_{CC} with all inputs held low.

FUNCTIONAL DESCRIPTION

Data Input

Data is entered into the FIFO on D_0–D_3 inputs. To enter data the Input Ready (IR) output should be HIGH, indicating that the first location is ready to accept data. Data then present at the four data inputs is entered into the first location when the Shift In (SI) is brought HIGH. A SI HIGH signal causes the RI to go LOW. Data remains at the first location until SI is brought LOW. When SI is brought LOW and the FIFO is not full, IR will go HIGH, indicating that more room is available. Simultaneously, data will propagate to the second location and continue shifting until it reaches the output stage or a full location. If the memory is full, IR will remain LOW.

Data Transfer

Once data is entered into the second cell, the transfer of any full cell to the adjacent (downstream) empty cell is automatic, activated by an on-chip control. Thus data will stack up at the end of the device while empty locations will "bubble" to the front. t_{PT} defines the time required for the first data to travel from input to the output of a previously empty device.

Data Output

Data is read from the O_0–O_3 outputs. When data is shifted to the output stage, Output Ready (OR) goes HIGH, indicating the presence of valid data. When the OR is HIGH, data may be shifted out by bringing the Shift Out (SO) HIGH. A HIGH signal at SO causes the OR to go LOW. Valid data is maintained while the SO is HIGH. When SO is brought LOW the upstream data, provided that stage has valid data, is shifted to the output stage. When new valid data is shifted to the output stage, OR goes HIGH. If the memory is emptied, OR stays LOW.

Input Ready and Output Ready may also be used as status signals indicating that the FIFO is completely full (Input Ready stays LOW for at least t_{PT}) or completely empty (Output Ready stays LOW for at least t_{PT}).

Expansion

FIFOs can be connected serially to form larger effective FIFOs: i.e., two 64 X 4 FIFOs can be connected to form a 128 X 4 FIFO, as shown in Figure 8. This is done by connecting the data outputs of the input side FIFO, A, to the data inputs of the output side FIFO, B, and Output Ready and Shift Out Pins of the input FIFO, A, to the Shift In and Input Ready pins of the output FIFO, B, respectively.

If both FIFOs are empty, the Input Ready/Shift Out pins between the FIFOs will be HIGH and the Output Ready/Shift In pins between the FIFOs will be LOW. If the Shift In pin of FIFO A is clocked, the data will propagate to the output. This will set the output full indicator, causing the Output Ready/Shift In combination to rise temporarily. However, since the Input Ready/Shift Out combination is HIGH this enables the Shift Out clock and the full indicator is reset immediately after the rise of Output Ready/Shift In. The result in a positive going pulse on the Output Ready/Shift In line which is t_{OPH}, Output Pulse High (empty), as shown in Figure 6. The FIFO chip is designed such that this pulse is greater than the minimum Shift In High time, t_{SIH}. The leading edge (LOW to HIGH) of this Output Ready pulse indicates that the new word is available at FIFO A's output. This LOW to HIGH transition on FIFO B's Shift In causes FIFO B to load the data from FIFO A into B's first (input) word. FIFO B responds by taking the Input Ready/Shift Out combination LOW. The trailing edge, HIGH to LOW transition of Output Ready/Shift In Pulse allows the word in FIFO B to propagate on to its output. Once propagation has begun, FIFO B will again raise Input Pulse Low (empty). The FIFO is designed such that this pulse is greater than the minimum Shift Out Low time, t_{SOL}.

Chip-to-chip communication for the case of two connected FIFOs, both full, with an empty location bubbling from the output to the input is similar to the both-empty case described above. When both FIFO A and B are full, Input Ready/Shift Out is LOW and Output Ready/Shift In is HIGH. When an empty location bubbles from the output of FIFO B to its input, Input Ready/Shift will temporarily go HIGH. Since Output Ready/Shift In is already HIGH, the Shift In clock is enabled and Input Ready/Shift Out will immediately go LOW. The result is a pulse on the Input Ready/Shift Out line which is t_{IPH}, Input Pulse High (full). The FIFO is designed such that this pulse is greater than the minimum Shift Out High time, t_{SOH}. The leading edge of this pulse will cause FIFO A to load new data at its outputs, and the trailing edge will cause the newly created empty location to bubble through FIFO A. The rise of the Input Ready/Shift Out pulse will cause Output Ready/Shift In to go LOW. Output Ready/Shift In will go HIGH again when new data has been loaded into FIFO A's output. This creates a negative pulse on the Output Ready/Shift In line which is t_{OPL}, Output Pulse Low (full). The FIFO is designed such that t_{OPL} is greater than the minimum Shift Out Low time.

