The essential components of the STRIP - system

Dabekaussen, A.H.J.

Award date:
1988

FACULTEIT DER ELECTROTECHNIEK

TECHNISCHE UNIVERSITEIT

EINDHOVEN

Vakgroep Electronische Schakelingen

THE ESSENTIAL COMPONENTS OF THE STRIP - SYSTEM

by A.H.J. Dabekaussen

Report of a master's thesis carried out at
Philips Medical Systems Netherlands
from 1 january to 30 september 1988
under responsibility of Prof. Dr. Ir. W.M.G.
van Bokhoven and Ir. J.H. van den Boorn (TUE),
coached by J.H. den Boef and Prof. Dr. Ir.
A.F. Mehlkopf (PMSN)

Abstract

This report contains the description of several components of the Simple Transmitter Receiver with Increased Performance (STRIP) system. As an overview the basic concepts of Magnetic Resonance (MR) are described and the function of the major components of an MR instrument are reviewed.

The following components have been investigated: the image reject mixer, digital synthesizer and single side band modulator. Other components are available from external suppliers. Major goals in the design of the STRIP-components are to reduce the costs of the whole system and the ability to produce them by MSD.

An **image reject mixer** (IRM) mixes the received MR signal down to the low frequency band (10 kHz - 1 MHz) and suppresses the with noise afflicted image band at the same time. The major problem is to develop a device with an extreme ratio f_h/f_l, this with the demands of the future in mind; instant heart imaging and only one STRIP-system for different field strengths. Investigations have shown that it is possible to realize the IRM with relatively simple RC circuits.

The second subject concerns the **digital synthesizer**. A weighting between costs and performance was made. Also computer controlled correction factors in the digital part are introduced to correct for errors in the analog modulator. This approach ensures highest available performance with an extreme simple analog SSB modulator. The architecture of the synthesizer was designed in synchronous logic to be able to implement the device directly in an Application Specific Integrated Circuit (ASIC).

Finally the combination **single side band** (SSB) **modulator** and digital synthesizer confirmed the expectations with regards to the desired performance. The modulator was tested to verify its purpose; to mix the signal of the synthesizer up to the MR frequency with the image band suppressed, this to ensure no ghosts from other slices and no wasting of RF power.

Contents page

FOREWORD

Philips Medical Systems Netherlands is manufacturer of advanced medical equipment such as ultrasonic scanners, röntgen tomographs and magnetic resonance (MR) systems. MR has shown a growing evolution the last few years.

In this line of evolution a predevelopment group started end 1987 defining the system and requirements for a new generation MR-systems to be released in 1991. An important innovation in the spectrometer of this system is the Simple Transmitter Receiver with Increased Performance (STRIP) as suggested by Prof. Dr. Ir. A.F. Mehlkopf and J.H. den Boef.

A few important components of this STRIP-system are not yet available from external suppliers or are quite expensive, this with the requirements of the STRIP-system in mind. These components are economical remunerative to be produced by PMSN, therefore these are further investigated. Major goals in the design are to reduce the costs of the whole system and the ability to produce them by PMSN.

In this report a feasibility study is presented concerning the technical aspects of three components: image reject mixer, digital synthesizer and single side band modulator.

1. INTRODUCTION: MAGNETIC RESONANCE THEORY

In this chapter an overview is given of the basic magnetic resonance (MR) principles. This is done to get an impression of the overall system necessary to perform MR imaging and spectroscopy. Also the underlying arguments of the architecture of the Simple Transmitter Receiver with Increased Performance (STRIP) system are treated. Finally a description is made of the STRIP system.

1.1 Principles of magnetic resonance

Magnetic resonance was discovered in 1946, independently from each other, by Purcell and Bloch as a method to determine magnetic properties of atomic nuclei. After the discovery of differences in the proton MR parameters (relaxation times) by Damadian (1971) imaging techniques were first developed by Lauterbur (1973) [Lit. 1].

Since then, image quality has improved steadily while imaging time has shown a gradual decrease, bringing MR within the reach of clinical practice.

Magnetic resonance is a phenomenon that is found with atomic nuclei that exhibit a certain amount of spin. Under normal conditions all nuclei have a random orientation (as a result of thermal Brownian motion). When the nuclei are placed within a magnetic field a small amount tends to align themselves slightly more parallel than anti parallel with the direction of the field. Therefore a small net macroscopic magnetization is created parallel to the field. Because of the thermal motion only one in a million will participate in this process. This small amount is still sufficient to yield a measurable net magnetization because the human body contains about 10^{22} hydrogen nuclei per gram of tissue [Lit. 2].

In the model depicted here the alignment is not perfect and the magnetic moments of the protons rotate around the direction of the static magnetic field B_0 with a well defined frequency, the Larmor frequency:

$$\bar{\omega}_\lambda = - \gamma \cdot \bar{B}_0 \qquad (1.1)$$

with ω_λ = the Larmor frequency

γ = the gyro magnetic ratio, a constant for the nucleus involved

B_0 = the static magnetic field strength

Fig. 1.1 shows the situation described here.

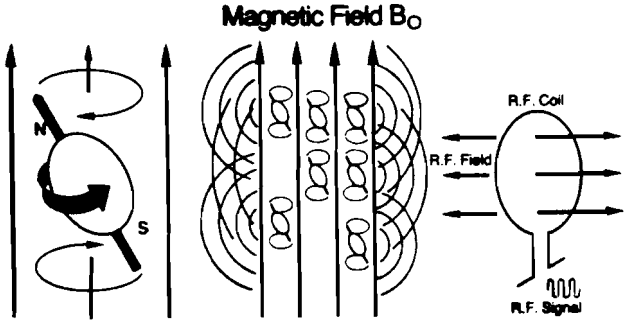


Fig 1.1: The alignment with the field B_0

The combined effect of these individual magnetic moments is called $\overline{M}$ the bulk magnetization. Fig. 1.2 gives the bulk magnetization $\overline{M}$, and its components M_x, M_y, M_z. $\overline{M}$ can be split into two components

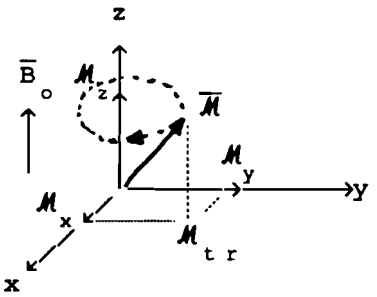


Fig. 1.2: The magnetization $\overline{M}$ and its components M_z and M_{tr}

M_z and M_{tr}, where the z-direction is the direction of the static field B_0. M_{tr} cannot be measured because the Larmor precessions are not in phase. To achieve an in phase Larmor precession of all spins, a second, time dependent magnetic field is applied orthogonally to the main magnetic field. This is done by means of an RF coil. A perturbation is caused which will drive the bulk magnetization $\overline{M}$ away from equilibrium. This RF field however must be time-dependent and matched to the Larmor resonance frequency.

The result of a so called 90° RF pulse is that the bulk magnetization $\overline{M}$ is rotated, along an axis perpendicular to the z-axis, by 90°. As a result the M_z component is rotated by 90° and starts rotating in the xy-plane with the Larmor frequency. The random phases of the spins will become coherent, in resonance with the applied RF frequency (the applied RF frequency is also the Larmor frequency).

When the RF pulse is switched off, two effects occur simultaneously:

- The magnetization will realign with the main field. The system returns to its equilibrium and $\overline{M}$ will increase:

$$M_z = M_o + (M_z - M_o)_{t=0} \exp\left(-\frac{t}{T_1}\right) \qquad (1.2)$$

where T_1 = the longitudinal or spin-lattice relaxation time.

M_o = magnetization $\overline{M}$ in equilibrium

t = 0; indicates the situation immediate after t=0.
In this case the remaining z-magnetisation $M_z = 0$.

- The separate Larmor precessions start to get out of phase due to very small differences in the precession frequencies of the individual nuclei. Therefore, the magnetization M_{tr} will decay rapidly:

$$M_{tr} = \left(M_x^2 + M_y^2\right)^{1/2} \qquad (1.3)$$

with:

$$M_x = (M_{tr})_{t=0} \exp\left(-\frac{t}{T_2}\right) \cos(\omega_\lambda t + \phi)$$

$$M_y = (M_{tr})_{t=0} \exp\left(-\frac{t}{T_2}\right) \sin(\omega_\lambda t + \phi)$$

and $\omega_\lambda = -\gamma B_0$

T_2 = the transverse or spin-spin relaxation time.

Notice that M_{tr} is given in its components M_x and M_y to illustrate the role of the Larmor frequency [Lit. 3].

A more precise definition of T_2 can be given if two effects are included. First T_2 caused by spin-spin interaction which causes local variations in the magnetic field. Secondly T_2^* as a result of inhomogeneities of the main field.

With a well chosen pulse sequence both effects can be made visible. For example an inversion recovery pulse sequence is used for measuring T_1. By applying a 180° pulse the M_z is inverted and starts realigning again. The time dependent magnitude of M_z can only be measured when a 90° pulse is used to turn M_z into the xy-plane.

For instance a spin echo pulse sequence is used to measure T_2. The result of such a spin echo sequence is shown in fig. 1.3. At any time when the magnetization M_{tr} is lost a 180° is applied to rephase the spins. T_2 is a result of the spin spin interaction and T_2^* is caused by field inhomogeneity.

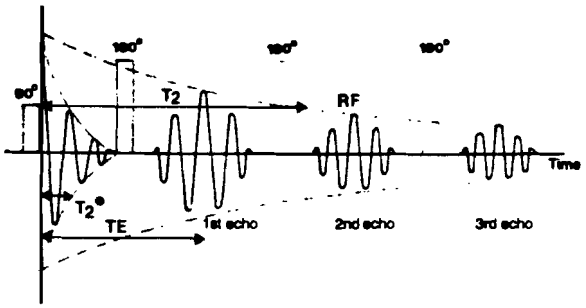


Fig. 1.3: Echoes as a result of rephasing 180° pulses

T_1 and T_2 depend on properties of the tissue. By measuring T_1 and T_2 various kinds of tissue can be made distinguished. An important application is now obvious, contrast is created mainly on the basic differences in T_1 and T_2. In tumors, for instance, both T_1 and T_2 are longer than in healthy tissue. Therefore a tumor is visible in an MR image.

A further extension of MR can be made when several other effects are taken into account. Other nuclei besides hydrogen also posses a certain amount of spin and can be made visible for instance sodium (Na). Another extension of MR is spectroscopy; in a small volume area of the body the metabolism can be studied by examining the spectra measured from that small volume. In these spectra the presence and concentration of several elements and chemical compounds can be determined.

With the methods described above it is possible to measure M_z. For imaging (and spectroscopy) however it is necessary to include spatial information. In order to obtain this information a gradient is superimposed on the static field B_0. MR-systems are normally equipped with 3 orthogonal gradient coils. Each gradient coil produces an additional magnetic field along the same axis as the static magnetic field B_0, but with a field strength which is linearly dependent on the coordinates. The total time dependent magnetic field can thus be represented by:

$$B(t) = B_0 + x\,G_x(t) + y\,G_y(t) + z\,G_z(t) \qquad (1.4)$$

By applying a gradient one can make the Larmor frequency spatial dependent. The first gradient that is used selects a layer in the z-direction, it is called the selection gradient. Exiting the spins with electromagnetic radiation within a narrow frequency band, by means of a specially shaped RF pulse (sinc pulse), confines the excitations to a particular layer. The coding in that layer in the x and y direction is done by the acquisition (G_x) and

evolution (G_y) gradient. Fig. 1.4 illustrates this situation.

During the period of the evolution gradient G_y nuclei will have their resonance frequencies altered according to their positions in the y-direction. If the gradient G_y is switched off and the preparation gradient G_x is then applied, nuclei at points along the y-direction will again precess at the same frequency, but the phases of the precessions will have been changed in a manner proportional to the local field strength experienced during the period of the evolution gradient G_y. For this reason the acquisition and evolution gradient are called frequency and phase encoding gradient. To get an impression of the gradient sequences necessary to encode an image fig. 1.5 is drawn.

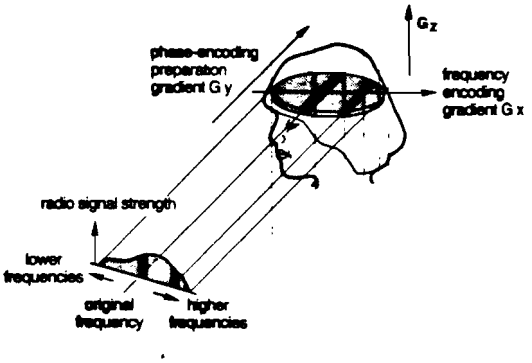


Fig. 1.4: Gradients are used to encode spatial information

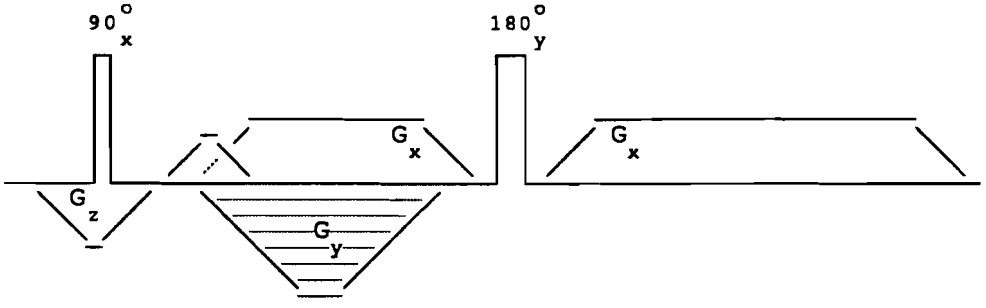


Fig. 1.5: A single echo multiple shot experiment

It may be clear from the foregoing that all information of the image is contained in a 2D-spectrum. For this reason a 2D Fourier transform may be applied to obtain a spectrum of the received MR signal. The image representing a mixture of proton density and the relaxation times T_1 and T_2, is the grey level representation of this spectrum. These steps are shown in fig. 1.6 .

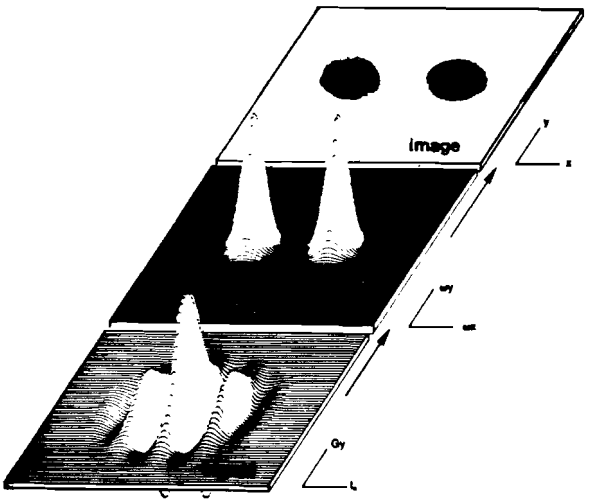


Fig. 1.6: Route from the signal to the image.

1.2 The basic instrument and the STRIP-system

In fig. 1.7 an overview is given of a basic instrument necessary for in-vivo imaging of the human body. The real time part consists of:

- the gradient system; gradient supplies, gradient coils to select layers and for spatial encoding.
- magnet system; mostly a super conductive electromagnet.
- shim coils; to adjust field homogeneity.
- transmit/receive system; consisting of transmitter, coils and receiver to generate RF pulses and to receive the MR signal.
- patient support; to control the position of the patient.
- physiology sensors for cardiac triggering, respiratory gating etc.
- control system to control the real time system.
- signal processing to perform on-line data reduction and in-line operations on the data.

Outside the dashed lines are:

- additional data processing
- initialization and presentation system to communicate with the user.
- archiving system to store different kinds of data.
- viewing system to display results or hard copy results.
- the user interface consisting of hardware and basic software to communicate with the user [Lit. 3].

