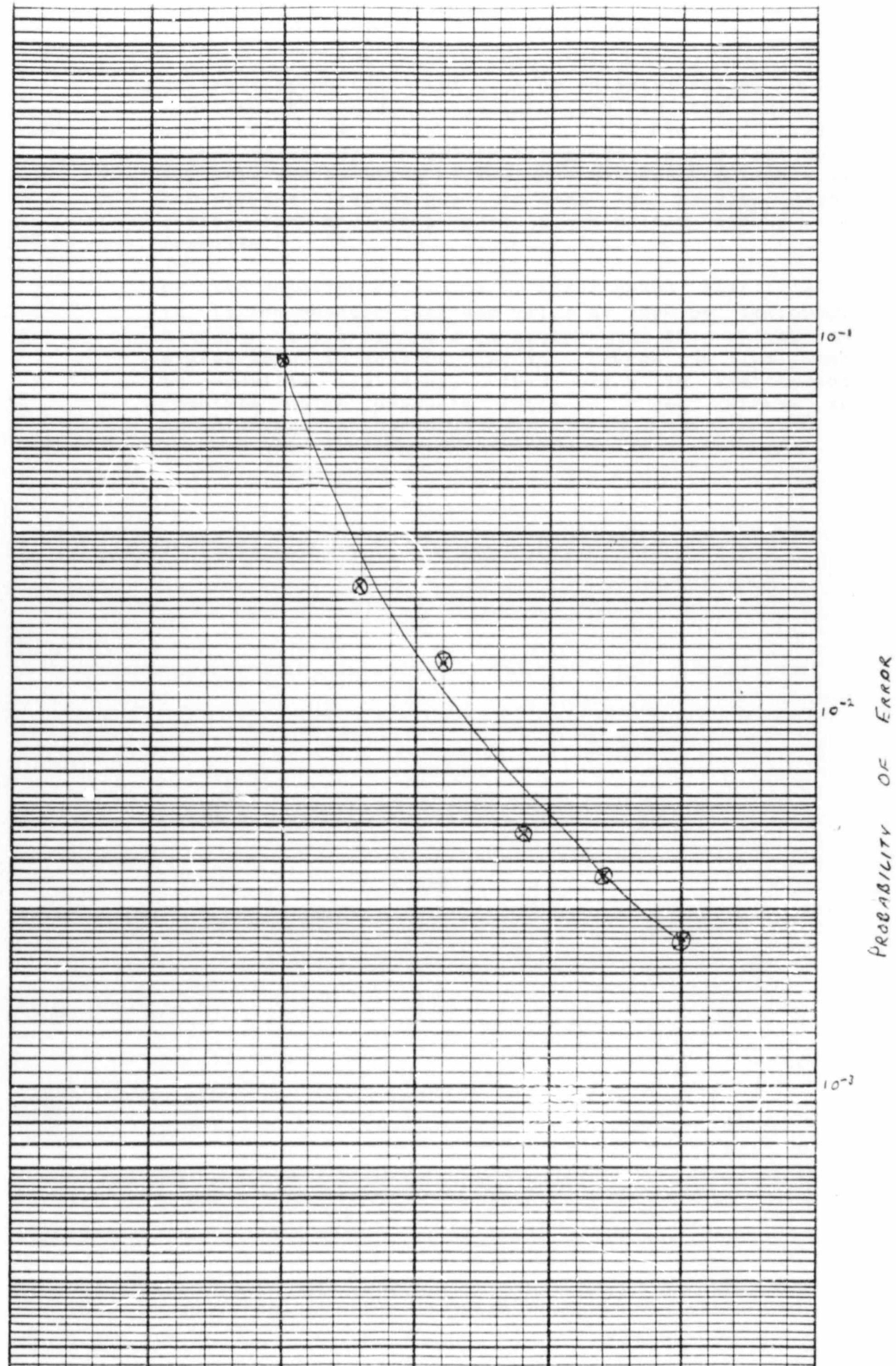


Fig 7.3 Plot of Bit Errors vs S/N for a = 1000

during testing that even with no injected noise, while running the modem unsynchronised to the data as explained above, using the circuit of Fig 7.1, random errors appeared. These are attributed to the fact that the data and sampling clocks were not synchronised. As above, the indications are that the results might differ for the case where synchronised clocks are feasible.

On the question of choice of convergence coefficient, it is recommended that the value of 1000 be employed. Although the error rate is higher with this value, as stated above in Section 7.2, it reduces the likelihood of equaliser latch-up. This value is the lowest possible which prevents latch-up, yet returns an acceptable error performance.

7.4 The Effect of Group Delay

This is an important area in terms of the intended application of the modem to long distance, multilink circuits. Although each stage of a link is pre- or post-equalised by means of fixed passive equalisers, it is not in general possible accurately to equalise any given circuit. Thus a channel has a built in error producing component in addition to noise, usually low, less than -30 dBm under worst case conditions except faults, measured wideband. Additionally, when maintenance is carried out on lines, characteristics may change, resulting in an unpredictable error performance. It is therefore highly desirable that the modem described here is able, through its adaptive equalisation capability, to adjust its operation to suit these time-varying channels.

Tests were carried out on the modem for tolerance to group delay. These were performed without any additional noise to avoid clouding the results. The circuit used for these tests is shown in Fig 7.4. The

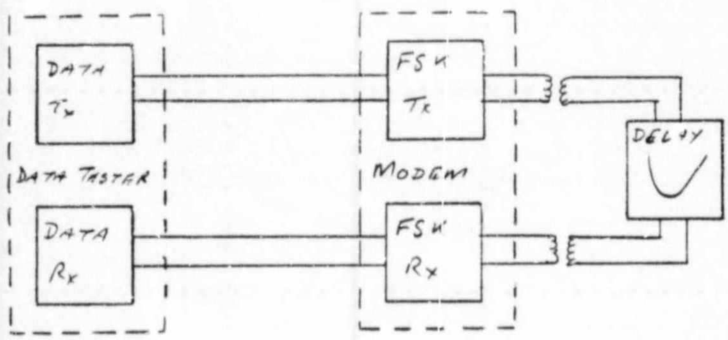


Fig 7.4 Connections for Group Delay Tests

channel was simulated by an equaliser set up to simulate a channel with an amplitude response as flat as possible, but with a group delay representative of a real channel. The equaliser used is a Convex C30 with independently variable delay and amplitude characteristics over a frequency range 600 Hz to 3000 Hz. Figures 7.5 and 7.6 show the amplitude and group delay responses of the channel as used. Note the delay between the mark and space frequencies. A more exaggerated response was tried, at which the modem also ran, but to obtain results from which comparisons could be drawn between different convergence factors, it was felt preferable to allow the modem to operate slightly below the extreme at which almost continuous errors occurred.

It is appreciated that the channel shown is only one example of many possible channel shapes, but this shape was considered sufficiently representative to determine whether an adaptive equaliser provides any benefit in