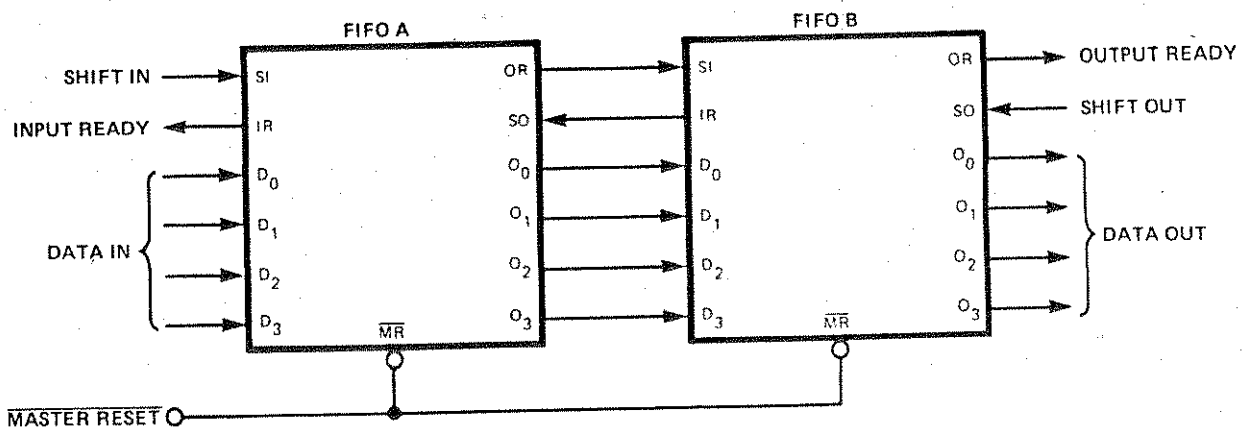


Figure 8. 128 X 4 FIFO

Word Length Expansion

Expansion of the FIFO to a longer word length is accomplished as shown in Figures 9 and 10. Careful attention must be paid to the generation of composite Input Ready and Output Ready signals. Because of device-to-device variation the Shift In (SI) to Input Ready (IR) delays will be different on the Input FIFOs. If a simple scheme such as shown in Figure 9 is used, then this variation must be allowed for, since composite Input Ready will go LOW as soon as the fastest IR goes LOW, but Shift In cannot go LOW until all three IRs for the input FIFOs are LOW. The user must provide sufficient delay to insure this.

Figure 9 shows one way to eliminate any potential timing problems. The gates and flip-flops are used to generate a Shift In or Shift Out pulse independent of the Shift In or Shift Out signal. The gating insures that Shift In does not go HIGH until all IRs are HIGH and does not go LOW until they are all LOW. The output gating operates in the same way. The only restriction on Shift In and Shift Out are that they must be long enough to clock the flip-flop and that they must occur after the composite Input Ready or Output Ready signal.