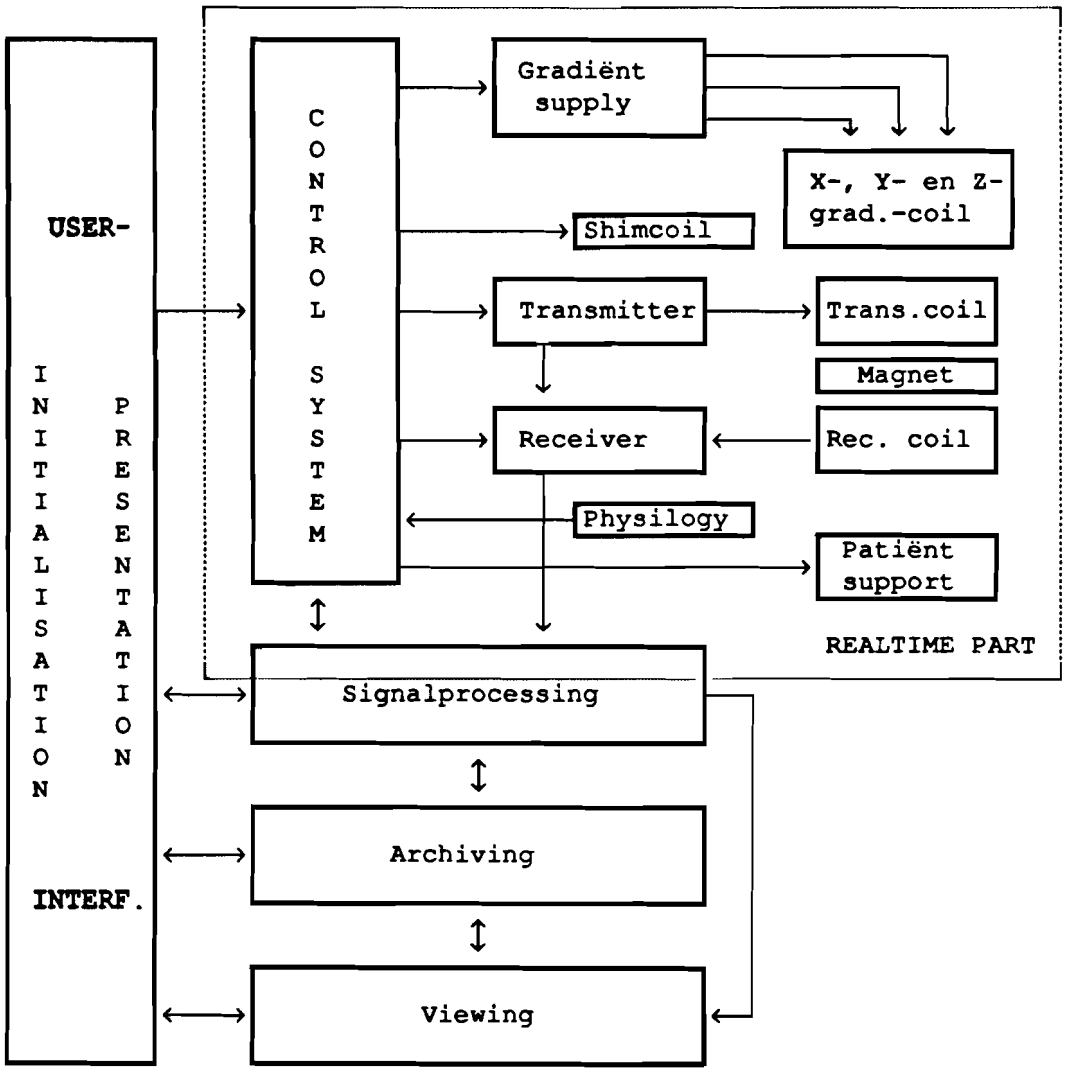


Fig. 1.7: Overview of an MR instrument

From the foregoing it may be clear that a transmit system is necessary to generate the RF pulses with the appropriate RF shape and frequency. In order to perform this operation a digital synthesizer can be used that is capable of amplitude, phase and frequency modulation in a phase continuous way. The signal of the synthesizer is then mixed upwards to the MR frequency by a single side band (SSB) modulator. An attenuator and power amplifier completes the transmit chain. A overview of the so called Simple Transmitter Receiver with Increased Performance (STRIP) is presented in fig. 1.8.

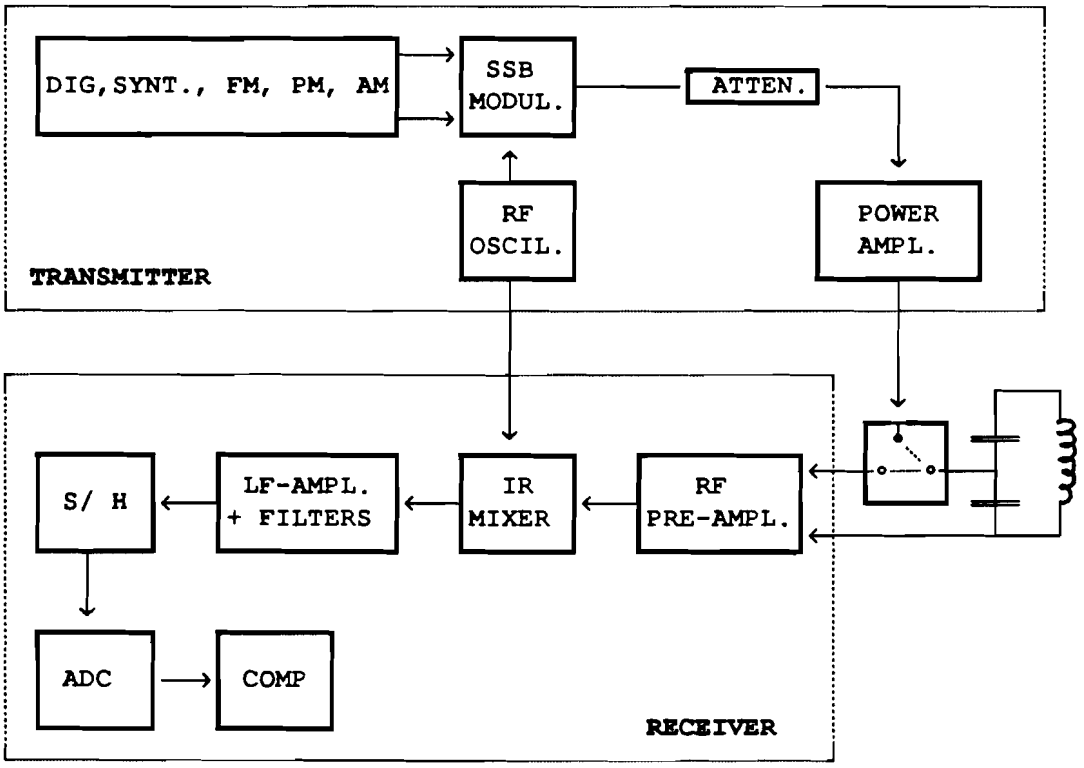


Fig. 1.8: Transmit/receive system according to the STRIP principle

The receive chain consists of a preamplifier with low noise figure, an image reject mixer (IRM) to mix the received MR signal down to almost base band (in a frequency band from 10 kHz - 1 MHz). An image reject mixer mixes the signal band to the desired output frequency band and suppresses noise from the image band at the same time. The receive chain is completed with LF amplifiers, sample and hold and the signal is digitized in an analog to digital converter. The computer is used to e.g. calculate the spectra and images from the raw data stream.

The STRIP architecture differs from a conventional architecture at the following points:
- the digital modulation of RF pulses by a digital synthesizer; this has the practical advantage of a better carrier and side band suppression.
- signals are not converted completely down to base band and are not located symmetrically around 0 Hz (as is the case with quadrature detection). Signals remain one sided from 0 Hz in almost base band (10 kHz - 1 MHz).
- signals after the IRM do not have the same phases as in a conventional quadrature detector. The signals have an extra phase caused by the offset frequency.This phase offset is not a problem because the offset phase is controllable by the digital synthesizer.

Because of its architecture the STRIP system has the following advantages:
- DC-drift, 1/f noise, hum, harmonics of hum do not produce artifacts in the MR images and spectra because these frequencies

are outside the spectral region of the sampled MR data.
- increased dynamic range because mixer input signals can be kept small as they are not disturbed by e.g. DC-drift.
- no quadrature detection and thus only one ADC channel.
- no quadrature artifact and no quadrature adjustments (in a quadrature system strong demands on the equality of both channels exists).
- no gain and bandwidth changes and thus no difficult filters and gain settings.
- no ghost-lines from intermodulation distortion from multiple strong lines because of the higher dynamic range.
- no ghost-lines from harmonic distortion of single lines because of the high absolute frequency of the sampled data. The harmonics fall outside the MR spectrum.
- no leak through from the transmit chain because during reception the transmitter can be switched to other frequencies.
- by operating at maximum sample rate it is possible to operate at maximum bandwidth and minimum gain. In this way maximum dynamic range is guaranteed.
- it is an all purpose system for imaging and spectroscopy (MRI/MRS).
- better reliability because the STRIP is a simple system.

Disadvantages:
- no real disadvantages are yet known [Lit. 3].

In this report three parts of the STRIP system are further developed. Chapter 2 describes the image reject mixer that operates on a very large high to low output frequency ratio. A basic element of such an IRM is a 90° phase shifter (quadrature hybrid). The major problem is to develop a hybrid for extreme frequency ratios. A very important part in the STRIP concept is the digital synthesizer, a design approach and considerations are described in chapter 3. Finally the SSB modulator capable of high spurious suppression is presented in chapter 4.

2. THE IMAGE REJECT MIXER

In this chapter the image reject mixer is described. This kind of mixer is used to mix a signal down to base band and suppress the image band at the same time (fig. 2.1). The major problem is to develop a device with an extremely large ratio f_h/f_l. Therefore first a general treatment of an image reject mixer is given to clarify the main principle. Secondly the theory behind the most important parts, the 90° phase shifters (quadrature hybrids) is given. Finally an image reject mixer to verify the principles is described.

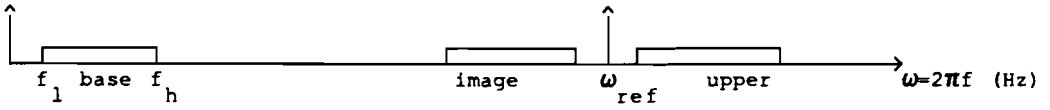


Fig. 2.1: Image rejection

2.1 Main principle and requirements

In fig. 2.2 the block diagram of an image reject mixer is sketched.The rejection of the image band can be demonstrated by the following calculation.

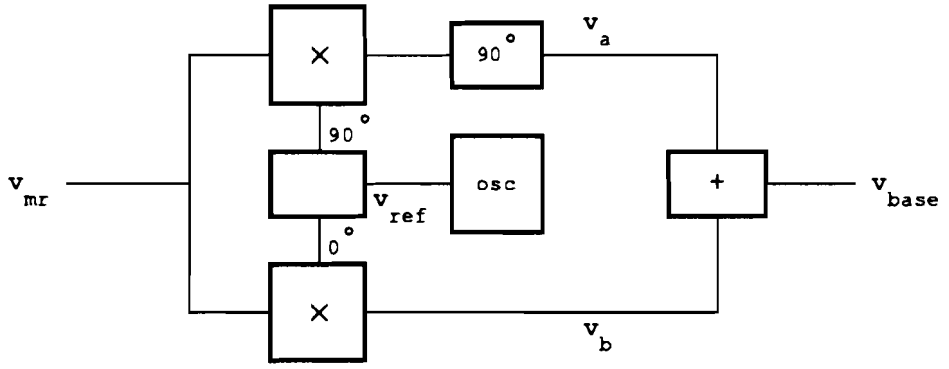


Fig. 2.2: Block diagram of an image reject mixer

Suppose: $v_{mr} = 2\,V_{mr}\cos(\omega_{mr}t + \phi_{mr})$ and $v_{ref} = \cos(\omega_{ref}t)$
then the signals V_a and V_b can be written as:

$$V_a = V_{mr}\{\mathrm{sgn}(\omega_{mr}-\omega_{ref})\cos[(\omega_{mr}-\omega_{ref})t + \phi_{mr}] - \cos[(\omega_{mr}+\omega_{ref})t + \phi_{mr}]\} \quad (2.1)$$

$$V_b = V_{mr}\{\cos[(\omega_{mr}-\omega_{ref})t + \phi_{mr}] + \cos[(\omega_{mr}+\omega_{ref})t + \phi_{mr}]\} \quad (2.2)$$

Addition after the 90° phase shift shows the result:

$$V_{base} = V_a + V_b = V_{mr}[1 + \mathrm{sgn}(\omega_{mr}-\omega_{ref})]\cos[(\omega_{mr}-\omega_{ref})t + \phi_{mr}]] \quad (2.3)$$

So in the ideal case frequencies above ω_{ref} are mixed down and frequencies underneath are extinguished. The reason to use image suppression is that the image band is afflicted with noise due to the fact that the preceding device in the system is a broad band

pre-amplifier. In the case of a non image reject mixer this noise is also mixed down to base band. The desired image rejection therefore can be expressed in terms of the maximum allowable deteriorated signal to noise ratio (SNR) at the output. The total effective receiver noise should only amount to at least 1 LSB at the ADC to ensure a negligible contribution of the ADC quantization noise, while maintaining the maximum possible dynamic range of the ADC (if noise contribution is less than 1 LSB, quantization noise will dominate).

It is known that if the effective receiver noise at the ADC equals 1 LSB, noise caused by digitizing amounts to 4% [Lit. 4]. With the 4% quantization noise in mind it may be clear that 1% noise contribution from the image band can be allowed:

$$N_{eff\ base} = \sqrt{N_{eff\ upper}^2 + N_{eff\ image}^2} \qquad (2.4)$$

with: $N_{eff\ base}$ = effective total noise power

$N_{eff\ image}$ = effective noise power in the image band

$N_{eff\ upper}$ = effective noise power in the upper band

If stated that the allowable increase of the total noise power level caused by the image band is 1%, one finds:

$$N_{eff\ base} / N_{eff\ upper} = \sqrt{1 + \frac{N_{eff\ image}^2}{N_{eff\ upper}^2}} = 1.01 \qquad (2.5)$$

thus:

$$N_{eff\ image}^2 / N_{eff\ upper}^2 = 0.02 = -16.9\ dB \qquad (2.6)$$

From this it can be concluded that an image rejection of at least 17 dB is required. The main design goal of the image reject mixer is to perform at the strongest requirement that can be made today or in the near future. It is expected that "snapshot imaging" of the heart demands the largest bandwidth [Lit. 5, 6]. Therefore the image reject mixer should operate at a very broad frequency range.

The bandwidth required with these kinds of experiments can be roughly estimated. Acquisition time of each echo is 300 μsec and in this time 128 complex samples are taken [Lit. 6]. The STRIP receiver doesn't use quadrature detection therefore 256 real samples are necessary. These 256 samples are cut from a larger set of samples (e.g. 512 because of the structure of the FFT) to reject DC, hum, 1/f noise etc. Thus the bandwidth BW according to sample rate f_s is:

$$BW = \frac{f_s}{2} = \frac{1}{2} \Big/ \frac{300 \cdot 10^{-6}}{512} = \frac{1.7 \cdot 10^6}{2} = 853\ kHz \qquad (2.7)$$

If a factor 20% in over account is taken, it can be concluded that a BW of 1 MHz at the output of the image reject mixer will be

sufficient also in the near future.

The high mixing frequency range depends on the strength of the static magnetic field and the gyro magnetic ratio of the elements that are measured. Table 2.1 gives various gyro magnetic ratios.

$$f = \gamma * B_0 \qquad (2.8)$$

with:

f = resonance frequency of the measured element in MHz
γ = gyro magnetic ratio in MHz/T
B_0 = static magnetic field in T

Table 2.1: Gyro magnetic ratio for various elements [Lit. 1]

Isotopes with spin:	gyromagnetic ratio (Mhz/T):
^{1}H	42.58
^{14}N	3.08
^{31}P	17.24
^{13}C	10.71
^{23}Na	11.26
^{39}K	1.99
^{17}O	5.77
^{2}H	6.53
^{19}F	40.05

It can be calculated that a high frequency bandwidth of 10 Mhz - 90 Mhz is sufficient to cover a full range of experiments involving magnetic field strengths from 0.25 T - 2 T.

Another problem is the different delay that signals of different frequencies undergo in the receiver chain. This group delay should be equal for different frequencies in the ideal case and implies a linear phase decline in the frequency region of interest. A nonlinear phase decline is difficult to correct for and could cause unacceptable artifacts. For spectroscopy this frequency region is 10 kHz at maximum and the allowable tolerance on the linear phase is about 1°. For imaging the allowable tolerance is 5° and the frequency region of interest depends on the kind of experiment but may be as high as 500 kHz.

Further requirements such as noise figure and spurious free dynamic range strongly depend on the performance of the total receiver chain and are therefore of secondary importance. Major goal is the desired image rejection over the extreme bandwidth. The requirements are summarized in table 2.2.

Table 2.2: Requirements on the image reject mixer

Item:	Specification:
MR frequency range	10 - 90 MHz
LF frequency range	10 kHz - 1 MHz
image rejection	> 17 dB
phase distortion	< 1° for spectroscopy BW ≤ 10 kHz
	< 5° for imaging

From these requirements on the image rejection an estimation on the phase errors that are allowed in the 90° phase shifters can be made. Vice versa, if an ideal 90° phase shifter is assumed a certain amplitude unbalance can be tolerated to achieve an image rejection of 17 dB. Fig. 2.3 plots the allowed tolerances [Lit. 7].

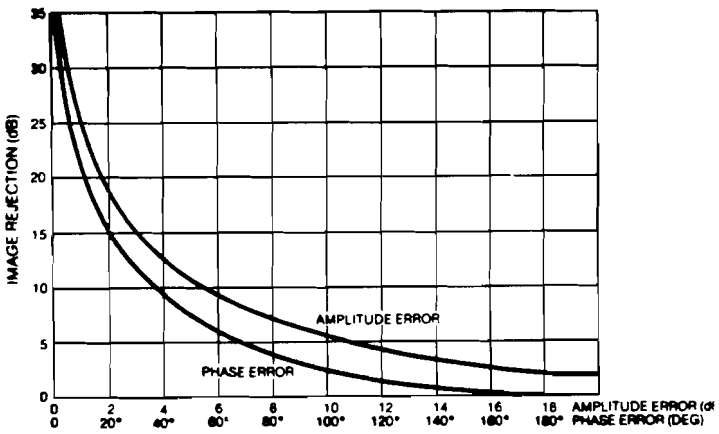


Fig. 2.3: Image rejection vs. phase and amplitude error

Mixers are available as of the shelf products in the desired frequency range so the major problem arises in the construction of the extreme wide band quadrature splitters (90° phase shifters). This is the subject of the next paragraph.