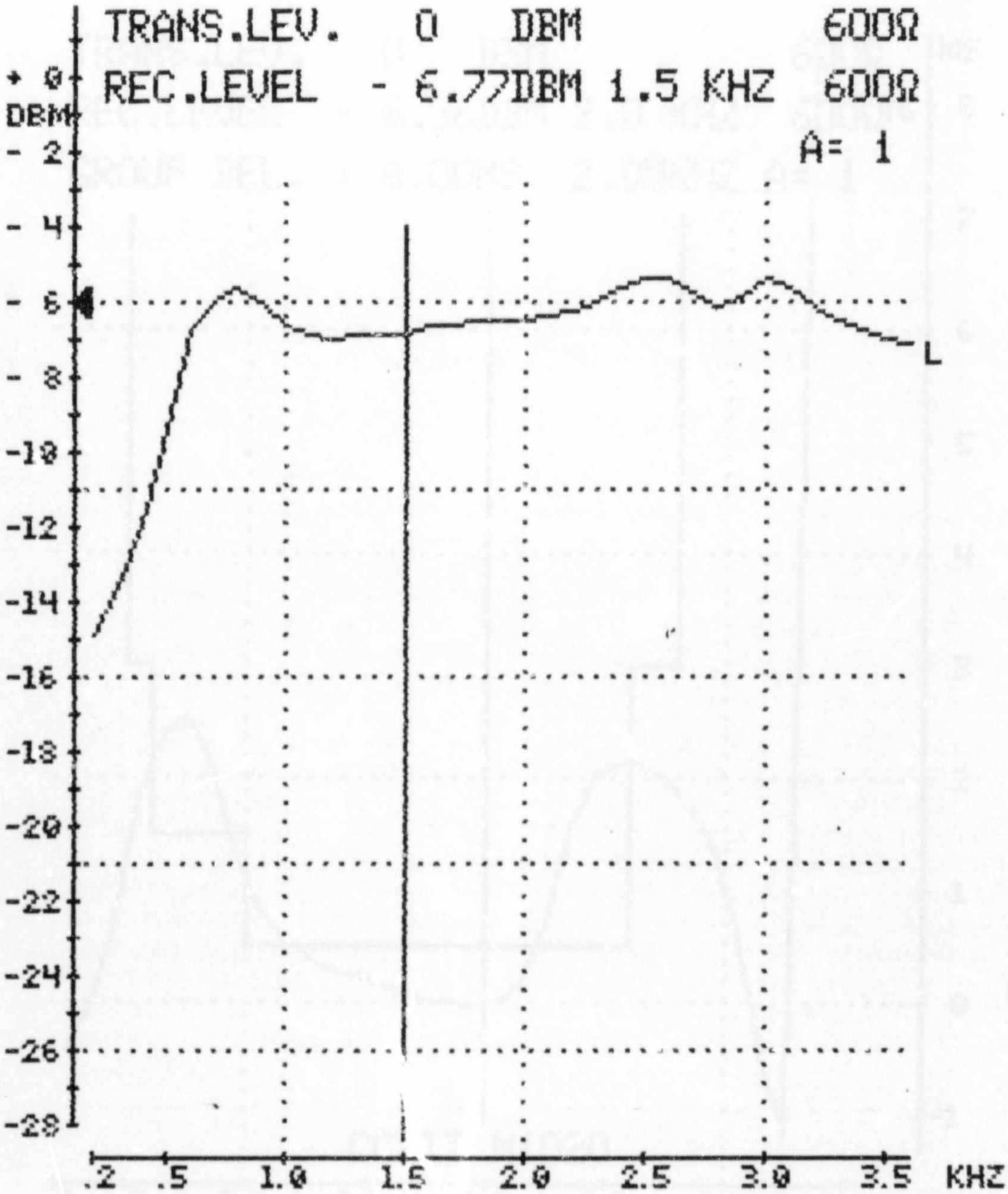


Fig 7.5 Amplitude Response of "channel" as simulated compensating for delay distortion.

Five convergence factors were tested, ranging from 0 (no adaptation) through to 4000. At each, ten tests were run using 1[illegible] bits and the bit errors noted, and then a single run with 10[illegible] bits. The results show that errors they tend to occur in bursts. This is because before the data test set loses synchronisation, there is usually a run of individual errors, but as synchronisation is lost, a burst of errors is registered before the counter is blocked. Counter unblocking occurs again only when the tester resynchronises. This causes large differences between data runs, but these average out overall and in the longer test are not so apparent.

The results are presented in Tables 7.2 and 7.3. Note in Table 7.3 the total errors without adaptation. It is

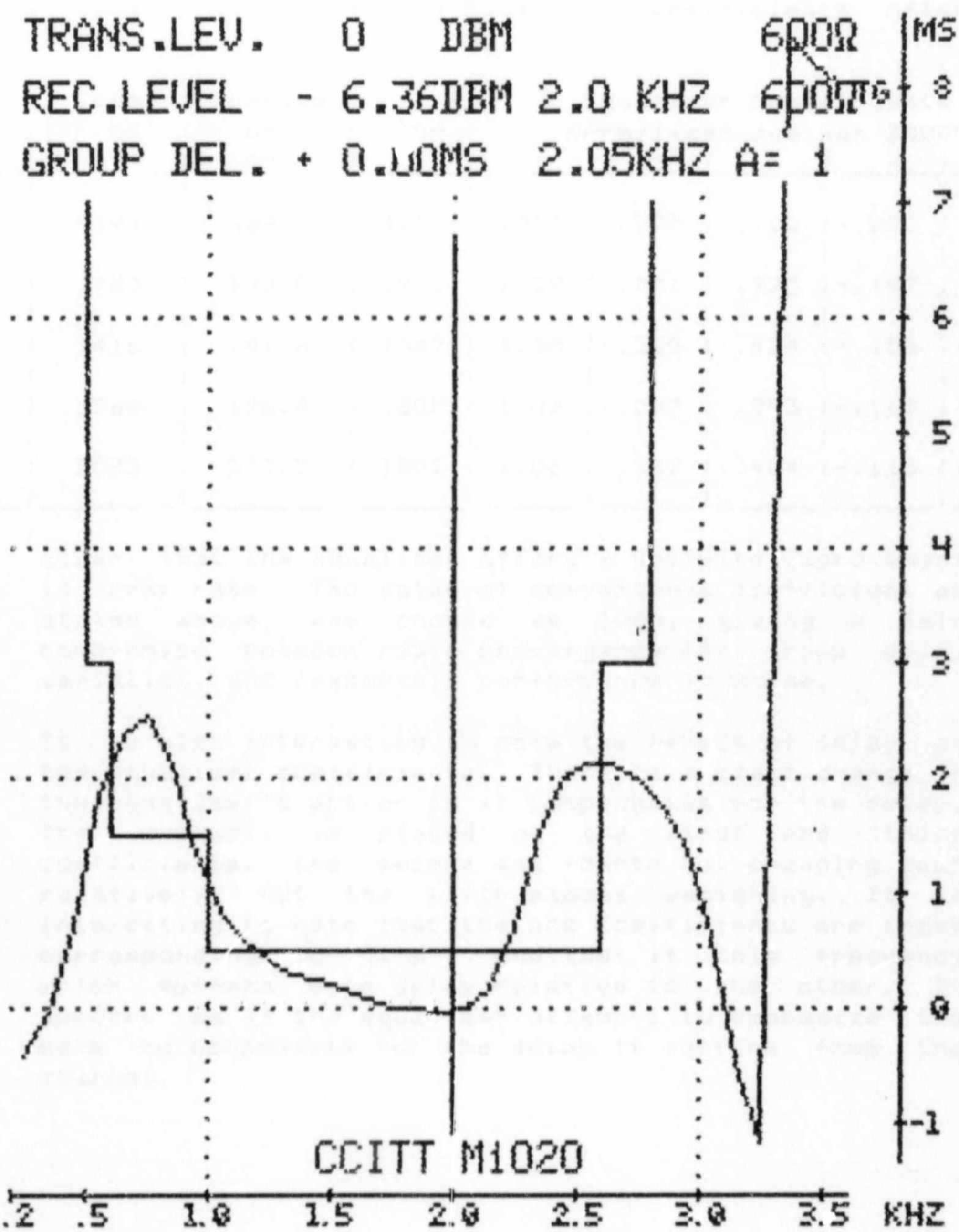


Fig 7.6 Group Delay Response of "channel" as simulated

Table 7.2 Error Rate for Simulated Channel

Test No. / a	1	2	3	4	5	6	7	8	9	10
0	549	559	328	479	427	774	404	365	334	473
500	168	119	254	238	179	194	217	111	193	247
1000	91	189	171	229	97	84	226	56	88	185
2000	155	59	221	187	183	125	156	257	182	239
4000	307	84	176	259	132	191	205	144	311	224

Table 7.3 Summary of Results and Equaliser Coefficients After Adaptation

Test a	Total of 10^4	Average of 10^4	10^8	Equaliser Coefficients Normalised against 30000 1	2	3	4	5
0	4692	469.2	4313	.050	-.200	1.00	-.200	.050
500	1920	192.0	1911	1.09	-.221	.973	-.147	.008
1000	1416	141.6	1867	1.08	-.220	.984	-.158	-.013
2000	1764	176.4	1807	1.07	-.237	.953	-.163	-.015
4000	2033	203.3	1802	1.06	-.222	.904	-.163	-.040

clear that the equaliser offers a definite improvement in error rate. The value of convergence coefficient as stated above, was chosen as 1000, giving a fair compromise between rapid convergence for group delay variation, and reasonable performance in noise.

It is also interesting to note the effect of delay on the equaliser coefficients. There is a clear change in the equaliser's action as it compensates for the delay. The emphasis is placed on the first and third coefficients, the second and fourth not changing much relatively, but the fifth almost vanishing. It is interesting to note that the odd coefficients are those corresponding to "1's", and that it this frequency which suffers more delay relative to the other. It appears as if the equaliser attempts to emphasize the mark to compensate for the delay it suffers from the channel.

8. CONCLUSION

8.1 Results

The requirement was to produce an FSK modem to operate on non-standard frequencies, at 1200 bd, over various multilink channels. An implicit requirement is that the device shall have a low bit error rate in the presence of possibly large levels of noise, or over a channel with poor delay characteristics, or a combination of both.

The design presented here produces results which will meet the requirements in most cases, the obvious area for attention being the noise performance. From the results obtained in Chapter 7, it can be seen that an improvement of say 3 dB in the noise performance of the system may be desirable, but not a necessity.

The modem returns very satisfactory results under conditions of severe group delay, already indicating a useful achievement in this area. This has been ascribed to the excellent delay characteristics of transversal filters, employed throughout the system, and only achievable in the flexible forms presented here by using digital processing.

8.2 Suggestions for Further Work

Several areas lend themselves to further concentrated development. One is the receiving or channel filter. It has been stated that the optimum filter is a cosine shape, but as has been shown, there is an improvement to be gained by using a filter with a slightly different shape. It may be possible to further improve the error rate by investigating the filter to find the optimum shape.

The post detection filter may be another area where a different approach may reap benefits, likewise the equaliser. The approach taken here, using 5 coefficients and 33 delays was adopted to save processing time and memory. It may be beneficial, however, to examine the possibility of spacing the coefficients by 4 samples instead of 8 ie there would be 9 coefficients and 33 delays. This may give improved rejection of intermediate interference components.