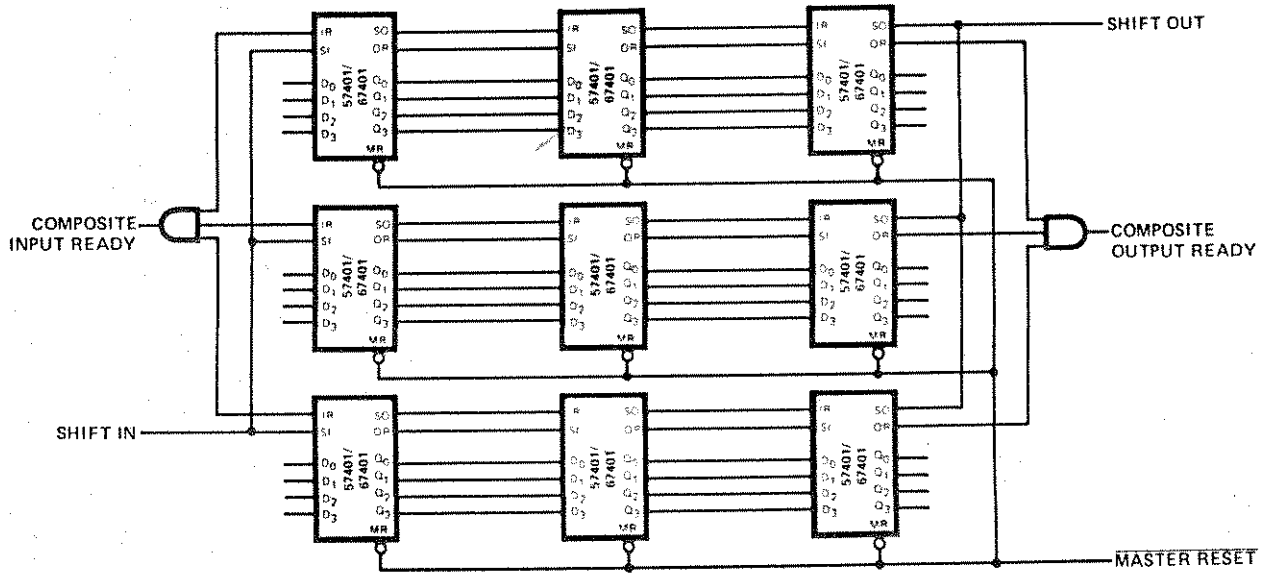


Figure 9. 192 X 12 FIFO — Unlatched Control Lines

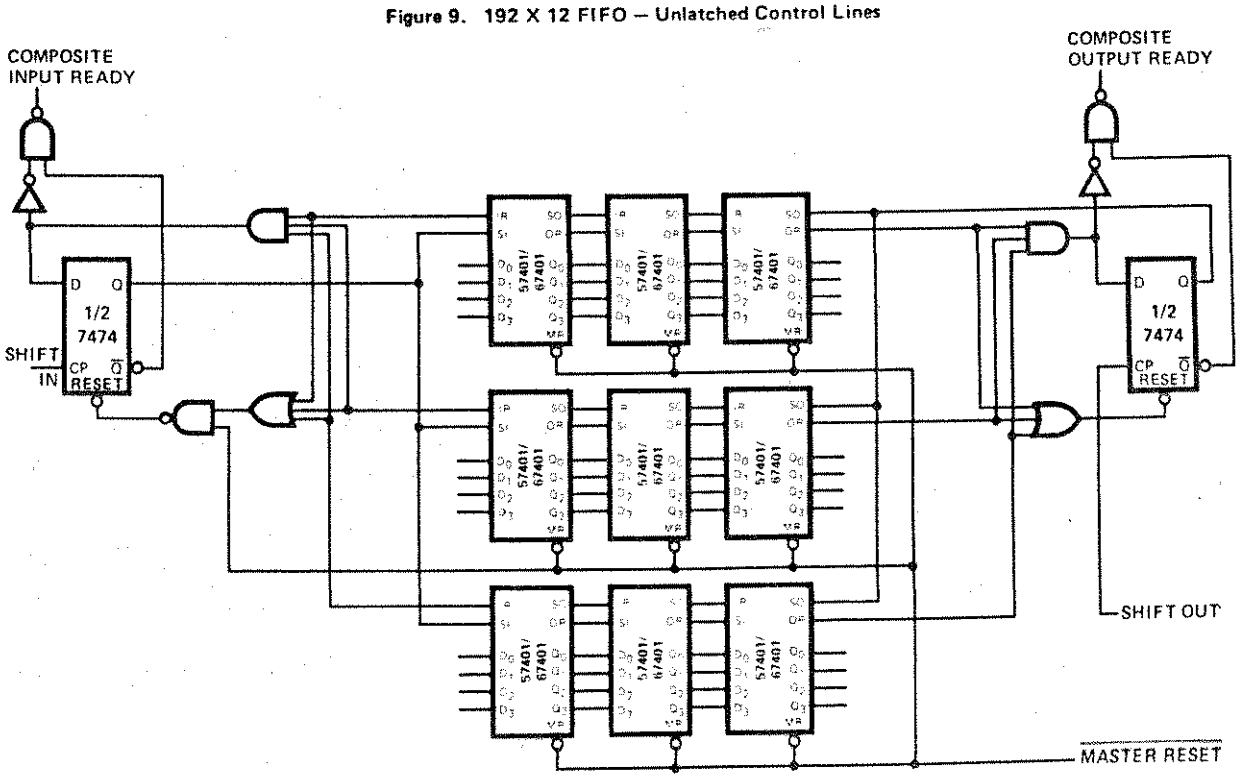


Figure 10. 192 X 12 FIFO — Latched Control Lines