2.2 Alternatives to realize wide band 90 degree phase shifters

Several ideas from the literature have been studied for a convenient solution. In table 2.3 these ideas have been summarized to give an overview. For further investigation references are given to the literature.

Several of these possibilities have also been studied by S.R.J. Thorn [Lit. 26]. PLL like principles don't seem to be suitable because of the circuit complexity and the expected phase noise. Experiments were carried out on the travelling wave divider. This device is difficult to build by hand but more suited for integration.

Table 2.3: Wide band 90° phase shifters

Principle:	Explanation:	Remarks:
PLL like principle:		
PLL	signals at phase detector are shifted 90°	can only be used as HF hybrid because of fixed amplitude, phase noise [Lit. 7, 25]
Travelling wave divider	four stage ring oscillator by parasitic coupled transistors	idem [Lit. 8, 25]
Digital frequency divider	2 flipflops divide by 2, outputs are shifted	idem odd harmonics of square wave are only 9.54 dB down by fundamental [Lit. 8, 25]
Filter concepts:		
Hilbert transformer	tapped delay line	very complex, lots of delay line sect. [Lit. 9, 10]
	convert up and filter	degraded performance because signal is mixed up and crystal filtered [Lit. 11]
Allpass filters	with LC filters	[Lit.12, 13, 14, 15]
	with RC filters	[Lit.16,17,18,19,25]
	with LRC filters	[Lit. 20, 21]
Transformer cicuit	with LC filters	[Lit. 22, 23, 24]
Various:		
Use capacitance	voltage and current are shifted 90°	amplitude correction necessary
Use varicaps	idem	Brucker spectrometer rather complex
Use integrator	cos integrated is sin	amplitude correction
Stripline	transformer like circuit	used at higher frequencies > 100 MHz [Lit. 24]
Digital synthesizer	create sin and cos output, use filtered harmonics because periodicy in freq. domain	fixed amplitude, expensive, phase noise

The digital solution is theoretically impossible because of the odd harmonics of a square wave. The first odd harmonic is only 9.54 dB down in comparison with the fundamental frequency. Therefore noise at the harmonic frequency also mixes down. This also has been verified experimentally.

The most promising solutions are the all pass filters and these are further investigated.

2.3 Fundamental phase splitting circuits

All pass filters have several properties which make them suitable to serve as wide band phase splitting circuits. This will be explained.

Simple first order networks such as $1+j\omega\tau$, $1/(1+j\omega\tau)$ and $(1-j\omega\tau)/(1+j\omega\tau)$ all have poles and zeros on the real axis and therefore transfer functions with a $\phi = \arctan(\omega\tau)$ phase decline. This is sketched in fig. 2.4. Only all pass structures have constant amplitude behavior. Such a $\phi = \arctan(\omega\tau)$ phase decline can be approximated by a straight line for a certain frequency span on a logarithmic plot.

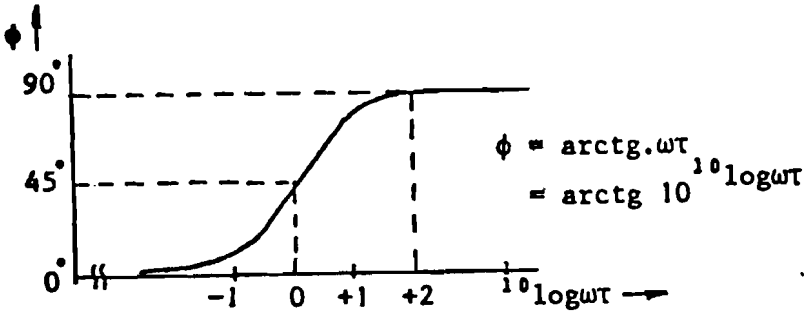


Fig. 2.4: Logarithmic frequency versus phase plot

A frequency independent phase shift of arbitrary amount can be construct from networks with a linear phase versus frequency plot. The phase difference between two of these networks is then frequency independent in a limited frequency span. This is illustrated by fig. 2.5. The question arises how to construct such a linear phase decline with the aid of only $\phi = \arctan(\omega\tau)$ phase behavior. As stated before a logarithm can be approximated by an arctan over a certain frequency span.

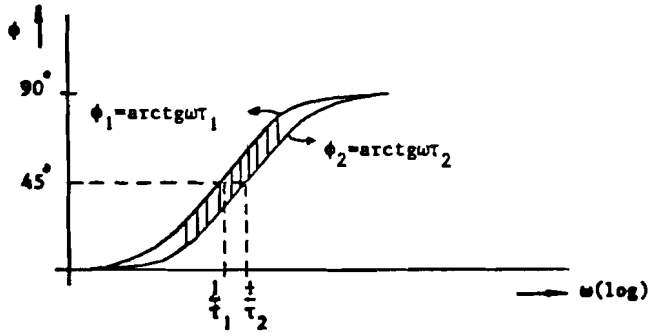


Fig. 2.5: Constant phase difference over limited frequency span

If the poles and/or zeros of such a network are divided on a logarithmic basis (in frequency) it is possible to derive a network with a prescribed phase tolerance over a certain bandwidth. This is shown in fig 2.6. The more sections are used the less phase ripple on the ideal line will occur. The basic construction for a 90° phase shifter is shown in fig. 2.7. Ideal infinite sections of the networks A and B have a logarithmic phase

decline but the phase difference between the networks is chosen to be 90°.

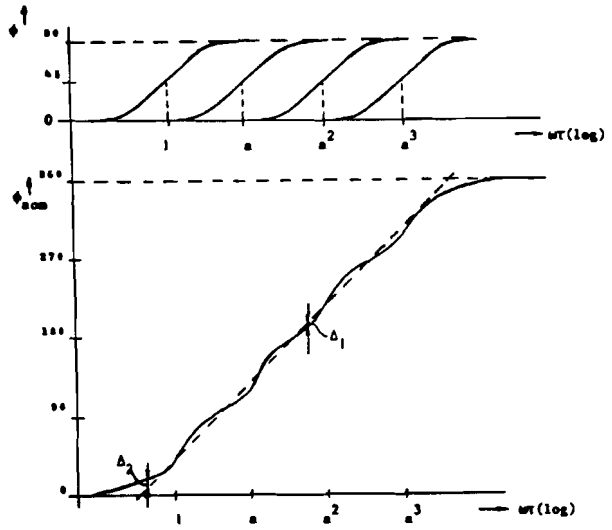


Fig. 2.6: Phase tolerance over a certain bandwidth

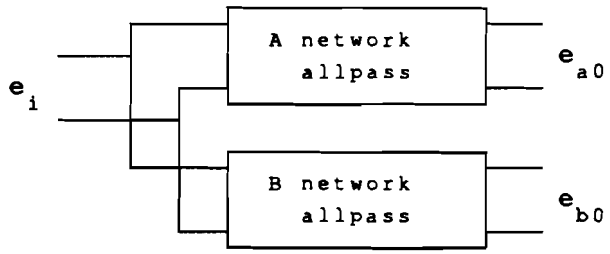


Fig. 2.7: Basic construction of the 90° phase shifter

The major problem arises in the calculation of the poles of the different sections. Since in the foregoing it was assumed that over a certain frequency span an unlimited number of poles could be geometrically arranged. With a limited number of poles a linear phase decline should be approximated as closely as possible. Several authors have published methods of approximation. One of the most commonly used methods is to use Chebychev polynomials and elliptic functions [Lit. 13, 14, 15, 16, 17, 18, 19]. After the approximation the problem of the network synthesis must be solved.

This can be done in several ways, using several kinds of networks consisting of RC, LC or RLC components. In the following first a RC network derived by the above mentioned method will be described to illustrate the logarithmic behavior. Afterwards the method will be discussed and extended and finally a FORTRAN program that can be used to obtain results is provided.

2.4 Analysis of a basic phase shift network

A basic phase shift section is sketched in fig. 2.8. Notice that 2 of these sections, the A network and the B network are required to

obtain a complete 90° phase shifter (fig. 2.7).

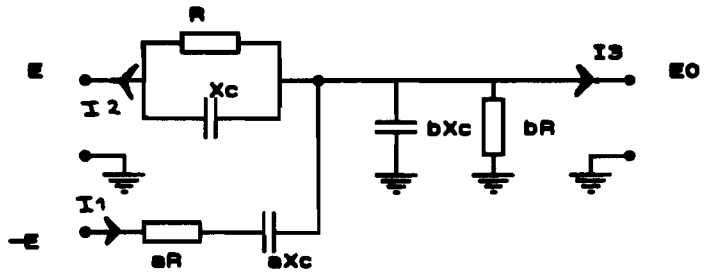


Fig. 2.8: Basic phase shift network

The three currents shown in fig. 2.8 may be written as follows:

$$I_1 = -(E_0 + E)/\,a(R-jX_c) \tag{2.8}$$

$$I_2 = (E - E_0)(R-jX_c)/jRX_c \tag{2.9}$$

$$I_3 = -E_0(R-jX_c)/jbRX_c \tag{2.10}$$

Kirchoff current law states:

$$I_1 = I_2 + I_3 \tag{2.11}$$

Substituting for I_1, I_2 and I_3:

$$-(E_0 + E)/a(R-jX_c) = (E - E_0)(R-jX_c)/jbRX_c - E_0(R-jX_c)/jbRX_c \tag{2.12}$$

Equation 2.12 may be solved for E_0, the output signal:

$$E_0 = \frac{b}{b+1} E \frac{a(R^2-X_c^2) + jRX_c(1-2a)}{a(R^2-X_c^2) - jRX_c\frac{(b+2a+2ab)}{(b+1)}} \tag{2.13}$$

The magnitude of E_0 is independent of X_c (all pass structure) when:

$$(b + 2a + 2ab)/(b+1) = 1 - 2a \tag{2.14}$$

Therefore, with the case as in eq. 2.14:

$$E_0 = \frac{b}{b+1} E \arg\left(2 \arctan\left\{ \left[\frac{1-2a}{a} \right] \frac{RX_c}{R^2-X_c^2} \right\} \right) \tag{2.15}$$

By substitution of:

$$a = 1/(M+2) \tag{2.16}$$

$$b = (M-2)/4 \tag{2.17}$$

$$X_c/R = \omega_{0m}/\omega \tag{2.18}$$

Equation 2.15 becomes:

$$E_0 = -\frac{M-2}{M+2} E \arg\left(2 \arctan\left\{ \frac{M}{\frac{\omega_{0m}}{\omega} - \frac{\omega}{\omega_{0m}}} \right\} \right) \tag{2.19}$$

Fig 2.9 illustrates several phase versus normalized frequency

ω/ω_{om} characteristics obtainable with the network of fig. 2.8. The family of curves is derived by variation of the parameter M over the range indicated. To achieve the best approximation of a straight line the value M = 3.8 is the most suitable [Lit. 19].

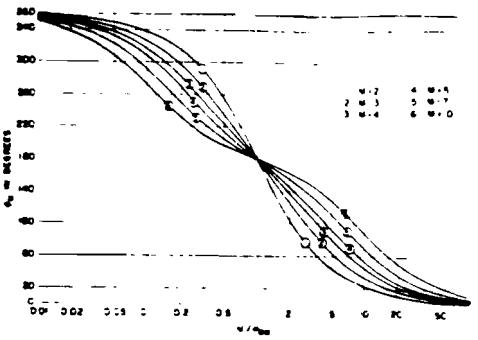


Fig. 2.9: Normalized phase shift characteristics obtainable with the network of fig. 2.8.

2.5 Approximation techniques for practical design of a phase splitter

Several authors have proposed techniques for the design of a 90° phase shifter. The most convenient solutions are proposed by H.J. Orchard [Lit. 13] and D.K. Weaver [Lit. 17]. These circuits were simulated, the first uses LC all pass networks as shown in fig. 2.10., the latter RC networks.

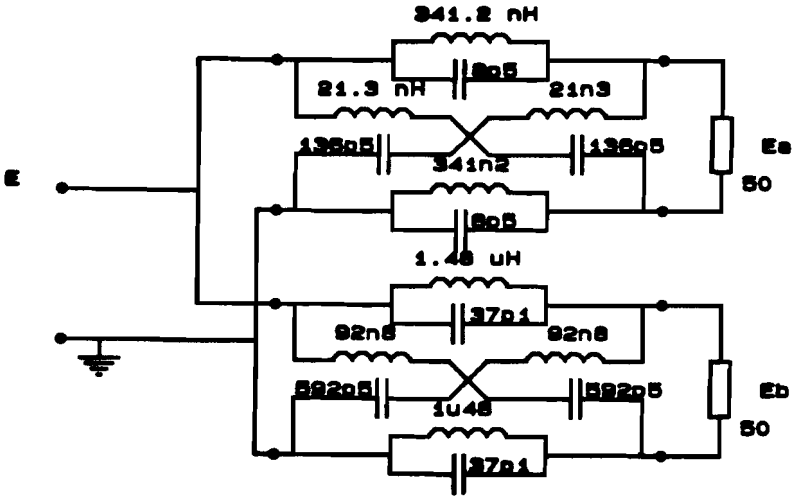


Fig 2.10: Orchard LC 90° phase splitter 10 Mhz - 200 Mhz

The Orchard LC networks have the following disadvantages:
- LC all pass networks are less suited for 10 kHz - 1 Mhz because of the large inductors.
- the networks have balanced outputs, this is inconvenient and should be altered to a unbalanced version. This is possible and described in [Lit. 10].
- the networks contain inductors which generally should be avoided

if possible because of decreased performance in comparison with capacitors (large tolerances). They also pose greater problems in production.

The results of the simulation can be found in appendix 1.

The most elegant solution for both the low frequency device (10 kHz - 1 Mhz) and the high frequency device (10 Mhz - 90 Mhz) is found in networks of the kind as proposed by D.K. Weaver [Lit. 17]. They consist of RC sections as drawn in fig. 2.11. His method will be described and explained because of the short and "cookbook" fashion in the article, further it will be extended to an alternative network synthesis.

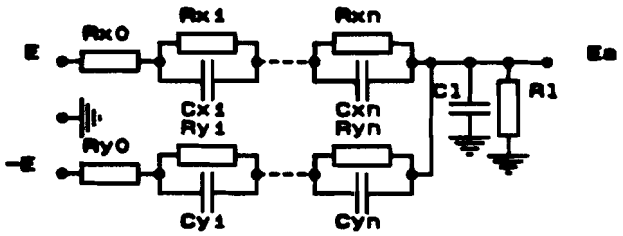


Fig. 2.11: Weaver RC all pass network

2.6 RC quadrature hybrid design procedure

Two all pass networks, connected as shown in fig. 2.7 can be designed to have a constant phase difference over a certain frequency span. The Chebychev type of equal ripple approximation is used, and only the special case of approximating a 90° phase difference will be covered. The networks can be further restricted so that they can be constructed with resistances and capacitances only.

The general RC impedance function is given by:

$$Z_{RC}(p) = \frac{a_n}{b_n} \prod_{k=1}^{n} \frac{p-z_k}{p-p_k} \qquad z_k, p_k < 0 \qquad (2.20)$$

We may say that all RC impedance functions have simple poles and zeros alternating on the negative real axis of the p-plane [Lit. 27]. The general response functions of the two all pass networks having RC restriction are given by:

$$f_a(p) = \frac{e_{a0}}{e_i} = K_a \frac{(p-\sigma_{a1})(p-\sigma_{a2})(\ )(\ )}{(p+\sigma_{a1})(p+\sigma_{a2})(\ )(\ )} \qquad (2.21)$$

$$f_b(p) = \frac{e_{b0}}{e_i} = K_b \frac{(p-\sigma_{b1})(p-\sigma_{b2})(\ )(\ )}{(p+\sigma_{b1})(p+\sigma_{b2})(\ )(\ )} \qquad (2.22)$$

The problem of design can be divided into 2 main parts. In the

first part the pole zero pairs (σ's) are calculated. This determines the response function.The second part is to find an RC synthesis that realizes this function.

The calculation of the pole zero pairs should be preceded by a determination of the network complexity n from bandwidth-ripple curves (fig. 2.12). The bandwidth ratio is defined as:

f_h/f_l =upper band edge/lower band edge (2.23)

e = phase tolerance in the band in degrees (2.24)

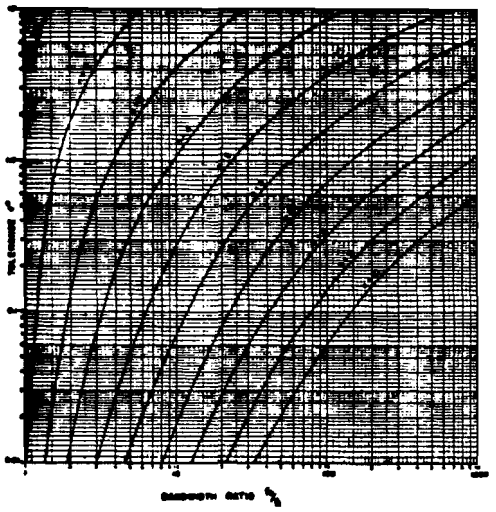


Fig. 2.12: Bandwidth-ripple curves

This kind of bandwidth-ripple curves has been published by several authors [Lit. 13, 15, 16, 17]. Once the network complexity n is known, the next step is the calculation of the poles and zeros (σ's). Locating the pole-zero pairs of the phase difference function on the real axis to achieve the optimum solution involves the Chebychev approximation and the use of an elliptic function for conformal mapping of the p-plane into the u-plane where the Chebychev approximation is taking place.