A method of reducing the effects of jitter may offer an interesting area of study, as this is an inherent problem with all sampled data systems. An improvement here would certainly be visible in the error rate of the modem, but probably not in the tolerance to delays.

It is clear therefore, that the coverage presented here is limited to the basic concept, and the proof of its viability, and that many avenues remain open for detailed further examination on this topic.

8.3 Work Done

This paper has taken a close look at a very specific application of signal processing, digital signal processing to be specific, viz the design of a modem. Modem design offers some interesting challenges to the designer in that the characteristics of the channel play a large part in determining whether or not a design will perform as expected, or better or worse. Many aspects are unpredictable without large quantities of mathematics which can only attempt to predict the end result.

Previous work on the subject of data transmission has been considered as the logical basis for the discussion presented here. This is developed upon, taking into account the peculiarities of the system requirements.

Each aspect of the design of an FSK modem has been considered in detail from the transmitter through the receiving filter, FM detector, post detection filter and adaptive equaliser, presenting the problems encountered in their development and their solution. The result is a modem which compares very favourably with the existing modem, especially in tolerance to group delay.

Concluding, then, this area of study offers many interesting possibilities for utilising the flexibility of digital processes in achieving improved performances from devices previously thought to be firmly in the analog domain.

9. REFERENCES

[1] E.D. Sunde, "Theoretical Fundamentals of Pulse Transmission - I", Bell System Technical Journal, Vol 33, pp 721-788, May 1954

[2] William R. Bennett and James R. Davey, "Data Transmission", McGraw-Hill, 1985

[3] Ferrel G. Stremler, "Introduction to Communication Systems", Addison-Wesley, 1977

[4] William D. Stanley, "Digital Signal Processing", Reston, 1975

[5] Arthur B. Williams, "Electronic Filter Design Handbook", McGraw-Hill, 1981

[6] 2920 Analog Signal Processor Design Handbook, Intel Corp., 1980

[7] A.P. Clark, "Advanced Data-Transmission Systems", John Wiley and Sons, 1977

[8] Carlos A. Belfiore and John H. Park Jr,"Decision Feedback Equalization", Proc. IEEE, Vol 67, No.8, August 1979

[9] Rajendra Kumar and John B. Moore, "Adaptive Equalization via Fast Quantized State Methods", IEEE Trans. Commun. Vol COM-29, No.10, pp 1492-1501, October 1981

[10] E.H. Satorius and J.D. Pack, "Application of Least Squares Lattice Algorithms to Adaptive Equalization", IEEE Trans. Commun., Vol COM-29, No.2, pp 136-142, February 1981

[11] D.D. Falconer, V.B. Lawrence and S.K. Tewksbury, "Processor-Hardware Considerations for Adaptive Filter Algorithms", ICC Conf 1981, pp 57.5.1-57.5.6

[12] Bernard Widrow, John McCool, Michael G. Larimore and C. Richard Johnson Jr, "Stationary and Non-Stationary Learning Characteristics of the LMS Adaptive Filter", Proc NATO Advanced Study Institute, Postovenere, La Spezia, Italy, 1976, pp 357-393

[13] TMS 32010 User's Guide, Texas Instruments,1983

APPENDIX A: FILTER DESIGN EXAMPLE

Design a Transversal Bandpass Filter with a 3 dB bandwidth of 1000 Hz with a centre frequency of 1500 Hz. Assume a sampling frequency of 8 kHz. Group delay should be flat.

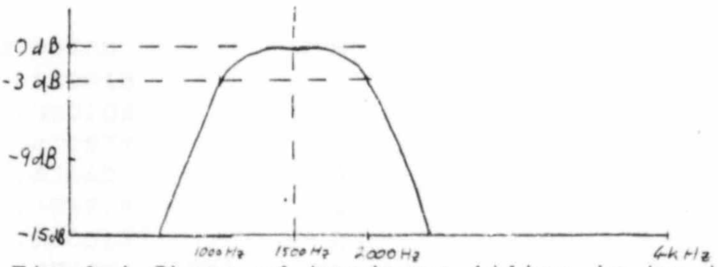


Fig A.1 Shape of bandpass filter to be designed

Start by designing a low-pass filter whose total pass-bandwidth is equal to that of the desired bandpass filter. This should be emphasised as the actual bandwidth of a low-pass filter is twice the cut-off frequency. Since the transformation applied to a low-pass filter to convert it to bandpass is the Fourier Transform frequency shifting property, this will transform the whole band of negative and positive frequencies to the higher frequency. Design therefore a low-pass prototype with a cut-off frequency of 500 Hz = f_L.

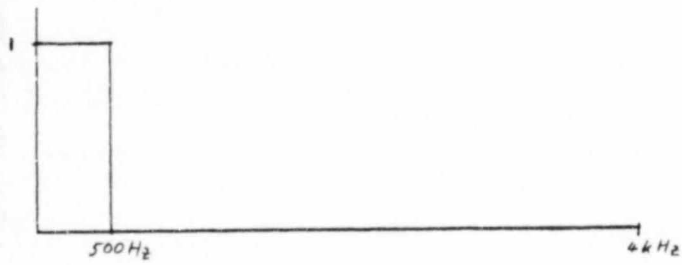


Fig A.2 The shape of the desired low-pass filter

The coefficients of a transversal filter are simply points on the impulse response of the desired filter.

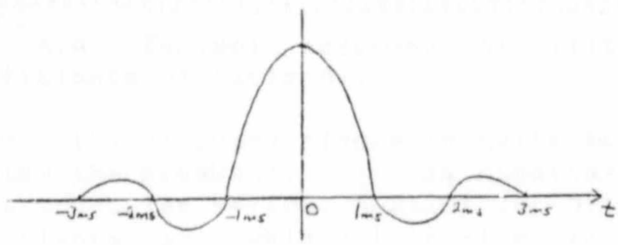


Fig A.3 Impulse response of filter in Fig A.2. Note the zero crossing times

To obtain adequate stopband rejection with freedom from large ripples, at least the first zero-crossing on either side of the origin should be included. This means that 17 coefficients should be allowed, although the outer coefficients will be zero as the sampling frequency is an exact multiple of the cut-off frequency. These coefficients will lie on the curve

$$g(t) = \frac{\sin 2\pi f_L t}{2\pi f_L t} \qquad \text{A.1}$$

where $t = nT_{sample}$ $\qquad n=-8,-7,\ldots 0,\ldots 7,8$ $\qquad$ A.2

$$\therefore a_n = \frac{\sin 2\pi f_L n T_{sample}}{2\pi f_L n T_{sample}} \qquad \text{A.3}$$

The resulting coefficients are:

Table A.1 Coefficients as derived from A.3

-8	0.00000	8
-7	.139215	7
-6	.300106	6
-5	.470529	5
-4	.63662	4
-3	.784214	3
-2	.900317	2
-1	.974496	1
0	1.00000	

The calculated response given by these coefficients is shown below.

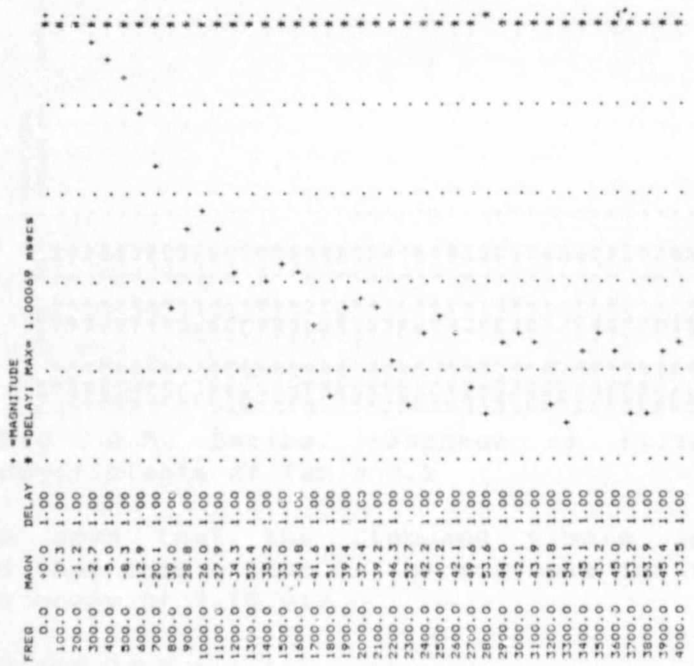


Fig A.4 Decibel response of filter using the coefficients of Table A.1

As can be seen, the stopband ripple is quite severe, averaging about 40 dB below the passband. This is unsatisfactory, and is the result of a truncated series. Application of a Hamming window to the coefficients of Table A.1 results in a much better stopband rejection. The Hamming window is

$$h(t) = 0.54+0.46 \cos \frac{2\pi t}{T} \qquad \text{for } |t| \le \frac{T}{2} \qquad \text{A.4}$$

$$= 0 \text{ elsewhere}$$

where $T = 2\,T_L$

and when the coefficients are scaled according to the Hamming window weightings the following coefficients are obtained

Table A.2 The coefficients scaled by h(t)

-8	0.00000	8
-7	.0160119	7
-6	.0644423	6
-5	.171257	5
-4	.343775	4
-3	.561524	3
-2	.779016	2
-1	.940373	1
0	1.00000	

The response calculated from these coefficients is shown below

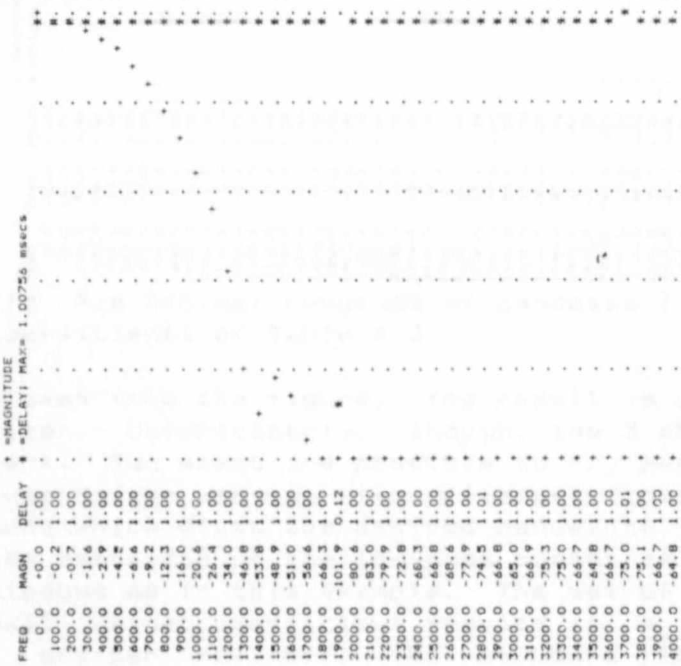


Fig A.5 Decibel response of filter using the coefficients of Table A.2

It can be seen that the stopband ripple is substantially reduced and that this filter is now suitable for transforming to bandpass by means of 4.10 viz

$$a_n = a_n \times 2 \cos 2\pi n \frac{f_c}{f_{sample}} \qquad \text{A.5}$$

$$n=-11,-10,\ldots-1,0,1,\ldots10,11$$

where f_c is the bandpass centre frequency.

The resulting coefficients are shown in Table A.3

Table A.3 The final bandpass coefficients

-8	0.00000	8
-7	-.0122548	7
-6	.0911358	6
-5	.31644	5
-4	-2.96117E-06	4
-3	-1.03756	3
-2	-1.10169	2
-1	.719732	1
0	2	

The response of the bandpass filter can be seen below in Fig A.6

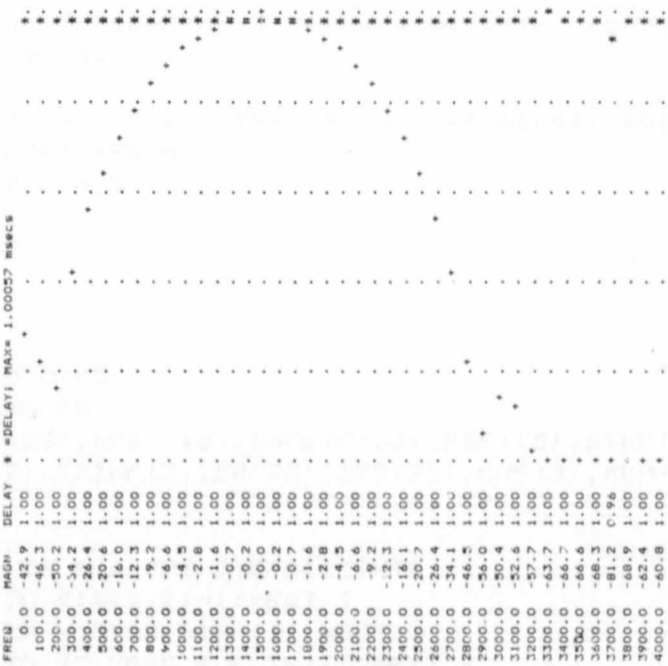


Fig A.6 Decibel response of bandpass filter using the coefficients of Table A.3

As can be seen from the figure, the result is a fairly sharp bandpass filter. Unfortunately, though, the 3 dB points are not quite correct. Two means are possible to try and correct this. One is to use Kaiser windows with different coefficients until one is found which gives the desired bandwidth. Second is to increase the bandwidth of the low-pass prototype filter and use the same windows as in this example. The use of a Kaiser window with a small Kaiser coefficient results in a filter with a potentially sharper roll-off. The stopband ripples are also higher, so there is a trade-off between roll-off and ripple. A better means of ensuring rapid roll-off is to use more coefficients, but this depends on the time constraints placed on the system.

It should be noted that the group delay is flat, except for small inconsistencies which are due either to rounding errors in the display routine, or where the amplitude response is almost undefined due to a very high attenuation.

The Filter Design package routines used in this example were IDLP for the original coefficients of Table A.1, CWIN resulting in Table A.2, and ILPBP to give Table A.3. SETX was used to set the frequency axis up, GDZC to calculate the responses, and DBGDOUT to display the results.

APPENDIX B: THE MODEM SIMULATION ROUTINE

As discussed in Chapter 6, the simulation routine written in BASIC is presented here.

```
10 REM    INITIALISE ALL ARRAYS AND VARIABLES, AND READ IN FILTER
          COEFFICIENTS
20 DEFINT A,B,J-N,S
30 F5=9600
40 PI=3.14159
50 GAIN=.5
60 XA=.01
70 Y0=0
80 Y1=2*PI*1050/F5
90 Y2=2*PI*875/F5
100 DIM XINF(23),XPDF(15),INFLTR(23),PDF(15),DINF(3)
110 DIM SEQ(9),XDINF(3),EQ(33),XEQ(33),CHF(3),XCH(3)

120 REM
130 REM READ IN ALL THE COEFFICIENTS
140 FOR I=1 TO 9:SEQ(I)=1:NEXT I
150 FOR I=1 TO 3:READ DINF(I):NEXT I
160 FOR I=1 TO 23:READ INFLTR(I):NEXT I
170 FOR I=1 TO 15:READ PDF(I):PDF(I)=PDF(I):NEXT I
180 FOR I=1 TO 33:READ EQ(I):NEXT I
190 FOR I=1 TO 3:READ CHF(I):NEXT I

200 REM
210 REM USER ENTRY OF VARIABLES
220 INPUT "NO. OF BITS IN TEST";NBITS
230 INPUT "TYPE OF SEQUENCE 1:1(1),7:1(2),1:7(3),PR(4)";L

240 REM
250 REM PR SEQUENCE GENERATOR - 511 bit PR sequence
260 FOR K=0 TO NBITS
270  IF K MOD 100=0 THEN LPRINT K
     :FOR I=1 TO 33 STEP 8
     :LPRINT I,EQ(I)
     :NEXT I
280  IF L=1 THEN 400
290  IF L=2 OR L=3 THEN 360
300  IF SEQ(9)=SEQ(5) THEN S=0 ELSE S=1
310  FOR J=8 TO 1 STEP -1
320   SEQ(J+1)=SEQ(J)
330  NEXT J
340  SEQ(1)=S
350  GOTO 410
360  S=K MOD 8
370  IF S=>1 THEN S=1
380  IF L=3 THEN IF S=1 THEN S=0 ELSE S=1
390  GOTO 410
400  S=K MOD 2
410  FOR A=1 TO 8

420 REM
430 REM INPUT DATA FILTER
440    DFOUT=0:XDINF(1)=S
450    FOR J=3 TO 1 STEP -1
460     DFOUT=DFOUT+DINF(J)*XDINF(J)
470     IF J>1 THEN XDINF(J)=XDINF(J-1)
480    NEXT J
```

```
490 REM
500 REM FSK TRANSMITTER
510   YO=YO+Y1+Y2*DFOUT
520   FSKIP=SIN(YO)

530 REM
540 REM CHANNEL SIMULATING FILTER
550   FOR J=3 TO 1 STEP -1
560    FSKIP=FSKIP+CHF(J)*XCH(J)
570    IF J>1 THEN XCH(J)=XCH(J-1)
580   NEXT J
590   XCH(1)=FSKIP
600 REM
610 REM NOISE GENERATOR
620   XNOISE=GAIN*2*(RND-.5)
630   FSKIP=FSKIP+XNOISE

640 REM
650 REM INPUT FILTER
660   INFOUT=0:XINF(1)=FSKIP
670   FOR J=23 TO 1 STEP -1
680    INFOUT=INFOUT+INFLTR(J)*XINF(J)
690    IF J>1 THEN XINF(J)=XINF(J-1)
700   NEXT J

710 REM
720 REM LIMITER
730   IF INFOUT=>0 THEN DIFFIN=1 ELSE DIFFIN=0

740 REM
750 REM DIFFERENTIATOR AND ENVELOPE DETECTOR
760   PDFIN=ABS(DIFF2-DIFFIN):DIFF2=DIFFIN

770 REM
780 REM POST DETECTION FILTER
790   PDFOUT=0:XPDF(1)=PDFIN
800   FOR J=15 TO 1 STEP -1
810    PDFOUT=PDFOUT+PDF(J)*XPDF(J)
820    IF J>1 THEN XPDF(J)=XPDF(J-1)
830   NEXT J
840   PDFOUT=PDFOUT/2.50645-1

850 REM
860 REM EQUALISER
870   EQOUT=0:XEQ(1)=PDFOUT
880   FOR J=33 TO 1 STEP -8
890    EQOUT=EQOUT+EQ(J)*XEQ(J)
900   NEXT J

910 REM
920 REM SLICER
930   IF EQOUT>0 THEN SOUT=1 ELSE SOUT=-1
940   CORR=-(EQOUT-SOUT)*XA
950   FOR J=33 TO 1 STEP -8
960    EQ(J)=EQ(J)+XEQ(J)*CORR
970   NEXT J
980   FOR J=33 TO 2 STEP -1
990    XEQ(J)=XEQ(J-1)
1000   NEXT J
1010   PRINT TAB(25);S;(SOUT+1)/2;EQOUT
```

```
1020  NEXT A
1030  PRINT "COUNT,DIN,DOUT:";K;S;SOUT
1040 NEXT K
1050 PRINT CHR$(7)
1060 GOTO 220
1070 END

1080 REM
1090 REM DATA INPUT FILTER COEFFICIENTS
1100 DATA .25,.5,.25

1110 REM
1120 REM INPUT FILTER COEFFICIENTS
1130 DATA -.00516992,.0485119,.119303,-.0158904,-.229235,-.145084,
          .0167147,-.384957,-1.00781,-.558389,1.05243,2,1.05243,
          -.558389,-1.00781,-.384957,.0167147,-.145084,-.229235,
          -.0158904,.119303,.0485119,-.00516992

1140 REM
1150 REM PDF COEFFICIENTS
1160 DATA .0623596,.169765,.320113,.5,.685956,.848827 .960338,1,
          .960338,.848827,.685956,.5,.320113,.169765,.0623596

1170 REM
1180 REM EQUALISER COEFFICIENTS
1190 DATA .172108,0,0,0,0,0,0,0,0,-.990076,0,0,0,0,0,0,0,4.72413,0,
          0,0,0,0,0,0,-.907739,0,0,0,0,0,0,0,.159846

1200 REM
1210 REM CHANNEL SIMULATOR COEFFICIENTS
1220 DATA .718752,-.64733,.310679
```