The Jacobi elliptic functions are doubly periodic functions of a complex variable u. The elliptic functions combine the properties of circular and hyperbolic functions and have real and imaginary periods. In contrast to trigonometric functions, which have the property of being repeated in strips of the complex plane, the properties of the doubly periodic functions are repeated in parallelograms or rectangles [Lit. 28].

$$p = tn(z,k) \qquad (2.24)$$

This is the elliptic tangent transformation and maps the p-plane in a doubly periodic fashion. The mapping of the ω-axis branches twice at right angles in such a way that a band of the ω-axis maps parallel to the σ-axis. To obtain 90° phase difference the poles and zeros are alternated with equal spacing along the mapping of

the σ-axis and should obey the rules of the RC all pass function realizability [Lit. 13].

If a certain number of poles and zeros are taken the locations are completely determined by the above procedure. The bandwidth ratio f_h/f_l determines the modulus k of the elliptic tangent. After these pairs are located in the u-plane they are found in the p-plane using eq. 2.24. Then the desired transfer function is known. Also Jacobian Theta functions can be used for approximations in numerical calculations of the elliptic functions [Lit. 18]. The following formulas use this kind of approximation.

$$k' = f_l/f_h \tag{2.25}$$

$$k = \sqrt{1 - k'^2} \tag{2.26}$$

$$m = \frac{1}{2}\,\frac{1 - \sqrt{k}}{1 + \sqrt{k}} \tag{2.27}$$

$$q' = m + 2\cdot m^5 + 15\cdot m^9 + \ldots \tag{2.28}$$

$$q = \exp(\pi^2/\ln(q')) \tag{2.29}$$

$$\phi_{ar} = \frac{90}{2n}(4r-3) \qquad \text{for } r = 1, 2, \ldots, n/2 \text{ if } n \text{ is even}$$
$$\text{or } r = 1, 2, \ldots, (n+1)/2 \text{ if } n \text{ is odd} \tag{2.30}$$

$$\phi_{br} = \frac{90}{2n}(4r-1) \qquad \text{for } r = 1, 2, \ldots, n/2 \text{ if } n \text{ is even}$$
$$\text{or } r = 1, 2, \ldots, (n-1)/2 \text{ if } n \text{ is odd} \tag{2.31}$$

$$\phi_{ar}' = \arctan \frac{(q^2-q^6)\sin 4\phi_{ar}}{1+(q^2+q^6)\cos 4\phi_{ar}} \qquad \text{A network} \tag{2.32}$$

$$\phi_{br}' = \arctan \frac{(q^2-q^6)\sin 4\phi_{br}}{1+(q^2+q^6)\cos 4\phi_{br}} \qquad \text{B network} \tag{2.33}$$

$$\sigma_{ar} = \frac{1}{\sqrt{k'}} \tan(\phi_{ar} - \phi_{ar}') \tag{2.34}$$

$$\sigma_{br} = \frac{1}{\sqrt{k'}} \tan(\phi_{br} - \phi_{br}') \tag{2.35}$$

With these the response functions of the A network and B network are given by:

$$f_a(p) = K \prod_{r=1}^{n/2 \text{ or } (n+1)/2} \frac{(p-\sigma_{ar})}{(p+\sigma_{ar})} \tag{2.36}$$

$$f_b(p) = K \prod_{r=1}^{n/2 \text{ or } (n-1)/2} \frac{(p-\sigma_{br})}{(p-\sigma_{br})} \tag{2.37}$$

Next, the maximum K should be determined to realize the network with RC elements only. The network will have a certain minimum attenuation depending on the value of K. This attenuation should be minimized:

- find values of p for which $d/dp(f_a(p)) = 0$
- substitute the negative values of p into the function $f_a(p)/K$, the resulting numbers having magnitudes greater than 1. The maximum number is the reciprocal of K_a'.
- repeat this for $f_b(p)$, if n is even the maximum should be the same ($K = K_a' = K_b'$). If n is odd and K_b' comes out different choose the smaller ($K = \min(K_a', K_b')$).

2.7 The synthesis of the RC quadrature hybrid

The next few steps concern the synthesis of the practical network.The realization problem does not have an unique result, there are several possible configurations . Two of them will be discussed. A convenient configuration to use is the half-lattice driven by a balanced voltage source (fig. 2.13)[Lit. 10]. In practice the balanced source can be made by operational amplifiers in the low frequency case or transformers at high frequencies.

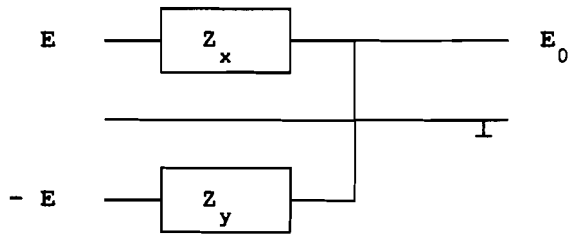


Fig. 2.13: All pass half lattice network

The response function of fig. 2.13 is given by:

$$f(p) = \frac{1 - Z_x/Z_y}{1 + Z_x/Z_y} \tag{2.38}$$

Solving this for Z_x / Z_y we obtain:

$$\frac{Z_x}{Z_y} = \frac{1 - f(p)}{1 + f(p)} \tag{2.39}$$

Values of p that make $1 - f(p) = 0$ must either be zeros of Z_x or poles of Z_y. Similarly solutions of $1 + f(p) = 0$ must be poles of Z_x or zeros of Z_y. The impedance functions Z_x and Z_y have to contain only resistances and capacitances and therefore all their poles and zeros are on the negative real axis. This condition can be satisfied if K_a and K_b in eq. 2.21 and eq. 2.22 are small enough [Lit. 17]. This implies that the network will have a certain minimum attenuation depending on the values found for K_a

and K_b. If for the case of n=4, K is computed it is found that K= 0.45 Notice that the case of n=4 has been treated in paragraph 2.4..The factor b/(b+1) = 0.45 from eq. 2.13 and M = 3.8 from eq. 2.17 thus agree with this method.From now on the procedure should be repeated for the A network and the B network, because synthesis in principle is identical for both. Therefore $f_a(p)$ and $f_b(p)$ are represented by f(p).

The solutions of:

$$f(p) - 1 = 0 \qquad (2.40)$$

are called σ_{01}, σ_{23}, σ_{32}, σ_{45}, σ_{54} etc. such that $|\sigma_{01}| > |\sigma_{23}| > |\sigma_{32}| > \ldots\ldots$ etc.

The solutions of:

$$f(p) + 1 = 0 \qquad (2.41)$$

are called σ_{12}, σ_{21}, σ_{34}, σ_{43}, σ_{56} etc. such that $|\sigma_{12}| > |\sigma_{21}| > |\sigma_{34}| > \ldots\ldots$ etc.

The first realization will have the structure of fig. 2.14a. and will be derived from the basic network in fig. 2.13 by extracting a load impedance (R_1 and C_1 according to fig. 2.15).

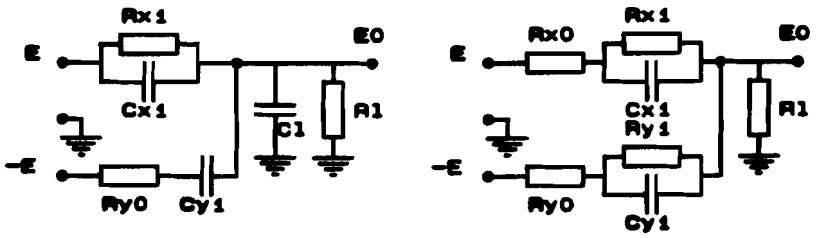


Fig. 2.14: a) network for n=4 without resistive source impedance b) network for n=4 with resistive source impedance

Two impedance functions of the form of eq. 2.42 and eq. 2.43 should be constructed, alternately dividing the poles between the numerator of Z_x and the denominator of Z_y, and the zeros between the numerator of Z_y and the denominator of Z_x. For the case n=4 the values in eq. 2.42 - 2.57 are really substituted:

$$Z_x = K_x \frac{(p-\sigma_{32})(\ )}{(p-\sigma_{21})(p-\sigma_{34})(\ )} = K_x \frac{N(p)}{D(p)} \qquad (2.42)$$

$$Z_y = \frac{K_x}{K_d} \frac{(p-\sigma_{12})(\ )}{(p-\sigma_{01})(p-\sigma_{23})(\ )} = \frac{K_x}{K_d} \frac{N(p)}{D(p)} \qquad (2.43)$$

If desired σ_{12} and σ_{21}, σ_{23} and σ_{32} may be reversed. The above described procedure of alternate division should be continued until the last root is left. If the last root is a solution of eq. 2.40 it must be a pole of Z_y (eq. 2.43) otherwise a solution of eq. 2.41 should be a pole of Z_x. Notice that the degrees of the numerator and denominator are different. This is essential for the realization of fig. 2.14a (after the extraction of the load impedance).

The constants are related by:

$$K_d = (1-K)/(1+K) \tag{2.44}$$

The absolute value of K_x is arbitrary and is determined by ranges of available elements. The above described procedure ensures the RC realizability of the network, the conditions for which are:
- The poles and zeros are on the negative real axis
- The poles and zeros are interlaced
- The left most value is a zero, which might be at infinity
- The right most value is a pole

If an alternative division is made also obeying these rules the configuration of fig. 2.14b will be the result (after the extraction of the load impedance). This will be showed later.

Next the load impedance will be extracted, according to fig. 2.15.

The network load capacity is:

$$C_1 \leq K_d/(K_x \pi f_1) \tag{2.46}$$

To find the load conductance evaluate:

$$G_x = -\frac{1}{K_x}\,\frac{\sigma_{32}(\,)(\,)}{\sigma_{21}\sigma_{34}(\,)(\,)} \tag{2.47}$$

Notice that the σ's from eq. 2.42 are taken.

$$G_y = -\frac{K_d}{K_x}\,\frac{\sigma_{12}(\,)(\,)}{\sigma_{01}\sigma_{23}(\,)(\,)} \tag{2.48}$$

Notice that the σ's from eq. 2.43 are taken.

If $G_x < G_y$ then $G_1 = 2G_x$ but if $G_x > G_y$ then $G_1 = 2G_y$.

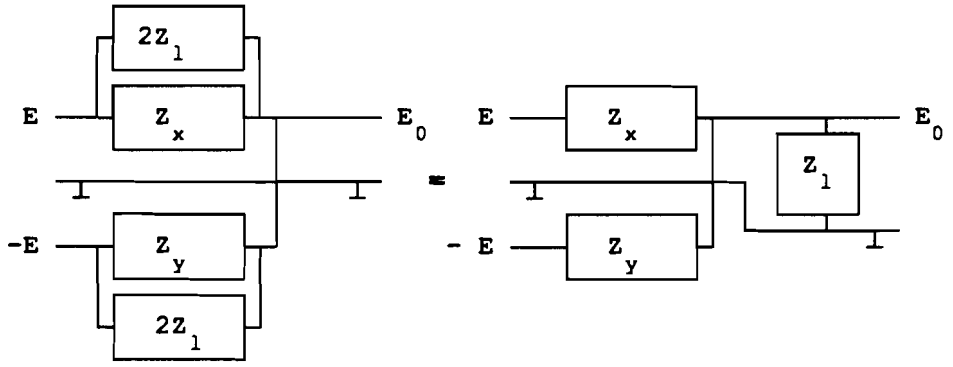


Fig 2.15: Extracting of a load impedance

If:

$$Z_x(p) = K_x\, N(p)/D(p) \qquad (2.49)$$

then:

$$Z_{x1}(p) = K_x \frac{K_x N(p)}{D(p) - \frac{Kx}{2}(G_1 + 2\pi f_1 C_1) N(p)} \qquad (2.50)$$

If the load is extracted the final network can be calculated if Z_{x1} is expanded into partial fraction form:

$$Z_{x1}(p) = R_{x0} + \frac{K_{x1}}{p - p_{x1}} + \frac{K_{x2}}{p - p_{x2}} + \ldots. \qquad (2.51)$$

Element values can be obtained by:

$$C_{xr} = 1/\,(K_{xr} 2\pi f_1) \qquad r = 1, 2, 3, \ldots \text{ etc.} \qquad (2.52)$$

$$R_{xr} = -\,K_{xr}/p_{xr} \qquad r = 1, 2, 3, \ldots \text{ etc.} \qquad (2.53)$$

In eq 2.52-53 f_1 appears because in the foregoing $p = j\Omega = j\omega/\omega_1$ is normalized against ω_1.

Notice that R_0 and p_1 may be zero, in which case R_1 is infinite, i.e. open circuit. K_x may be given a value to get practical available element values. Finally it should be remarked that this procedure for Z_x (eq. 2.49-53) has to be repeated for Z_y. The results obtained with the first type of realization are in full agreement with the results obtained in paragraph 2.4. Paragraph 2.4 covers only $n = 4$, if higher orders ($n > 4$) are needed the above described method should be used.

Second, the above described procedure will be altered to obtain the realization of fig. 2.14b. This realization can be driven by a non ideal source with source impedance. Construct two impedance functions of the form:

$$Z_x = K_x \frac{(p-\sigma_{01})(\ )(\ )}{(p-\sigma_{21})(\ )(\ )} = K_x \frac{N(p)}{D(p)} \tag{2.54}$$

$$Z_y = \frac{K_x}{K_d} \frac{(p-\sigma_{12})(\ )(\ )}{(p-\sigma_{23})(\ )(\ )} = \frac{K_x}{K_d} \frac{N(p)}{D(p)} \tag{2.55}$$

Notice that the σ's should be divided in such a way that the degree of the numerator equals the degree of the denominator. This is elementary for the realization of fig. 2.14b. Next, a load conductance can be extracted from the network. In order to remove the load conductance evaluate:

$$G_x = -\frac{1}{K_x} \frac{\sigma_{21}(\ )(\ )}{\sigma_{01}(\ )(\ )} \tag{2.56}$$

Notice that the σ's from eq. 2.54 are taken.

$$G_y = -\frac{K_x}{K_d} \frac{\sigma_{23}(\ )(\ )}{\sigma_{12}(\ )(\ )} \tag{2.57}$$

Notice that the σ's from eq. 2.54 are taken. If $G_x<G_y$ then $G_1 = 2G_x$ but if $G_x>G_y$ then $G_1=2G_y$.Now extract the load:

$$Z_x(p) = K_x\, N(p)/D(p) \tag{2.58}$$

then:

$$Z_{x1}(p) = K_x \frac{K_x N(p)}{D(p)-\frac{Kx}{2} G_1 N(p)} \tag{2.59}$$

The rest of the procedure is identical to eq. 2.51-53. It has been proven that for the network of fig. 2.14b it is impossible to extract a load conductance and capacitance and at the same time retain realizability. It also has been proven that the eq. 2.47-48 and 2.56-57 provide the largest load conductance that can be extracted while retaining network realizability, as expressed by the rules for the positions of the poles and zeros.

2.8 The practical realization and measurements on the hybrids

It is obvious that the above described procedure involves a lot of tedious computation. Therefore a FORTRAN program has been written to cover the required networks for the image reject mixer. The source listing of this program can be found in appendix 2. The program covers n = 4 for both realizations of fig. 2.14a and 2.14b, and n = 6 for fig. 2.16a. The n = 6 case was used for the LF hybrid and fig. 2.16a shows the resulting network, 2.16b shows the HF hybrid (n = 4).

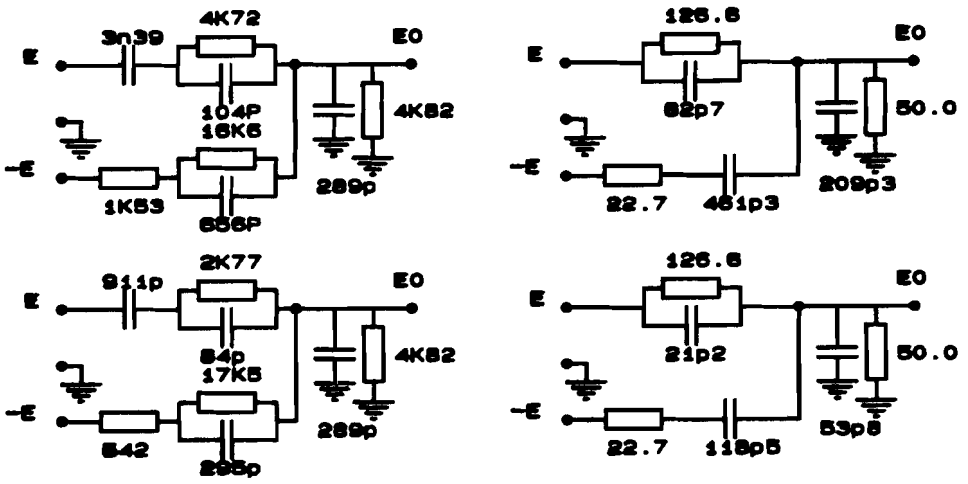


Fig. 2.16: a) LF hybrid, n = 6 (10kHz-1Mhz) b) HF hybrid, n = 4 (10Mhz-90Mhz)

The complete LF hybrid including the driver stage is sketched in fig. 2.17. In this form it has been used to construct the image reject mixer. Fig. 2.18 shows the HF hybrid.