APPENDIX C: CIRCUIT DIAGRAMS AND PAL STATE TABLES

Two Programmable Array Logic devices are used in the circuit of the modem, one as a data multiplexer, the other as an I/O chip select controller, and for miscellaneous gates. Their state tables are presented here.

Table C.1 State Table for 16L8 PAL Used as Data Multiplexer

There are two select lines: SL1 and SL2, which can take up three different combinations.

SELECT / OUTPUT	Switch Centred SL1=1:SL2=1	Switch Up SL1=1:SL2=0	Switch Down SL1=0:SL2=1
TxD1	$\overline{\text{TxD2}}$	$\overline{\text{TxD3}}$	0
RTS1	$\overline{\text{RTS2}}$	$\overline{\text{RTS3}}$	0
RxD2	$\overline{\text{RxD1}}$	0	TxD3
RLS2	$\overline{\text{RLS1}}$	0	RTS3
RxD3	0	$\overline{\text{RxD1}}$	TxD2
RLS3	1	$\overline{\text{RLS1}}$	RTS2

Table C.2 State Table for 20L10 PAL Used as I/O Selector

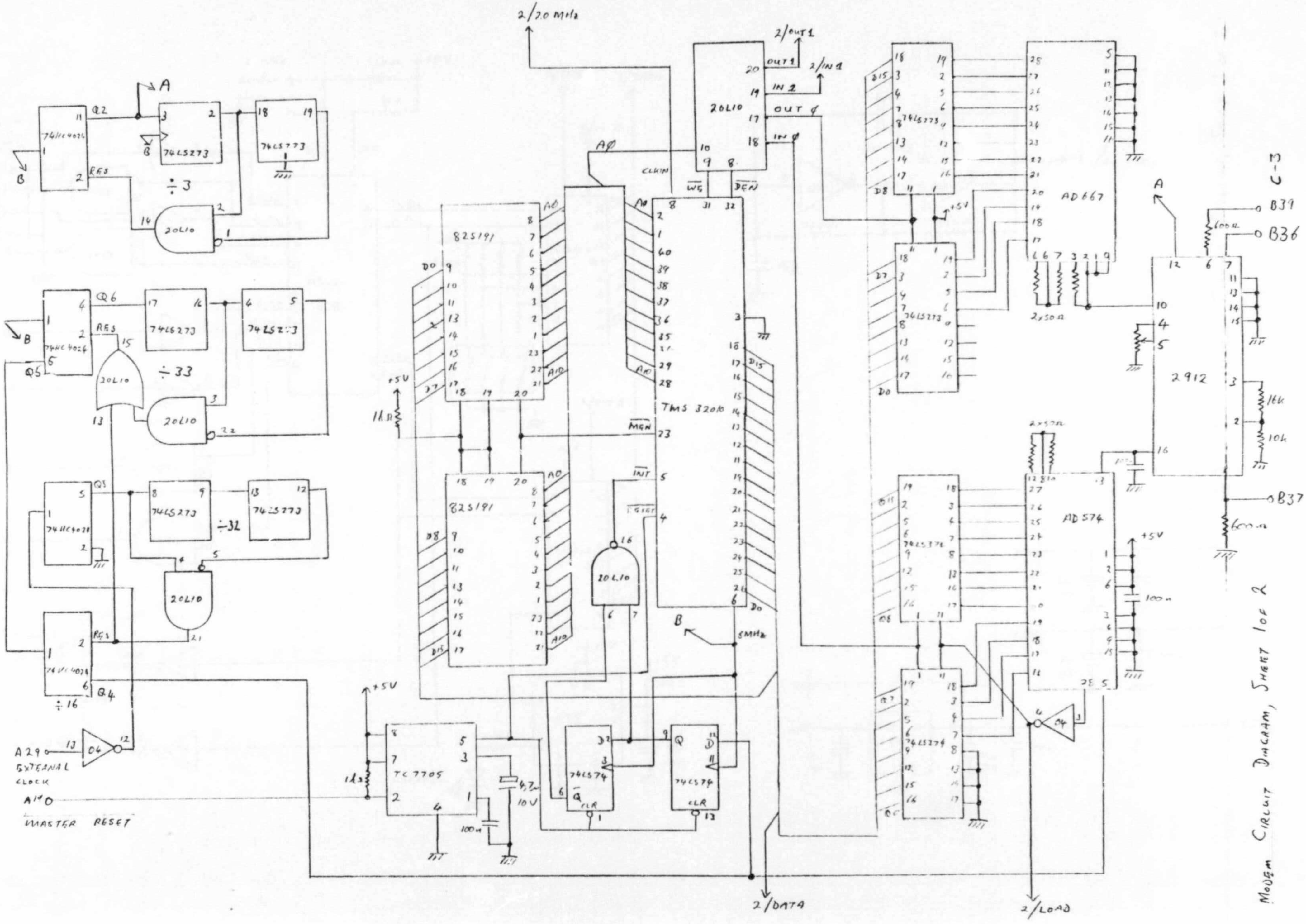
Only the I/O section is defined below. Unless otherwise indicated, the outputs are always "1".

INPUTS			OUTPUTS			
A0	$\overline{\text{WE}}$	$\overline{\text{DEN}}$	OUT0	IN0	OUT1	IN1
0	0	1	0			
0	1	0		0		
1	0	1			0	
1	1	0				0