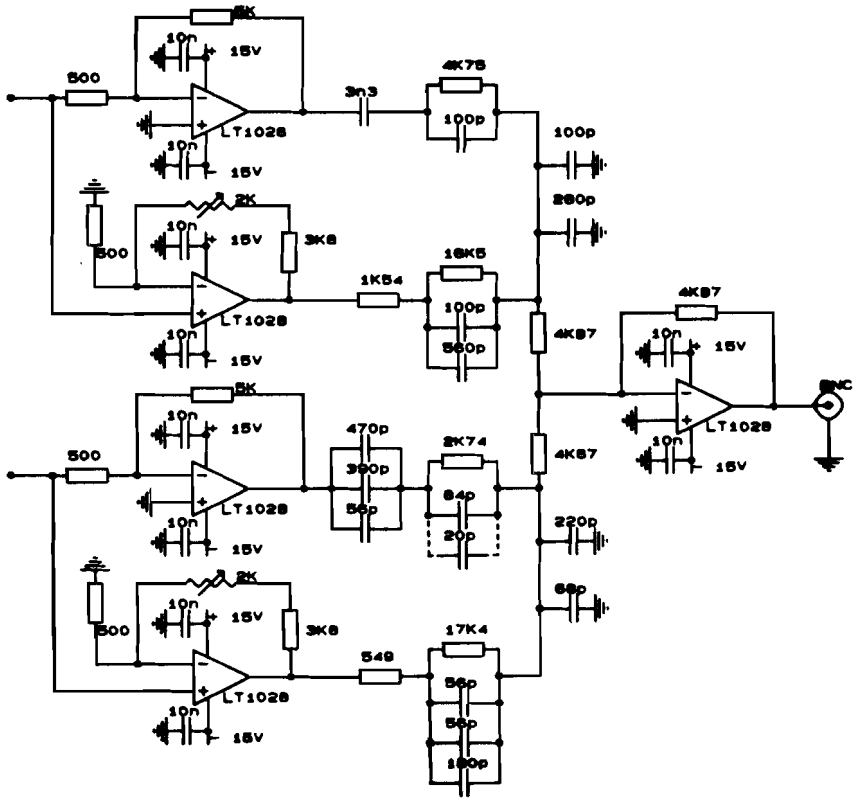


Fig. 2.17: The complete LF hybrid, including driver stages

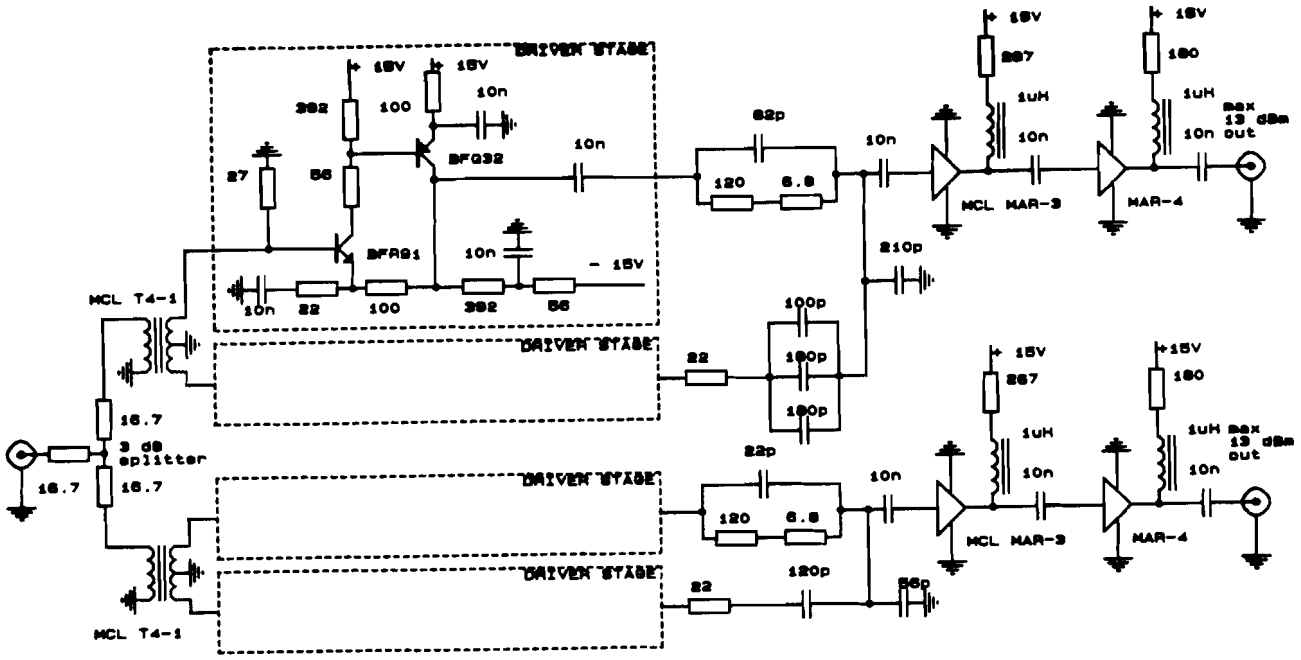


Fig. 2.18: The complete HF hybrid, including driver stages

The following remarks can be made about the realization:
- The RC circuit is relative insensitive to element tolerances. Simulations confirming to this were made and are summarized in table 2.4. This includes both the LF and HF quadrature hybrid.

Table 2.4: Influence of element tolerances on phase and amplitude Simulated on LF hybrid n=6.

Tolerance: R	Tolerance: C	Phase deviation from 90 degree:	Amplitude unbalance in dB:
1%	1%	± 2.6	0.3
1%	5%	± 6.0	1.1
1%	10%	± 7.2	1.9
5%	5%	± 7.2	1.1
5%	10%	± 12.5	2.1
10%	10%	± 12.8	2.4

- The LF circuit can be built by standard components.

- A few parasitic effects can be corrected for.

- The HF circuit is more sensitive to imperfections, such as by parasites (mostly parasitic inductances caused by leads) that have more influence at higher frequencies. At these high frequencies capacitors should be used that have short leads and remain their function also at these frequencies. Therefore SMD's (Surface Mounted Devices) are recommended.

- The HF hybrid is also built with the first realization described in paragraph 2.7 (fig. 2.14a). For HF circuits with 50 Ω source impedance the second realization looks more promising but the parasitic inductance of the driver stages and the necessary transformer make it very difficult to build.

- The performance of the HF circuit is slightly degraded by the inductive output impedance of the driver stages. This can not be avoided because the inductive behavior is fundamental and caused by the group delay of the transistor. Fortunately the caused phase shift can be corrected by an extra delay in the phase shift network (e.g. piece of wire). The amplitude unbalance is of less importance because the device is used to drive a mixer. A double balanced diode mixer is almost insensitive to a few dB more LO power.

- Parasitic capacitances of a resistor can be compensated for by the accompaning capacitor and parasitic resistance of the capacitor can be taken into account by the resistor.

- If the inductive behavior of the driver stage is known it can be partly corrected for by an optimization program.

- To avoid a extra inductive behavior of the driver the transformer in the HF hybrid should be placed in front of the driver stages.

The measurements that were carried out can be found in appendix 3. The following characteristics were measured:
- absolute gain and phase LF hybrid
- phase difference and amplitude difference LF hybrid
- frequency span that can be allowed for a phase deviation $\leq 5^{\circ}$ from the straight line approximation
- absolute gain and phase HF hybrid
- phase difference and amplitude difference HF hybrid
- absolute gain and phase HF driver stage
- output impedance of the HF driver stage
- spurious at the output of the HF hybrid

From these measurements on the RC quadrature hybrids it can be concluded that:
- The LF hybrid can be built without difficulties, only a slight correction is necessary for non idealities of the op amp stage.

- The inherent insertion loss of the network (for n = 4: 7dB and n = 6: 14dB) can be canceled by amplification.

- The op amp type (LT 1028) in the LF hybrid realization should be replaced because (e.g. AD 9610):
  a) The disappointing noise performance
  b) It can not deliver the full output power required to drive the network. The op amps saturate before full output level is reached.

- The HF hybrid should be built by SMD components. A few parasitic effects can be corrected for.

- The driver stages of the HF hybrid can be improved to get less spurious, the achieved spurious level however is sufficiently low.

- The achieved 90° phase shift tolerances and amplitude unbalance are summarized in table 2.5.

Table 2.5: Achieved performance of the RC quadrature hybrid

LF hybrid (10kHz-1Mhz):	Result:
deviation from 90 degree:	< ± 2 degree
amplitude difference A and B:	< 0.4 dB
amplitude imbalance:	< 0.5 dB
frequency span with $\Delta\phi \leq \pm 5^{\circ}$:	10 - 110 kHz 310 kHz - 1 MHz
HF hybrid (10MHz-90MHz):	
deviation from 90 degree:	< ± 1.2 degree
amplitude difference A and B:	< 1.2 dB
amplitude imbalance:	< 2.5 dB

2.9 Simulations and measurements on the image reject mixer

Using the LF and HF hybrid the complete image reject mixer can be assembled. The elaborated scheme is sketched in fig. 2.19. In order to compare the results obtained with the practical realization of fig. 2.19 and the theoretically expected behavior, some simulations were carried out. The simulations consider the expected image rejections with the used networks, the expected group delay and the phase and amplitude behavior of the networks. The results can be found in appendix 4.

In the built image reject mixer it can been seen that a filter has been inserted between the mixer and the LF hybrid (fig. 2.19). This LP filter is used to terminate the mixer properly at high frequencies (>10 Mhz). Two alternatives are given in appendix 5 with their measured behavior. Both satisfy.

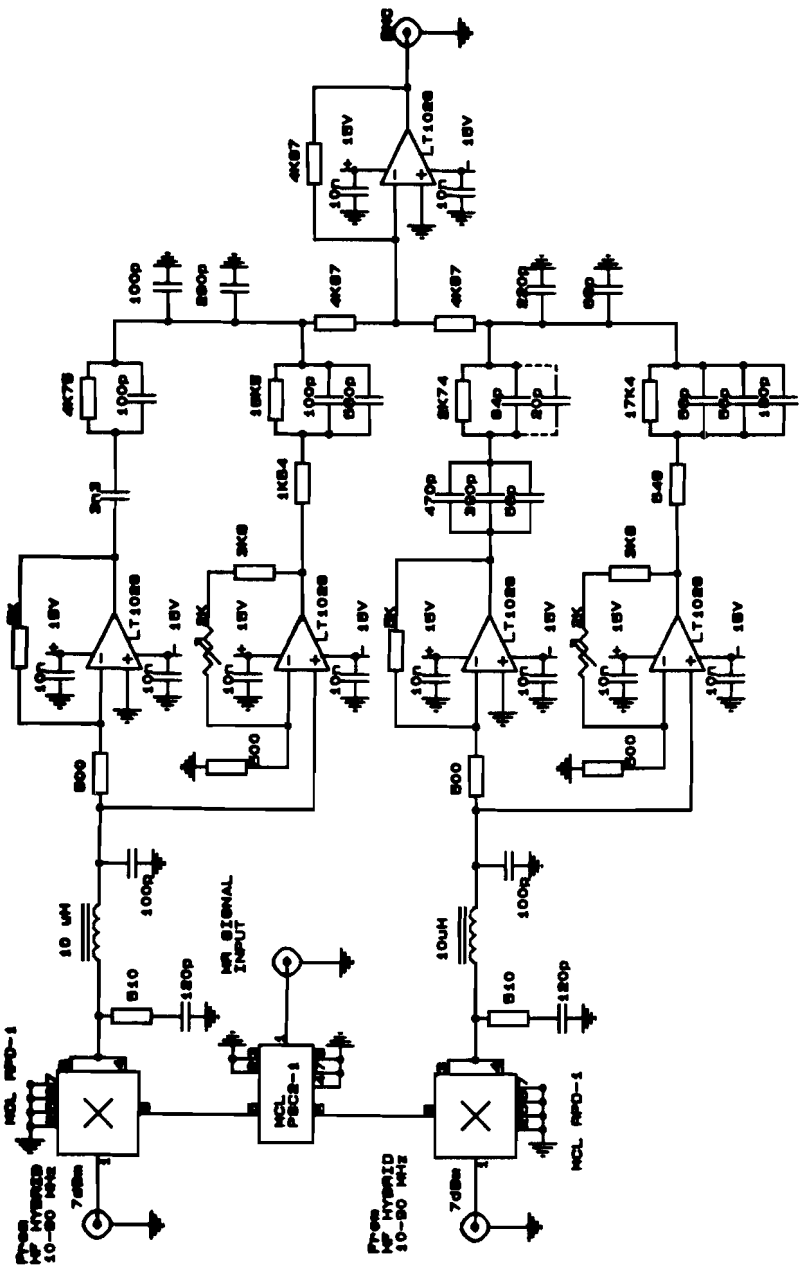


Fig. 2.19: Build image reject mixer

Finally measurements on the image reject mixer were carried out. In table 2.6 the measurements and their results are summarized. For measurement plots see appendix 6.

Table 2.6: Measurement summary of the image reject mixer

HF frequency:	Image rejection:	LF frequency span:
f = 10 MHz	> 22 dB	10 kHz - 1 MHz
21 MHz	> 30 dB	10 kHz - 1 MHz
50 MHz	> 25 dB	10 kHz - 1 MHz
64 MHz	> 30 dB	10 kHz - 1 MHz
90 MHz	> 21 dB	10 kHz - 1 MHz
Group delay 10 kHz - 1MHz:		16 - 0.21 µsec
Third order inter. mod. prod. at input level: - 10 dBm		
after mixer:		60 dB below carrier
at output:		60 dB below carrier
Noise figure		23.3 dB

2.10 Conclusions and recommendations on the image reject mixer

The built image reject mixer meets its specifications very well. The desired image rejection is amply realized over the desired bandwidth. The measured spurious-free dynamic range agrees with the used level 7 mixer, for stronger requirements higher level mixers should be used (spurious-free dynamic range is the range between output peak and highest spurious level).

If the achieved phase linearity doesn't satisfy, a phase correction in the anti-alaising LP filter at the ADC should be added. Butterworth LP filters have the ability to correct a phase.

If the LF and HF hybrid are considered in detail, it can be concluded that the LF hybrid can be built easily with standard components. If desired even stronger requirements with the above described design can be realized . This can done in two ways, first by using higher order networks and secondly by using new technology (e.g. SMD components).

The only change that must be recommended is to replace the LT 1028 (e.g. AD 9610) because of two reasons. The first reason is to improve the noise behavior. As shown in appendix 7 the noise behavior of the LT 1028 degrades at 200 kHz. Examination of the data sheets of the LT 1028 show that this behavior is in agreement with its specification. In appendix 7 also a few suggestions are denoted to build low noise op amps. The second reason is that at higher output levels the LT 1028 saturates and can not deliver full output level to the RC network. This causes a degraded dynamic range of the LT 1028.

The HF hybrid can be realized properly but good components should be used (SMD components). The main problem are the inductive parasites present in the driver stages. These are the limiting factors if stronger requirements (such as larger bandwidth) should be met. The spurious free dynamic range of the HF hybrid's driver stages is satisfactory but can be improved.

Finally it can be remarked that the described image reject mixer meets the challenge posed by the device made by Merrimac. The major advantages are the improved performance and reduced costs.

3. THE DIGITAL SYNTHESIZER

First, to get an impression of its architecture, the digital synthesizer is described. This is done to understand the underlying arguments of this realization. This is followed by calculations to verify the design against the demands asked by the MR application (spurious level, frequency resolution etc.). Also simulations are made to verify the design and predict the practical behavior. Finally the measurement results of the built synthesizer are presented.

3.1 The architecture of the synthesizer

As suggested by its name the digital synthesizer generates its output waveforms by converting digital numbers representing sample points of the desired waveform. These digital numbers are stored in a read only memory (ROM) and therefore the desired output waveform can be constructed by reading through the ROM. In this case the desired output waveforms are a sine and a cosine. The analog representation is made by a digital to analog converter (DAC).

Fig. 3.1 sketches the most elementary functions of the synthesizer. The ROM with sine table and the DAC are obvious. An adder with the output fed back to one of the inputs is used to scan through the ROM addresses. In this way the output frequency of the synthesizer can be adjusted by setting another increment in the adder. This increment sets the scan step size through the ROM. Therefore a larger increments implies a higher frequency.

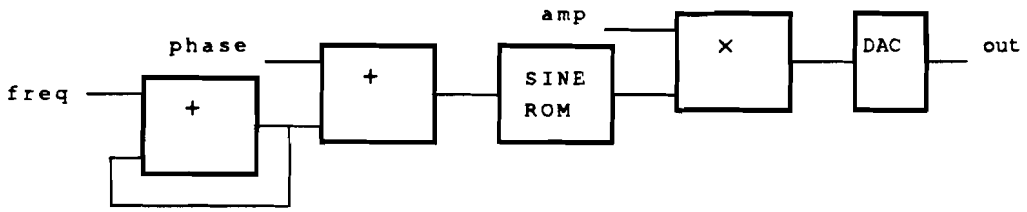


Fig 3.1: Elementary diagram of the synthesizer

The next logical extension is the addition of phase modulation by another adder. In this way a phase modulation of a certain angle is realized by jumping over cq. skipping a certain number of addresses in the ROM. The last extension in the elementary diagram is the amplitude modulation. This can easily be realized by multiplication of the output with the desired amplitude. All these extensions are implemented in such a way that a phase continuous operation of the synthesizer is guaranteed.

3.2 Further extensions in the architecture of the synthesizer

As stated in the former paragraph the most elementary functions of the synthesizer are the frequency adder, the phase adder and the amplitude multiplier. So far just an ordinary synthesizer is described [Lit. 29]. The application in the STRIP system asks for two 90^0 shifted output signals, because the synthesizer is used to drive a single side band modulator. This can be realized in this

synthesizer architecture by adding a second channel. The ROM table in the second channel should contain a cosine function in comparison to a sine function in the first channel.

The synthesizer should also be able to reverse the output sine instantaneously. The effect of this is that the derivative of the phase of the complex output signal becomes negative. Since the momentary frequency of the digital synthesizer equals this derivative, one can consider the frequency to be negative. This means that the desired output signal band of the SSB shifts from the upper side band part to the lower side band part. The sign bit at the frequency input determines whether the sine output will lead or lag the cosine output by 90°.

The synthesizer is used to drive a single side band modulator that operates on the phase shift method. These modulators usually suffer from three problems:
- Carrier leak through, due to offset problems in the mixers and driving stages
- Insufficient side band suppression caused by phase shift problems
- Insufficient side band suppression caused by inequality of the amplitude in the driving signals.

For a further explanation of an SSB modulator the reader is referred to chapter 4. Also calculations illustrating these problems are given. In order to solve these problems one could design a sophisticated analog SSB modulator. In the STRIP system a different approach is chosen. The problems discussed above could easily be solved if the driving synthesizer is able to vary the following parameters corresponding to the 3 problems mentioned above:
- offset at every output of the synthesizer.
- phase difference between the sine and cosine output.
- amplitude difference between the sine and cosine output.

Channel 1 in the synthesizer is used to make a phase correction by introducing an extra adder to be able to select a certain phase offset in the sine table with regard to the cosine table. Similarly the signal of channel 1 can be multiplied with a certain factor, in channel 2 no multiplication is introduced so the amplitude of channel 1 can be changed with regard to channel 2.

Finally, the offset is introduced by adding a small number to the output before D/A conversion is accomplished. With these extensions on the diagram of fig. 3.1 the overview of the realized synthesizer is sketched in fig 3.2. In this way it is possible to correct for errors in the analog part by changing parameters in the digital synthesizer.

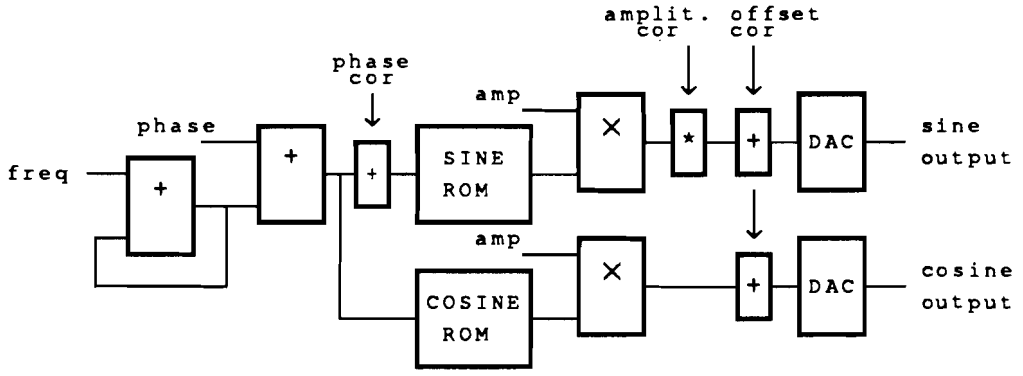


Fig. 3.2: The digital synthesizer with correction factors

In fig 3.3 an overview is given of the realized synthesizer. A few remarks should be made on this design:

- The logic before the sine and cosine table uses unsigned magnitude numbers to be able to address the ROM in a straight forward manner and to use a simple rounding circuit.

- The logic after the sine and cosine table uses 2 complement fractions because of the multipliers and the amplitude modulation. In this way a sine value between -1 and +1 can be multiplied with an amplitude ≤ 1.

- The design with synchronous logic is made in such a way that the same approach and architecture can be used to implement the synthesizer in a Application Specific Integrated Circuit (ASIC).

- In order to save pipeline registers the correction factors are set by DIP-switches. They occur directly in front of the correction adders and multipliers. In a final realization the correction factors should be controlled by the computer in order to calibrate the transmitter path of the STRIP system. Then of course the pipelines (register stages) are necessary if a consequent, completely synchronous design is desired.

- The DACs use an unsigned binary code in which 00..00 represents $-V_{fullscale}$, 10..00 = 0V
  and 11..11 = $V_{fullscale}$-1 LSB.
  Therefore the 2 complement number should be converted into this code by inverting the MSB.

- Another rounding circuit at the output of the ROM could be added, this has not been built in this design in order to save parts. Notice that the rounding after the ROM concerns 2 complement numbers. The rounding minimizes the momentary phase error that occurs if the 23 bits of the phase adder are truncated to 13 bits. The rounding in front of the ROM looks to the 14th bit in order to decide to add 1 to the remaining 13 bits. Rounding only makes sense if more as 1 bit is rounded. The average error in truncating more as 1 bit is always positive. The rounding shifts this error to values around 0.

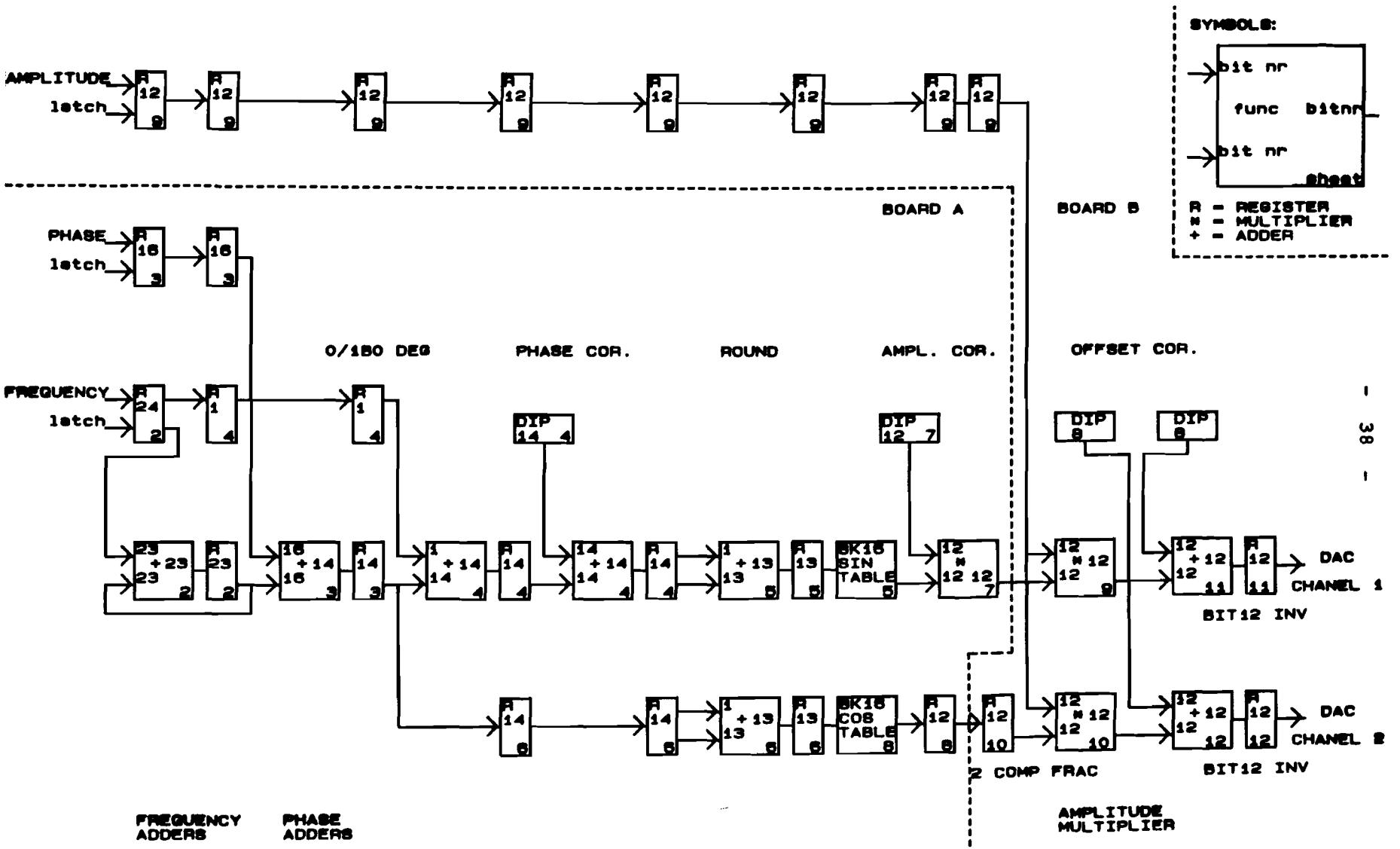


Fig. 3.3: Overview of the digital synthesizer

- Both positive and negative phase corrections should be possible. Therefore an adder is used that can address every location in the ROM. In a final design only small phase corrections are necessary so not all of the used 14 bits are useful.
- The offset correction in this design is not yet complete. At the moment only positive corrections are possible.

3.3 Design considerations and requirements

In fig. 3.3 several numbers are visible representing the width of the data flow (in bits) between several parts. These widths are chosen with the demands of the STRIP system in mind. The following calculations are made to verify the requirements versus the design.

In table 3.1 these demands are summarized. First the desired frequency resolution is considered. A resolution of about 1 Hz is asked for by spectroscopy applications.

Table 3.1: demands on the synthesizer

Item:	Specification:
Frequency range	0 - 1 MHz
Frequency resolution	$\simeq$ 1 Hz
Spurious level	< - 60 dB b.c.

The required number of bits of the frequency adder to achieve a frequency resolution of about 1 Hz can be calculated as follows:

Suppose the desired output signal is:
$S(t) = \cos(\omega t)$ with $\omega = 2\pi f$ (3.1)

Then is the phase as function of time:
$\phi(t) = \omega t$ and $d\phi_{resol}(t)/dt = \omega_{resol} = 2\pi f_{resol}$ (3.2)

If the full period of the sine wave is stored in a table of N positions, the minimal phase step is:
$d\phi_{resol} = 2\pi/N$ (3.3)

From eq. 3.2 and 3.3 with dt = 100 nsec (the clock frequency of the synthesizer = 10 MHz = 1/dt) follows:
$d\phi_{resol} = 2\pi f_{resol} dt = 2\pi/N$ (3.4)
$N = 1/f_{resol} dt = 10^7$ (3.5)

The minimal number of bits k follows from:
$2^k > N$ (3.6)
$k > \log N/\log 2 = 23{,}25$ (3.7)

So a frequency adder of 23 bits is needed to achieve the desired frequency resolution. One could also expect a table of 2^{23} = 8 Mbyte positions to store the sine wave. It is obvious, however, that this is too expensive. In order to reduce the number of locations in the sine ROM, a trick is used. The frequency adder is

implemented in full 23 bits but only 13 bits are used to address the ROM. So at low synthesizer frequencies the same location in the ROM will be read several times because only the MSBs from the frequency adder are used to address the ROM.

As a result of the limited ROM size two types of discretization errors occur. The first is caused by the rounding of the limited ROM table length and the second by the rounding of the limited ROM table with (and thus the limited number of bits at the DAC).

In the remainder of this paragraph, first the effects of the limited ROM table length will be treated and secondly the effect of the limited number of DAC bits.

The limited ROM table length has a rounding result on the phase of the output sine and cosine. If the scanning through the ROM is considered in detail two effects can be discovered:

- The ROM will be read in a non periodic way for frequencies of which the derivative of the phase doesn't equal an exact multiple of the scan time interval (phase resolution step) of the sine table. Consider that the first time the ROM is read from location 1 and the period of the desired output sine doesn't fit on an exact multiple of the period of the ROM. Then the ROM will be read with a certain changing offset from location 1 every new period. In practice the desired phase will be averaged from several scans through the ROM. It is obvious that phase errors occur due to the process described.

- The phase resolution step of the ROM doesn't equal the desired phase step of the output frequency. A rounding will be performed towards to nearest resolution step that can be achieved with the used ROM size. In practice a not exact phase resolution step, and therefore a certain phase modulation will be the result.

So also the effect of a non periodic scanning through the ROM results in a small phase modulation at the output. This effect has two advantages. First the required frequency resolution can be achieved without an extremely large ROM because the desired resolution occurs as an averaging of several scans through the ROM. Second the averaging has a certain spreading and smoothing effect for the unwanted frequency peaks caused by a limited number of bits at the DAC and ROM table. This is confirmed by the simulations described in the next paragraph.

Thus the limited number of bits for the length of the ROM table causes a small amount of phase modulation. The resulting spurious level caused by the phase modulation can be estimated and from these calculations follows a minimal ROM table size. Consider the ideal output:

$$S(t) = A_c \cos(\omega t) \qquad (3.8)$$

The real output due to rounding errors caused by the ROM table will be:

$$S_{real}(kt_d) = A_c \cos(\omega k t_d + \theta(k)) \quad \text{for } k = 0,1,2... \tag{3.9}$$

The phase modulation in eq. 3.10 can be represented by:

$$\theta(k) = \sum_{n=1}^{N} \frac{\Delta\omega}{\omega_m} \sin \omega_m k t_d \tag{3.10}$$

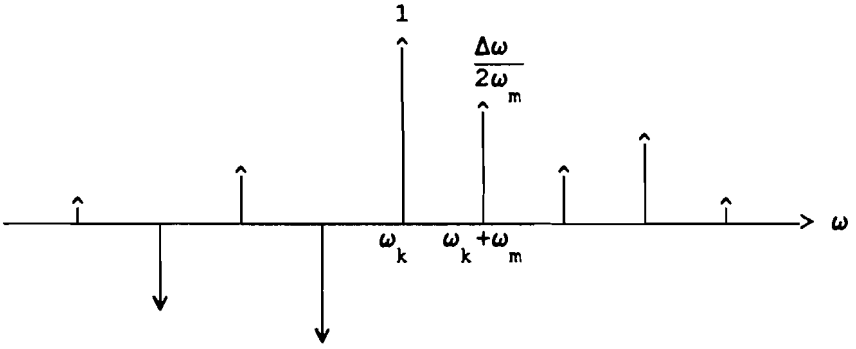


Fig 3.3: Frequency spectrum phase modulated signal

Fig. 3.3 shows that the highest spurious level caused by the phase modulation lies $\Delta\omega / 2\omega_m$ down in comparison with the fundamental tone. The maximum error that can be made by rounding between two locations in the ROM is half a resolution step:

$$1/2\ \theta_{resol} = \frac{1}{2} \frac{2\pi}{2^q} \tag{3.11}$$

The modulation index $\beta = \Delta\theta = \frac{\Delta\omega}{\omega_m}$ is:

$$\beta = \Delta\theta = \frac{\Delta\omega}{\omega_m} = 2\pi / 2^q \tag{3.12}$$

From eq. 3.10, eq. 3.11 and eq. 3.12 follows the level from the first side band caused by the phase modulation:

$$\Delta\omega / 2\omega_m \text{ (dB)} = 20\ ^{10}\log (\pi / 2^q) > 60 \text{ dB b.c.} \tag{3.13}$$

So $q \geq 9$ bits [Lit. 30].

In this case also a certain margin is necessary so in the design 13 bits are chosen to allow a reasonable ROM size of 8K.

In the foregoing the ROM table width was assumed to be unlimited. If only a limited number of DAC bits is used, this will have a rounding result on the output. The rounding results in a certain amplitude error and therefore amplitude modulation of the output.The maximum error that can be made depends on the number of bits and is half a quantization step:

$$\varepsilon_{max} = \frac{1}{2} \frac{1}{2^q} \tag{3.14}$$

The occurring amplitude modulation can be expressed by:

$$\varepsilon(t) = 0.5\ \varepsilon_{max} \cos(\omega + \omega_{\varepsilon}) + 0.5\ \varepsilon_{max} \cos(\omega - \omega_{\varepsilon}) \qquad (3.15)$$

with ω = the actual frequency of the synthesizer
ω_{ε} = the frequency of the rounding occurrence

It is obvious that to achieve a certain spurious level the rounding error should be limited by a sufficient number of DAC bits. To achieve a spurious level of - 60dB b.c. the number of DAC bits should be:

$$20\ ^{10}\log\ (\ 0.5\ \frac{1}{2}\ \frac{1}{2^{q}}\ ) > 60\ \text{dB b.c.} \qquad (3.16)$$

So q > 8 bits.

It should be noticed that this is a worst case approach. In practice each bit will have its own rounding frequency and thus amplitude modulation and the resulting spurious from this modulation will be lower.