Table C.3 Power Supply Connections of Major Devices

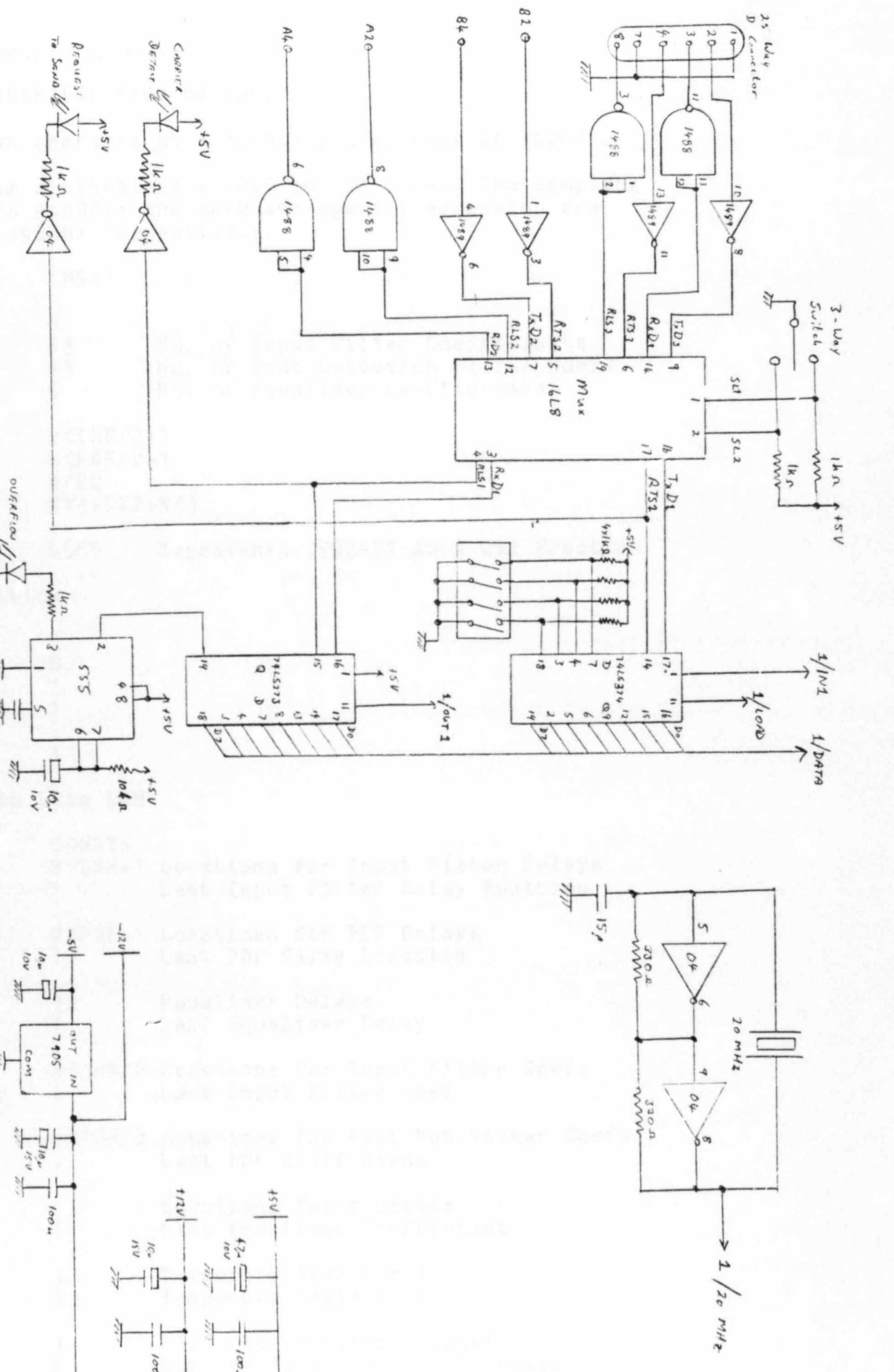
The power connections fo. these devices are not indicated on the drawing for clarity. All power connections should be decoupled to ground _|_ by a 100nF capacitor per device, per supply. Where two
///
grounds are indicated, the first is analog ground, the second digital ground. These are connected together in circuit.

DEVICE	Gnd	+5V	-5V	+12V	-12V
AD 574	9,15	1	-	7	11
AD 667	5,16	-	-	8	10
TMS 320	10	30	-	-	-
1488	7	-	-	14	1
1489	7	14	-	-	-
2912	15,11	9	8	-	-
16L8	10	20	-	-	-
20L10	12	24	-	-	-
74LS04	7	14	-	-	-
74LS74	7	14	-	-	-
74LS273	10	20	-	-	-
74LS374	10	20	-	-	-
82S191	12	24	-	-	-