It can be concluded that at least 8 bits at the DAC and thus at the ROM table are necessary to achieve a spurious level lower then - 60 dBc. To have a certain margin 12 bits are taken.

3.4 Simulations on the spurious level of the synthesizer

To verify the effects of the limited number of bits at the DAC and the limited table size of the ROM, some simulations were made. A FORTRAN program was written to emulate the behavior of the synthesizer. The program was written in such a way that the effect of the phase modulation caused by the limited ROM size was imitated. The program has one drawback due to calculation limitations on the VAX and therefore only calculates the frequency spectrum from 0 - 10 kHz. The resulting spectrum for synthesizer frequencies > 10 kHz is undersampled and thus aliased into the area from 0 - 10 kHz. Only the maximum spurious level is interesting so the problem discussed above is of minor importance.

The results are summarized in table 3.2 and the frequency spectrum plots can be found in appendix 8. The results agree with the calculations in paragraph 3.3 of quantization noise and phase modulation spurious.

Table 3.2: Simulation results

Length sine table:	Width sine table:	Spurious level:
512 bits	12 bits	> 58 dB
1024 bits	12 bits	> 64 dB
2048 bits	12 bits	> 69 dB
8192 bits	12 bits	> 78 dB
8192 bits	8 bits	> 72 dB

3.5 The realized synthesizer and its measured performance

In appendix 9 the detailed circuit diagrams of the realized synthesizer are drawn. The synthesizer was built upon two double size eurocards with synchronous TTL-LS and FAST logic. Also high speed 16 bit multipliers and 8k8 EPROMs are used. The digital to analog conversion is done by two (35 nsec settling time 12 bit) video DAC's. The design is straight forward and simple because of the synchronous logic. Attention should be paid to the cable interconnecting the two boards, it should be as short as possible because of the relatively high clock speed (10 MHz).

Further it is very important to minimize the glitch impulses generated by the DACs. Glitch impulses are caused by not exactly simultaneously switching DAC bits. In many high speed DAC applications, glitch performance is a critical specification. In the DAC data sheets several remedies are suggested. Most of them were implemented e.g. switching threshold of the DACs adjusted, driving the DAC by high speed CMOS 74HC374 octal latches, minimizing and equal length of input lines to DAC and finally fine tuning of the timing of the 3 MSBs of the DAC by an RC combination (deskewing). The most promising remedy the application of a sample and hold as deglitcher wasn't tested for the relatively high costs.

Several measurements were performed to compare the behavior of the built synthesizer to the results predicted by the simulation. The following tests were done. First an experiment to verify the ROM table length against the simulations, secondly the effects of the table width and finally the effects of the glitch impulses. The frequency plots can be found in appendix 10. The results agree with the simulation, compare for instance the cases with ROM length = 8192 bits and width = 6 bits. The glitch impulses however have an overruling effect on larger bit widths. This makes the results a bit harder to compare.

The glitches cause unwanted frequency peaks at every odd and even harmonic of the fundamental tone. This is exactly the switch frequency of resp. the second, third and so on bits of the DAC. The glitch impulses have a certain minimum value of $\simeq$ 150 - 350 pV sec. At high synthesizer frequencies the glitch energy has relatively more influence on the spectrum because the ratio of constant glitch period versus changing fundamental frequency period is lower.

One extra remark should also be made: at higher output frequencies (1.2 MHz) of the synthesizer the glitch impulses tend to disturb the nearby spectrum of the output by unwanted frequency peaks. It was found that these peaks disappear if the ROM table is read out the same way for every period of the output sine. By adjusting the time constant in the deskewing circuit these peaks get influenced. These spurious are about 64 dB down (appendix 10).

A final remark on the glitches is that they are bit pattern dependent. For a certain repeating bit pattern at the input of the DAC glitch energy can be minimized but at a different bit pattern

the glitch energy may be worse. A certain optimum should be found.

The required 60 dB b.c. spurious level could be reached after carefully minimizing the glitch impulses by the adjustments mentioned above. The results can be seen in appendix 10.

3.6 Conclusions and recommendations

The built configuration meets the specification of achieving a spurious level of 60 dB b.c., however after the glitch impulses have been carefully minimized. If stronger requirements should be met or if the fine tuning of the timing of the MSBs are inconvenient a deglitcher (sample and hold) should be used. Fig. 3.4 shows the expected improvement in spectral purity if a deglitcher is used [Lit.31].

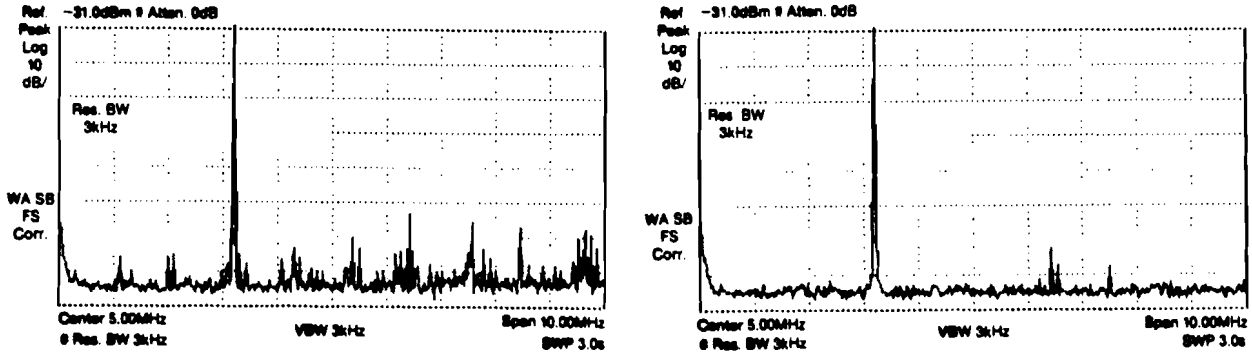


Fig. 3.4: Expected spectral purity improvement with deglitcher

The necessary table size and DAC width are 1024 and 8 bits. To have a certain margin a table size of 2048 by 10 bits also is convenient. Larger ROM sizes bring no advantages because the glitch impulses are the limiting factor on spectral purity. A trick that can be used to decrease the ROM size is to store only a quarter period of the sine (and cosine) in the ROM. Attention should be paid to store the sine (and cosine) in such a way that periodicity is guaranteed (not read a zero or 1 twice if the next quadrant is scanned). Extra logic should be added to create a sign and a quadrant bit.

4. THE SINGLE SIDE BAND MODULATOR

This chapter describes the single side band (SSB) modulator that is used to mix the output of the digital synthesizer to the MR frequency. These kind of modulators usually suffer from two problems; carrier leak through and imperfect side band suppression. These imperfections are treated mathematically to illustrate possible remedies. In this way the correction factors in the synthesizer will be made clear. Also measurement results on the realized SSB modulator are presented.

4.1 Main principle

With the set up as sketched in fig. 4.1 it is possible to create the required output signal.

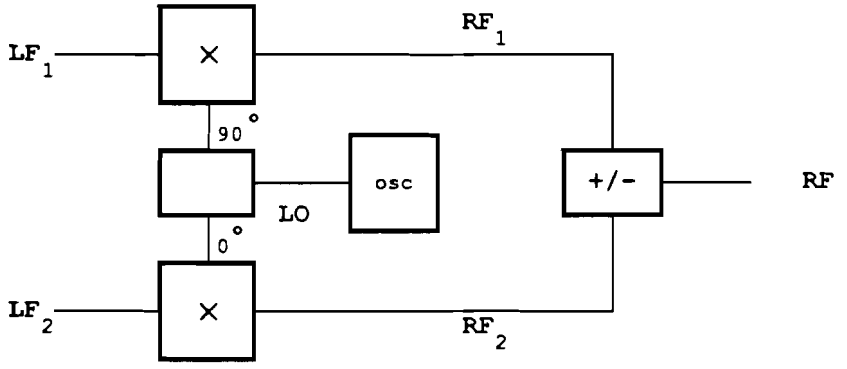


Fig. 4.1: Block diagram of the SSB modulator

Suppose the following signal are taken as input signals:

$$LF_1 = A(t) \cos \omega_m t \tag{4.1}$$

$$LF_2 = A(t) \sin \omega_m t \tag{4.2}$$

$$LO = \cos \omega_r t \tag{4.3}$$

Mixture leads to:

$$RF_1 = A(t)\cos \omega_m t \sin \omega_r t$$

$$= \frac{A(t)}{2} [\sin(\omega_r - \omega_m)t + \sin(\omega_r + \omega_m)t] \tag{4.4}$$

$$RF_2 = A(t)\sin \omega_m t \cos \omega_r t$$

$$= \frac{A(t)}{2} [-\sin(\omega_r - \omega_m) + \sin(\omega_r + \omega_m)t] \tag{4.5}$$

Addition shows the upper side band as result:

$$RF = RF_1 + RF_2 = A(t) \sin (\omega_r + \omega_m)t \tag{4.6}$$

Subtraction shows the lower side band:

$$RF = RF_1 - RF_2 = A(t) \sin (\omega_r - \omega_m)t \tag{4.7}$$

4.2 The effect of imperfections in the design and requirements

The calculation described above assumes all signals to be ideal. If amplitude or phase errors appear the output signal is

distorted. This distortion has two effects in practice. First the carrier leaks through and secondly the other side band is not so well suppressed. This can be made clear by the following calculation.

Suppose that with no LF modulation signal the LO signal leaks through the mixers (by parasitic coupling a 90° shifted signal leaks through) and that the LO signals at the mixers exhibit a phase unbalance ϕ. Furthermore an inequality in the conversion loss of the mixers is assumed. Taking the same ideal modulating signals we find:

$$LF_1 = A(t) \cos \omega_m t \tag{4.8}$$

$$LF_2 = A(t) \sin \omega_m t \tag{4.9}$$

$$LO = \cos \omega_r t \tag{4.10}$$

Mixture leads to:

$$\begin{aligned} RF_1 &= A(t)(1+a) \cos \omega_m t \sin(\omega_r t+\phi) + b \cos(\omega_r t+\phi) \\ &= \frac{A(t)}{2} (1+a) \left[\sin\{(\omega_r-\omega_m)t+\phi\} + \sin\{(\omega_r+\omega_m)t+\phi\} \right] \\ &\quad + b \cos(\omega_r t+\phi) \end{aligned} \tag{4.11}$$

$$\begin{aligned} RF_2 &= A(t) \sin \omega_m t \cos \omega_r t + c \sin \omega_r t \\ &= \frac{A(t)}{2} \left[-\sin(\omega_r-\omega_m) + \sin(\omega_r+\omega_m)t\right] + c \sin \omega_r t \end{aligned} \tag{4.12}$$

In this formula is:

a = amplitude unbalance in the upper channel (RF_1) with regard to the lower channel (RF_2)

b = offset in the mixer of the upper channel (RF_1)

c = offset in the mixer of the lower channel (RF_2)

ϕ = phase error in upper channel (RF_1)

the leak through from the modulating signals neglected.

For small phase errors ($\sin\phi \approx \phi$) the lower side band due to errors will be:

$$\begin{aligned} RF_1-RF_2 = &\frac{A(t)}{2} (\cos\phi(1+a)+1) \left[\sin(\omega_r-\omega_m)t\right] + \\ &\frac{A(t)}{2} (\cos\phi(1+a)-1) \left[\sin(\omega_r+\omega_m)t\right] + \\ &\frac{A(t)}{2} \phi(1+a) \left[\cos(\omega_r-\omega_m)t + \cos(\omega_r+\omega_m)t \right] + \\ &b \cos\phi \cos \omega_r t - b \sin\phi \sin \omega_r t - c \sin \omega_r t \end{aligned} \tag{4.13}$$

It can be shown that the result for the upper side band is analogous. From this is clear that a phase error results in worse side band suppression, amplitude unbalance also and offsets cause carrier leakage.

In the following calculation is shown that the synthesizer with correction factors is capable of correcting the imperfections

mentioned above:

$$LF_1 = A(t)(1+a')\cos(\omega_m t + \phi') + b' \tag{4.14}$$

$$LF_2 = A(t)\sin\omega_m t + c' \tag{4.15}$$

$$LO = \cos\omega_r t \tag{4.16}$$

Mixture, assuming the same imperfections in the mixers, leads to:

$$\begin{aligned} RF_1 &= A(t)(1+a)(1+a')\cos(\omega_m t+\phi')\sin(\omega_r t+\phi) + \\ &\quad b\cos(\omega_r t+\phi)+b'\sin(\omega_r t+\phi) \\ &= \frac{A(t)}{2}(1+a+a'+aa')[\sin\{(\omega_r+\omega_m)t+\phi+\phi'\} + \sin\{(\omega_r-\omega_m)t+\phi-\phi'\}] \\ &\quad + b[\cos\omega_r t\cos\phi - \sin\omega_r t\sin\phi] \\ &\quad + b'[\sin\omega_r t\cos\phi + \cos\omega_r t\sin\phi] \end{aligned} \tag{4.17}$$

$$\begin{aligned} RF_2 &= A(t)\sin\omega_m t\cos\omega_r t + c\sin\omega_r t + c'\cos\omega_r t \\ &= \frac{A(t)}{2}[-\sin(\omega_r-\omega_m) + \sin(\omega_r+\omega_m)t] \\ &\quad + c\sin\omega_r t + c'\cos\omega_r t \end{aligned} \tag{4.18}$$

In this formula is:

a' = amplitude correction in the upper channel (RF_1) with regard to the lower channel (RF_2)

b' = offset correction in the upper channel (RF_1)

c' = offset correction in the lower channel (RF_2)

ϕ' = phase correction in upper channel (RF_1)

also the leak through from the modulating signals neglected.

So the lower side band will be:

$$\begin{aligned} RF_1 - RF_2 &= \frac{A(t)}{2}(1+a+a'+aa'+1)[\sin(\omega_r-\omega_m)t] + \\ &\quad \frac{A(t)}{2}(1+a+a'+aa'-1)[\sin(\omega_r+\omega_m)t] + \\ &\quad (b\cos\phi + b'\sin\phi' - c')\cos\omega_r t + \\ &\quad (b'\cos\phi - b\sin\phi - c)\sin\omega_r t \end{aligned} \tag{4.19}$$

So the correction factors in the synthesizer are capable to cancel the errors in the SSB modulator ($a=-a'$, $b=c'$ and $c=b'$). The case for the upper side band is analogous.

Thus in a practical realization there are three specifications to be met. First carrier suppression by means of low offset voltages. Secondly side band suppression by means of negligible phase errors and thirdly side band suppression as a result of equality in amplitude of both LF signals.

The requirements on the carrier and side band suppression can be

roughly estimated. Imperfect suppression results in frequency components that excite areas of the body other then the desired slice. The signals from other slices cause artifacts in the desired slice. It can be stated that a 1% disturbance from other slices is allowable.

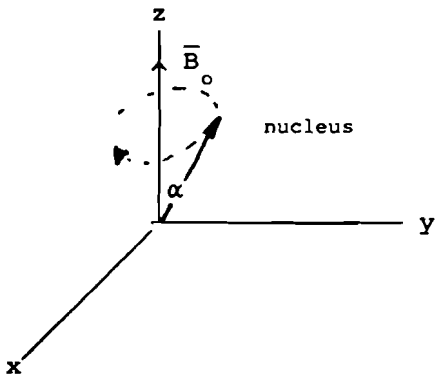


Fig 4.2: Precession of a nucleus with Larmor frequency ω_λ in the static field B_0

Consider a 90° pulse operating on the magnetization M of the desired slice (fig. 4.2). This pulse excites the spins in the desired layer and produces a transversal magnetization M_{tr}. Then 1% signal from other layers is allowed. The excitation angle in these other layers then is α with:

$$\sin \alpha \leq 0.01 \qquad (4.20)$$

$$\text{and thus } \alpha \leq 0.573^\circ \qquad (4.21)$$

$$\text{required suppression} \geq 20\ {}^{10}\log\left(\frac{90}{0.573}\right) = 43.9 \text{ dB} \qquad (4.22)$$

So if only one disturbance peak is present it should be 44 dB down. If more disturbance peaks are involved each peak should be even more down.

4.3 Design of the modulator

The circuit diagram of the realized modulator is sketched in fig. 4.3. The design is straightforward and simple, because in the STRIP system the imperfections of the modulator are corrected in the digital synthesizer. Passive mixers are chosen for their reliability. The used level 7 mixers offer enough dynamic range for this application. The op amps are used to adapt the output level of the DACs from the synthesizer to the level suitable for the mixers. Also an offset correction is added. The synthesizer was not yet capable of generating negative offsets, in this way the coarse tuning of the carrier and suppression can be made with the offset trimmers. The fine tuning can be done with DIP switches on the synthesizer (fig. 3.3). In a definite realization of the synthesizer this is not necessary of course.

The 90° hybrid is the device described in chapter 2 (fig. 2.16), for an explanation and recommendation on this hybrid the reader is referred to paragraph 2.8.