Modem Circuit Diagram, Sheet 1 of 2

C-3



MODEM CIRCUIT DIAGRAM, SHEET 2 OF 2

APPENDIX D: MODEM PROGRAM LISTING

```
*PROTOTYPE MODEM FOR 1200 bd FSK.
*
* This version operates at a sampling frequency of 9600Hz.
*
* This program synthesises a 1200 bd FSK modem. The sampling
* frequency is 9600Hz; the mark and space frequencies are
* 1050Hz and 1925Hz respectively.
*
        IDT     'M96'
*
NCONST  EQU     8
NCINF   EQU     23      No. of Input Filter Coefficients
NCPDF   EQU     15      No. of Post Detection Filter Coefs
NCEQ    EQU     5       No. of Equaliser Coefficients
*
NX1     EQU     NCINF/2+1
NX2     EQU     NCPDF/2+1
NX3     EQU     NCEQ
NCOEF   EQU     NX1+NX2+NX3
*
LPCOEF  EQU     4069    Represents .993477 AS A Q12 Fraction
*
* Port Definition
*
FSKIP   EQU     0
FSKOP   EQU     0
DATAIP  EQU     1
DATAOP  EQU     1
TESTIP  EQU     1
FLAGOP  EQU     1
*
* Constants in Data RAM
*
        DSEG    CONSTS
XINF    BSS     NCINF-1 Locations for Input Filter Delays
XINFN   BSS     1       Last Input Filter Delay Position
*
XPDF    BSS     NCPDF-1 Locations for PDF Delays
XPDFN   BSS     1       Last PDF Delay Location
*
XEQ     BSS     32      Equaliser Delays
XEQN    BSS     1       Last Equaliser Delay
*
INFLTR  BSS     NCINF/2 Locations for Input Filter Coefs
INFILN  BSS     1       Last Input Filter Coef
*
PDF     BSS     NCPDF/2 Locations for Post Det Filter Coefs
PDFN    BSS     1       Last PDF Coefficient
*
EQ      BSS     4       Equaliser Coefficients
EQN     BSS     1       Last Equaliser Coefficient
*
TEMP1   BSS     1       Temporary Register 1
TEMP2   BSS     1       Temporary Register 2
*
SIGENV  BSS     1       Filtered Rectified Signal
SIGLIM  BSS     2       Limiter Output and DIFF Delays
*
DATIN   BSS     3       Input Data
DATOUT  BSS     1       Output Data
FSKOUT  BSS     1       Output FSK
*
```

```
ANGMEM  BSS     1       Angle
*
ONE     BSS     1       Constant of Value 1
TSTMSK  BSS     1       Mask for Special Control Inputs
ANGMSK  BSS     1       Angle Mask
REF     BSS     1       '1' or '0' Decision Level
*
RMPDIF  BSS     1       Differential Increment
RAMPHI  BSS     1       Angle Increment for High Freq
*
CARREF  BSS     1       Carrier Detect Reference
A       BSS     1       Equaliser Convergence Coefficient
        DEND
*
        RORG
*
        B       MODEM
*
        B       WTINT
*
DTBLE   DATA    1               The Value One
        DATA    >00F0           Test Mask
        DATA    >FFF            Angle Mask
        DATA    >5060           '1' or '0' Decision Level
        DATA    -93             Differential Angle Increment Divided by 4
        DATA    821             MARK (High) Freq Angle Increment
        DATA    >DD0            Carrier Detect Reference Level
        DATA    -1000           Equaliser Convergence Coefficient
*
COEF    EQU     $
*
* Filter Coefficients
*
CINFIL  DATA    16384,8622,-4574,-8256,-3154,137
        DATA    -1189,-1878,-130,978,397,-42
*
CPDF    DATA    16384,15734,13907,11239,8192,5245
        DATA    2781,1022
*
CEQ     DATA    1500,-6000,30000,-6000,1500
*
* Sine Table
*
TBLE    DATA    0,804,1606,2404,3196,3981,4756,5520,6270,7005,7723
        DATA    8423,9102,9760,10394,11003,11585,12140,12665,13160
        DATA    13623,14053,14449,14811,15137,15426,15679,15893,16069
        DATA    16207,16305,16364,16384
*
        DATA    16364,16305,16207,16069,15893,15679,15426,15137
        DATA    14811,14449,14053,13623,13160,12665,12140,11585
        DATA    11003,10394,9760,9102,8423,7723,7005,6270,5520,4756
        DATA    3981,3196,2404,1606,804,0
*
        DATA    -804,-1606,-2404,-3196,-3981,-4756,-5520,-6270,-7005
        DATA    -7723,-8423,-9102,-9760,-10394,-11003,-11585,-12140
        DATA    -12665,-13160,-13623,-14053,-14449,-14811,-15137
        DATA    -15426,-15679,-15893,-16069,-16207,-16305,-16364
        DATA    -16384
*
        69,-15426
        DATA    -15137,-14811,-14449,-14053,-13623,-13160,-12665
        DATA    -12140,-11585,-11003,-10394,-9760,-9102,-8423,-7723
```

```
        DATA    -7005,-6270,-5520,-4756,-3981,-3196,-2404,-1606,-804
*
* Initialise On-Board RAM to Zero's
*
MODEM   LARK    AR0,143
        LARP    0
        ZAC
ZMEM    SACL    *
        BANZ    ZMEM
*
* Load the Constants into Data Memory
*
        LARK    AR0,ONE
        LARK    AR1,NCONST-1    No. of Constants
        LACK    DTBLE
READK   LARP    0
        TBLR    *+,1
        ADD     ONE
        BANZ    READK
*
* Test for Special Requirements
*
        IN      TEMP1,TESTIP    These may be test tones,200bd/1200bd etc
        LAC     TSTMSK          up to 16 different operations. The settings
        AND     TEMP1           on the DIP switches determine the operation
        SACL    TEMP1           to be executed
*
* Read the Filter Coefficients into Data RAM
*
        LARK    AR0,INFLTR
        LARK    AR1,NCOEF
        LACK    COEF
RDCOEF  LARP    0
        TBLR    *+,1
        ADD     ONE
        BANZ    RDCOEF
        LARP    0
*
        SOVM                    Set Overflow Mode
*
        EINT
LOOP    NOP
        NOP
        NOP
        NOP
        B       LOOP
*
* Implement the Actual Modem
*
WTINT
*
READ    IN      XINF,FSKIP      Read Input FSK Ready for
        LAC     ONE,15           Input Filter
        XOR     XINF
        SACL    XINF            Invert Sign Bit
        IN      DATIN,DATAIP    Read Data in
        LAC     ONE,15
        XOR     FSKOUT
        SACL    FSKOUT
        OUT     FSKOUT,FSKOP    Output FSK
        LAC     ONE,7
        XOR     DATOUT          Flip the Overflow Bit
```

```
        SACL    DATOUT
        OUT     DATOUT,DATAOP   Output Data
*
* Filter the Input  (See Section 4.5.1)
*
* Input Filter for 1200 bd Modem. Has Bandwidth from
* 1000Hz to 2000Hz. The Coefficients are Stored in the
* Name INFIL, and the Delayed Values in XINF.
*
*
INFIL   ZAC
        LT      XINFN
        MPY     INFILN
        LTD     XINFN-1
        MPY     INFILN-1
        LTD     XINFN-2
        MPY     INFILN-2
        LTD     XINFN-3
        MPY     INFILN-3
        LTD     XINFN-4
        MPY     INFILN-4
        LTD     XINFN-5
        MPY     INFILN-5
        LTD     XINFN-6
        MPY     INFILN-6
        LTD     XINFN-7
        MPY     INFILN-7
        LTD     XINFN-8
        MPY     INFILN-8
        LTD     XINFN-9
        MPY     INFILN-9
        LTD     XINFN-10
        MPY     INFILN-10
        LTD     XINFN-11
        MPY     INFILN-11
        LTD     XINFN-12
        MPY     INFILN-10
        LTD     XINFN-13
        MPY     INFILN-9
        LTD     XINFN-14
        MPY     INFILN-8
        LTD     XINFN-15
        MPY     INFILN-7
        LTD     XINFN-16
        MPY     INFILN-6
        LTD     XINFN-17
        MPY     INFILN-5
        LTD     XINFN-18
        MPY     INFILN-4
        LTD     XINFN-19
        MPY     INFILN-3
        LTD     XINFN-20
        MPY     INFILN-2
        LTD     XINFN-21
        MPY     INFILN-1
        LTD     XINFN-22
        MPY     INFILN
        APAC
*
        SACH    TEMP1,1         Save Filter Output
*
* The Carrier Detector  (See Section 4.6.1)
```

```
*
* Rectify the Signal
*
        ABS
        SACH    TEMP2,1
        LAC     TEMP2,5
*
* Low Pass Filter the Rectified Signal
*
FIL     LT      SIGENV
        MPYK    LPCOEF
        APAC
        SACH    SIGENV,4
*
* Implement Carrier Detect
*
        SUB     CARREF,12
        BGEZ    CARR
        ZAC
        SACL    DATOUT
        B       OFLOW
CARR    LACK    2
        SACL    DATOUT
*
* Frequency Discriminator  (See Section 4.5.2)
*
* Limiter for use with Frequency Discriminator
*
        LAC     TEMP1
        BGEZ    POS
        ZAC
        SUB     ONE,14
        SACL    SIGLIM
        B       DIFFTE
POS     LAC     ONE,14
        SACL    SIGLIM
*
* Differentiate the Incoming FSK
*
DIFFTE  LAC     SIGLIM+1,14
        SUB     SIGLIM,14
        DMOV    SIGLIM
*
DET     ABS                     Detect the Signal
        SACH    XPDF,1
*
* Implement Post Detection Filter  (See Section 4.5.3)
*
* The Coefficients are Stored in PDF, and the Delayed Values in XPDF.
*
PDFIL   ZAC
        LT      XPDFN
        MPY     PDFN
        LTD     XPDFN-1
        MPY     PDFN-1
        LTD     XPDFN-2
        MPY     PDFN-2
        LTD     XPDFN-3
        MPY     PDFN-3
        LTD     XPDFN-4
        MPY     PDFN-4
        LTD     XPDFN-5
```

```
        MPY     PDFN-5
        LTD     XPDFN-6
        MPY     PDFN-6
        LTD     XPDFN-7
        MPY     PDFN-7
        LTD     XPDFN-8
        MPY     PDFN-6
        LTD     XPDFN-9
        MPY     PDFN-5
        LTD     XPDFN-10
        MPY     PDFN-4
        LTD     XPDFN-11
        MPY     PDFN-3
        LTD     XPDFN-12
        MPY     PDFN-2
        LTD     XPDFN-13
        MPY     PDFN-1
        LTD     XPDFN-14
        MPY     PDFN
        APAC
*
        SUB     REF,15          Adjust the signal to be symmetrical about
        SACH    XEQ,1            zero
*
* Adaptive Equaliser  (See Section 4.5.4)
*
* Implement Equaliser Portion
*
        ZAC
        LT      XEQN
        MPY     EQN
        LTA     XEQN-8
        MPY     EQN-1
        LTA     XEQN-16
        MPY     EQN-2
        LTA     XEQN-24
        MPY     EQN-3
        LTA     XEQN-32
        MPY     EQ
        SACH    TEMP1,1         x
*
* Detect Data at Output of Equaliser
*
        BGZ     ZERO
        LAC     ONE
        B       DOUT
ZERO    ZAC
DOUT    ADD     DATOUT
        SACL    DATOUT
*
* Calculate the error signal
*
        LAC     TEMP1
        BGZ     HI
        ADD     ONE,12          e=x-s
        B       CORR
HI      SUB     ONE,12          e=x-s
CORR    SACL    TEMP1
        LT      TEMP1
        MPY     A
        PAC
        SACH    TEMP1,1         -ae
```

```
*
* Adjust Coefficients
*
        LT      TEMP1
* 5
        ZALH    EQN
        MPY     XEQN
        APAC
        ADD     ONE,15
        SACH    EQN
* 4
        ZALH    EQN-1
        MPY     XEQN-8
        APAC
        ADD     ONE,15
        SACH    EQN-1
* 3
        ZALH    EQN-2
        MPY     XEQN-16
        APAC
        ADD     ONE,15
        SACH    EQN-2
* 2
        ZALH    EQN-3
        MPY     XEQN-24
        APAC
        ADD     ONE,15
        SACH    EQN-3
* 1
        ZALH    EQ
        MPY     XEQ
        APAC
        ADD     ONE,15
        SACH    EQ
*
* Implement Equaliser Delays
*
        DMOV    XEQN-1
        DMOV    XEQN-2
        DMOV    XEQN-3
        DMOV    XEQN-4
        DMOV    XEQN-5
        DMOV    XEQN-6
        DMOV    XEQN-7
        DMOV    XEQN-8
        DMOV    XEQN-9
        DMOV    XEQN-10
        DMOV    XEQN-11
        DMOV    XEQN-12
        DMOV    XEQN-13
        DMOV    XEQN-14
        DMOV    XEQN-15
        DMOV    XEQN-16
        DMOV    XEQN-17
        DMOV    XEQN-18
        DMOV    XEQN-19
        DMOV    XEQN-20
        DMOV    XEQN-21
        DMOV    XEQN-22
        DMOV    XEQN-23
        DMOV    XEQN-24
        DMOV    XEQN-25
```

```
        DMOV    XEQN-26
        DMOV    XEQN-27
        DMOV    XEQN-28
        DMOV    XEQN-29
        DMOV    XEQN-30
        DMOV    XEQN-31
        DMOV    XEQN-32
*
* Check for Overflows
*
OFLOW   BV      OVFLOW  (See Section 4.6.3)
*
* FSK Modulator  (See Section 4.3)
*
MOD     LAC     ONE,1
        AND     DATIN
        BZ      RTSLOW
        LAC     ONE
        AND     DATIN
        SACL    DATIN
        ADD     DATIN+1,1
        ADD     DATIN+2
        DMOV    DATIN+1
        DMOV    DATIN
        SACL    TEMP1
        LAC     RAMPHI
        LT      TEMP1
        MPY     RMPDIF
        APAC
        ADD     ANGMEM
        AND     ANGMSK
        SACL    ANGMEM
        LAC     ANGMEM,11
        SACH    TEMP1
*
        LACK    TBLE
        ADD     TEMP1           Pointer
*
        TBLR    FSKOUT
*
OUTSIN  EINT
        RET
*
RTSLOW  ZAC                     Idle Output
        SACL    FSKOUT
        B       OUTSIN
*
OVFLOW  SOVM
        ADD     ONE,7
        B       MOD
*
        END
```

APPENDIX E: FILTER COEFFICIENT LISTINGS

All the coefficients listed below, have been normalised with respect to the largest, in this case, always the central coefficient.

Table E.1 Coefficients of Modem Channel Input Filter

n	Coefficient
1	-.00256348
2	.0242310
3	.0596924
4	-.00793457
5	-.114624
6	-.0725708
7	.00836182
8	-.192505
9	-.503906
10	-.279175
11	.526245
12	1.00000
13	.526245
14	-.279175
15	-.503906
16	-.192505
17	.00836182
18	-.0725708
19	-.114624
20	-.00793457
21	.0596924
22	.0242310
23	-.00256348

Table E.2 Post Detection Filter Coefficients

n	Coefficient
1	.0623779
2	.169739
3	.320129
4	.500000
5	.685974
6	.848816
7	.960327
8	1.00000
9	.960327
10	.848816
11	.685974
12	.500000
13	.320129
14	.169739
15	.0623779

Table E.3 Equaliser Start-Up Coefficients

n	Coefficient
1	.050000
2	-.200000
3	1.00000
4	-.200000
5	.050000

Table E.4 Coefficients of Cosine-Shaped Channel Input Filter

n	Coefficient
1	.0170727
2	.0181253
3	-.00682835
4	0
5	.0289580
6	-.00436069
7	-.0541745
8	0
9	.0208515
10	-.243289
11	-.551940
12	-.288645
13	.530200
14	1.00000
15	.530200
16	-.288645
17	-.551940
18	-.243289
19	.0208515
20	0
21	-.0541745
22	-.00436069
23	.0289580
24	0
25	-.00682835
26	.0181253
27	.0170727

The pinouts, internal architecture, and full instruction set are summarised in the following pages. Note that the actual device in use is the TMS 32010, without internal ROM, and consequently, pin 3, the MC/$\overline{MP}$ pin, is held low.

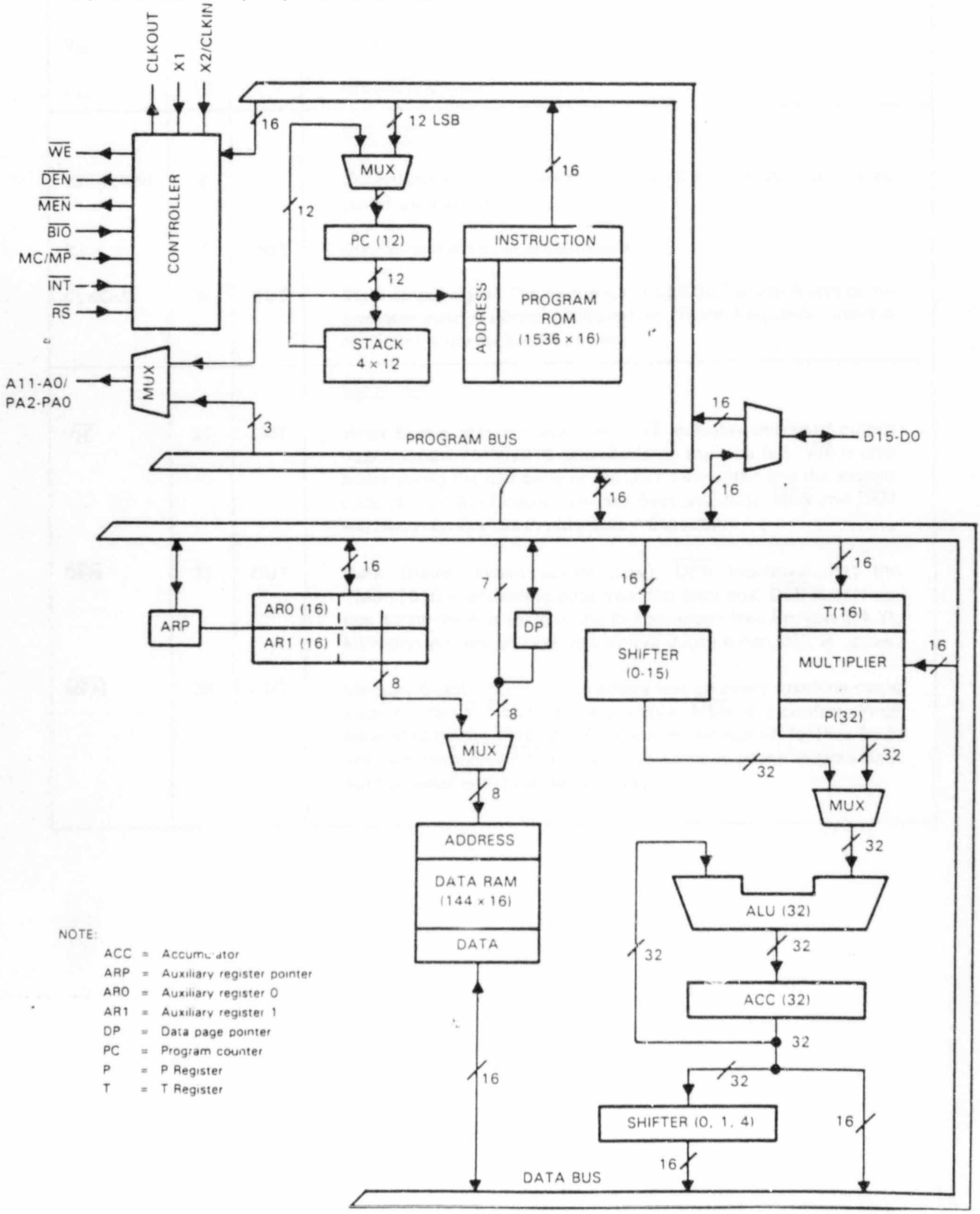


Fig F.1 Block Diagram of the TMS 320M10

Table F.1 TMS 32010 Pin Descriptions

SIGNAL	PIN	I/O	DESCRIPTION
			POWER SUPPLIES
V_{CC}	30		Supply voltage (+ 5 V NOM)
V_{SS}	10		Ground reference
			CLOCKS
X2/CLKIN	8	IN	Crystal input pin for internal oscillator (X2). Also input pin for external oscillator (CLKIN).
X1	7	OUT	Crystal input pin for internal oscillator
CLKOUT	6	OUT	Clock output signal. The frequency of CLKOUT is one-fourth of the oscillator input (external oscillator) or crystal frequency (internal oscillator). Duty cycle is 50 percent.
			CONTROL
$\overline{\text{WE}}$	31	OUT	Write Enable. When active (low), $\overline{\text{WE}}$ indicates that valid output data from the TMS32010 is available on the data bus. $\overline{\text{WE}}$ is only active during the first cycle of the OUT instruction and the second cycle of the TBLW instruction (see Section 3.4.3). $\overline{\text{MEN}}$ and $\overline{\text{DEN}}$ will always be inactive (high) when $\overline{\text{WE}}$ is active.
$\overline{\text{DEN}}$	32	OUT	Data Enable. When active (low), $\overline{\text{DEN}}$ indicates that the TMS32010 is accepting data from the data bus. $\overline{\text{DEN}}$ is only active during the first cycle of the IN instruction (see Section 3.4.3). $\overline{\text{MEN}}$ and $\overline{\text{WE}}$ will always be inactive (high) when $\overline{\text{DEN}}$ is active.
$\overline{\text{MEN}}$	33	OUT	Memory Enable. $\overline{\text{MEN}}$ will be active low on every machine cycle except when $\overline{\text{WE}}$ and $\overline{\text{DEN}}$ are active. $\overline{\text{MEN}}$ is a control signal generated by the TMS32010 to enable instruction fetches from program memory. $\overline{\text{MEN}}$ will be active on instructions fetched from both internal and external memory.

Table F.1 TMS 32010 Pin Descriptions (Continued)

SIGNAL	PIN	I/O	DESCRIPTION
			INTERRUPTS
$\overline{RS}$	4	IN	Reset. When an active low is placed on the $\overline{RS}$ pin for a minimum of five clock cycles, $\overline{DEN}$, $\overline{WE}$, and $\overline{MEN}$ are forced high, and the data bus (D15 through D0) is tristated. The program counter (PC) and the address bus (A11 through A0) are then synchronously cleared after the next complete clock cycle from the falling edge of $\overline{RS}$. $\overline{RS}$ also disables the interrupt, clears the interrupt flag register, and leaves the overflow mode register unchanged. The TMS32010 can be held in the reset state indefinitely.
$\overline{INT}$	5	IN	Interrupt. The interrupt signal is generated by applying a negative-going edge to the $\overline{INT}$ pin. The edge is used to latch the interrupt flag register (INTF) until an interrupt is granted by the device. An active low level will also be sensed. (See Section 2.10.)
$\overline{BIO}$	9	IN	I/O Branch Control. If $\overline{BIO}$ is active (low) upon execution of the BIOZ instruction, the device will branch to the address specified by the instruction (see Section 2.9).
			PROGRAM MEMORY MODES
MC/$\overline{MP}$	3	IN	Microcomputer/Microprocessor Mode. A high on the MC/$\overline{MP}$ pin enables the microcomputer mode. In this mode, the user has available 1524 words of on-chip program memory. (Program memory locations 1524 through 1535 are reserved.) The microcomputer mode also allows an additional 2560 words of program memory to reside off-chip. A low on the MC/$\overline{MP}$ pin enables the microprocessor mode. In this mode, the entire memory space is external, i.e., addresses 0 through 4095. (See Section 2.3.1.)
			BIDIRECTIONAL DATA BUS
D15	18	I/O	D15 (MSB) through D0 (LSB). The data bus is always in the high-impedance state except when $\overline{WE}$ is active (low).
D14	17	I/O	
D13	16	I/O	
D12	15	I/O	
D11	14	I/O	
D10	13	I/O	
D9	12	I/O	
D8	11	I/O	
D7	19	I/O	
D6	20	I/O	
D5	21	I/O	
D4	22	I/O	
D3	23	I/O	
D2	24	I/O	
D1	25	I/O	
D0	26	I/O	