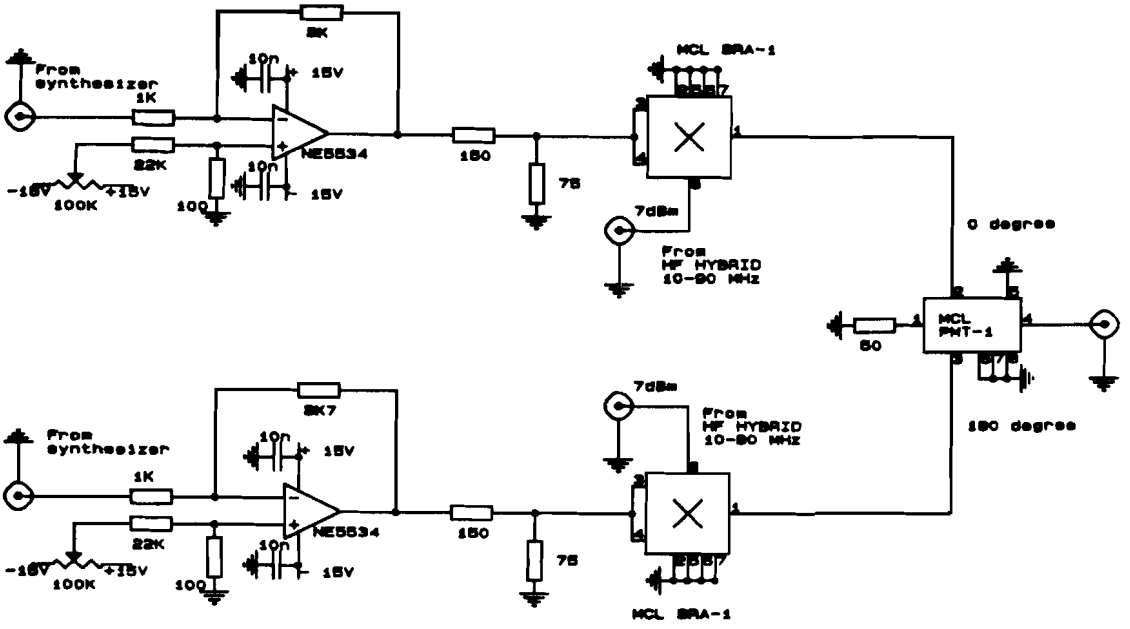


Fig 4.3: Circuit diagram of the modulator

4.4 Results and conclusions

In order to verify the capabilities of the correction factors the modulator was tested with the digital synthesizer. At different MR frequencies (21 MHz and 64 MHz) the carrier and side band suppression are measured with optimum correction factors for those frequencies. It can be concluded that at every synthesizer frequency (0 Hz - 1.6 MHz) a carrier and side band suppression of 70 dB can be achieved with the appropriate correction factor for that frequency.

Also the degradation of the suppression was measured for a certain correction factor for the whole synthesizer frequency span. The results are summarized in table 4.1, for spectrum plots see appendix 11.

Table 4.1: Measurement results

MR frequency	Synth. freq.	Carrier and sideband suppression b.c.
21 MHz	155 kHz	> 75.1 dB
	969 kHz	> 72.4 dB
64 MHz	4.76 kHz	> 73.7 dB
	946 kHz	> 68.9 dB
21 MHz	1.04 MHz - 5 kHz	69.1 - 41.2 dB
	9.8 kHz - 1.1 MHz	72.8 - 39.8 dB
64 MHz	956 kHz - 4.7 kHz	65.8 - 40.8 dB
	4.7 kHz - 1.2 Mhz	72.2 - 35.4 dB

From these measurements it can be concluded that by introducing

digital correction factors an excellent performance in the transmit chain can be obtained with regard to synthesizer and modulator. If the correction factors are controlled by the instrument control system any desired carrier and side band level can be reached. For large frequency sweeps it is advised to adapt the correction factor to the region of interest or to adjust the factors during the sweep.

An algorithm for optimizing the correction factors should contain the following steps:

- adjust the amplitude balance for optimum carrier suppression
- search for the phase correction with maximum side band suppression.
- adjust the amplitude and find the optimum correction factor with maximum side band suppression.

Acknowledgement

First of all I wish to thank J.H. den Boef and Prof. Dr. Ir. A.F. Mehlkopf for making it possible to work on such interesting subjects in the field of MR. Especially for offering their time, for useful suggestions and the pleasant cooperation.

I also want to thank all the members of the predevelopment project group and the colleagues of the hardware laboratory, without their support, this work would not have been so fruitful. Further I'm grateful to R. Proksa of Philips Forschungs Laboratorium Hamburg for sending information from his work on digital synthesizers.

Finally, I want to thank Prof. Dr. Ir. W.M.G. van Bokhoven and Ir. J.H. van den Boorn for allowing me to work at Philips Medical Systems Netherlands under their responsibility.

References

[Lit. 1] "Principles of MR imaging", Philips Medical Systems Division, 1985.

[Lit. 2] W. Koops, "MR compendium 1985", Philips Medical Systems Division, 1985.

[Lit. 3] Prof.Dr.Ir. A.F. Mehlkopf, Dr. W.M.M.J. Boveé, "Spinafbeelding: Theorie, methoden en instrumentatie", diktaat TUD, vakgroep spectroscopie en stralings technologie, sectie spin imaging, c58, mei 1988.

[Lit. 4] R.R. Ernst, J. Magn. Res., 4, pp. 280, 1971.

[Lit. 5] Richard R. Rzedzian, Ian L. Pykett, "Instant images of the human heart using a new whole body MR imaging system", AJR:149,pp. 245-250, aug. 1987.

[Lit. 6] David D. Stark, William G. Bradley, Joseph T. Ferruci, "Advances in body imaging enhance utility of MRI", Diagnostic Imaging International,pp. 30-42, feb. 1988.

[Lit. 7] Catalogus no. 17a, Anaren microwave components Inc, pp. 161, 1984.

[Lit. 8] ir. J.H. vd Boorn, ir. J.A.W. Faatz, diktaat v/h vak "Electronica bijzondere onderwerpen", TUE 1987.

[Lit. 9] W.D. Kasperkovitz, " Frequentie delers voor ultra-hoge frequenties", Philips tech. T.38,blz. 50-60,1978/79,no. 2.

[Lit. 10] ir. A.P. Verlijsdonk, "Telecommunicatiesystemen", diktaat v/h vak "Telecommunicatiesystemen",blz. 2.0-2.55, nr. 5512, THE.

[Lit. 11] P. Leuthold,"Filternetzwerke mit digitalen Shiebe-registern", Philips Research Reports Suppl. 1967, no. 5.

[Lit. 12] Richard L. Campbell, "A novel SSB modulator for laboratory signal generators",Mini-Circuits application note AN-1,pp. 1.1-1.4,1987.

[Lit. 13] H.J. Orchard, "Synthese of wideband two-phase networks", Wireless Engineer, pp. 72-81, march 1950.

[Lit. 14] R.G. Manton, "Hybrid networks and their uses in radio-frequency circuits, The radio and electronic engineer, pp. 473-489,vol. 54, no.11/12, nov/dec 1984.

[Lit. 15] S. Darlington, "Realization of a constant phase difference", Bell System Technical Journal,pp. 94-104, vol. 24, jan. 1950.

[Lit. 16] W. Saraga, "The design of wide-band phase splitting networks", Proc. of the I.R.E., 38, pp. 754-770, july 1950.

[Lit. 17] D.K. Weaver, "Design of RC wide-band 90 degree phase difference network", Proc. of the I.R.E.,pp. 671-676, vol. 42,april 1954.

[Lit. 18] S.D. Bedrosian, "Normalized design of 90 degree phase difference networks, I.R.E. trans. on circuit theory, pp. 128-136, vol CT-7, june 1960.

[Lit. 19] D.E. Norgaard, "The phase shift method of single sideband signal generation", pp. 1718-1735, Proc. I.R.E, vol. 44, dec. 1956.

[Lit. 20] H.J. Orchard, "The synthesis of RC networks to have prescribed transfer functions", Proc. I.R.E., pp. 428-432, vol. 39, april 1941.

[Lit. 21] D.G.C. Luck, "Properties of some wide-band phase splitting networks", Proc. I.R.E., pp. 147-151, vol. 37,

feb. 1949.

[Lit. 22] R.B. Dome, "Wide-band phase shift networks", Elektronics, pp. 112-115, dec. 1946.

[Lit. 23] R.E. Fisher, "Broadband twisted wire quadrature hybrids", IEEE trans. on microwave theory and techniques, pp. 355-357, vol. MTT-21, no. 5, may 1973.

[Lit. 24] R.E. Fischer, "Twisted wire quadrature hybrid directional couplers", QST, pp. 21-23, jan 1978.

[Lit. 25] T.C. Edwards, "Foundations for microstrip circuit design", Wiley-Intersience, 1981. [LPG 81 EDW]

[Lit. 26] S.R.J. Thorn, "Travelling wave dividers in direct conversion receivers", Philips Research Laboratories, Technical note no. 2386, jan 1986.

[Lit. 27] E.A. Guillemin, "Synthesis of passive networks",John Wiley & Sons, New York, pp. 64-72,pp. 107-140, 1957.[LAH 57 GUI]

[Lit. 28] Anatol I. Zverev, "Handbook of filter synthesis", John Wiley & Sons, New York, pp. 107-136, 1967.

[Lit. 29] Dr. Sönke Mehrgardt, Dipl.-Phys. Herbert Alrutz," Digitaler sinusgenerator hoher präzision", Elektronik, pp 53-57, no 5, 11.3.1983.

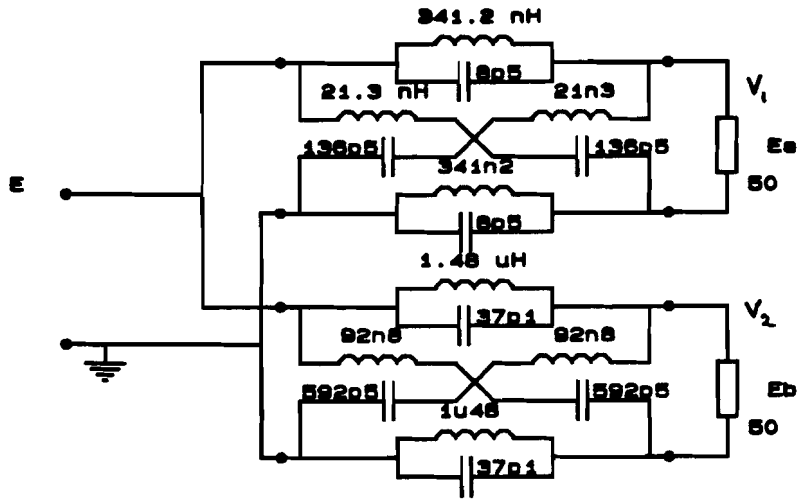
[Lit. 30] Vadimir Manassewitsch, "Frequency synthesizers theory and design", John Wiley & Sons, New York, pp. 51-61, 1980.

[Lit. 31] "Fulfilling the promise: a new high performance direct digital synthesizer design", Update product developments, vol XIII, no. 3, Burr-Brown Corporation, 1987.

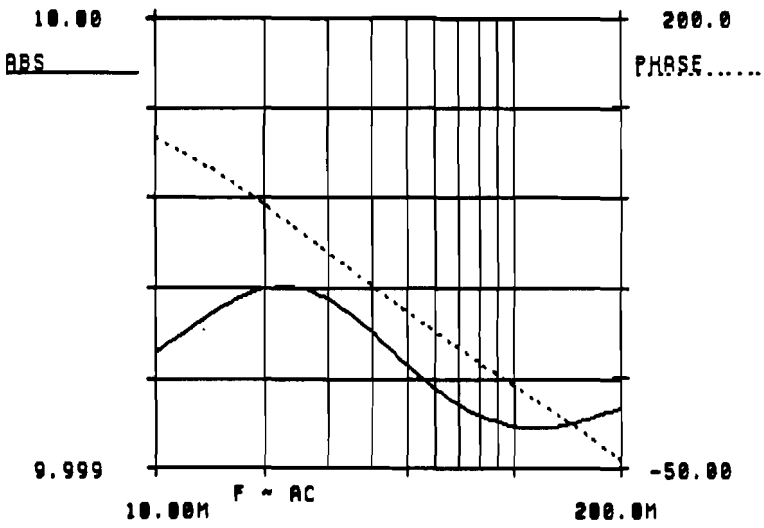
Appendix 1: Simulations on LC and RC allpass networks

This simulation shows the absolute phase and amplitude of the Orchard LC 90 degree phase shifter (10 - 200 MHz) from fig. 2.10. Afterwards the phase difference between the 2 outputs of the networks is given (should be 90 degree).

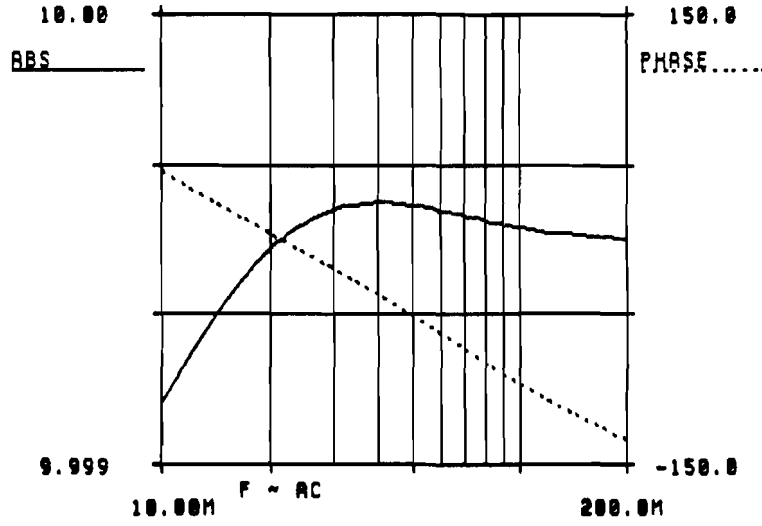
Simulated network:



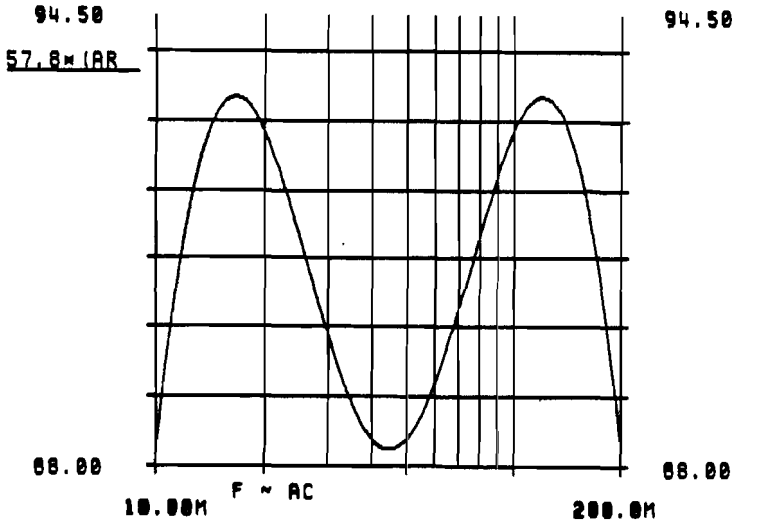
Absolute phase and amplitude output V1:



Absolute phase and amplitude output V2:

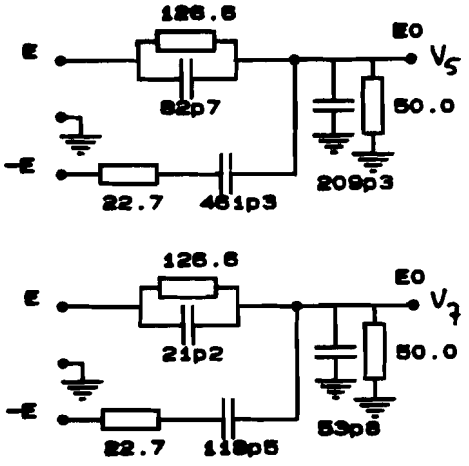


Phase difference between V1 and V2:

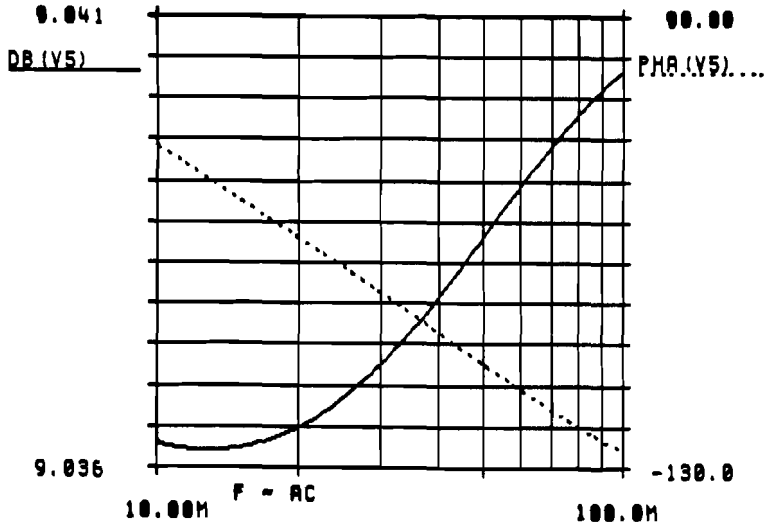


This simulation shows the amplitude and phase of a Weaver RC 90 degree phase shifter. The order of this network is n = 4. This network is used for the HF hybrid (10 - 90 MHz). Afterwards the phase difference between the output signals is given for two 90 degree shifted input signals (should be 0 degrees).

Simulated network:



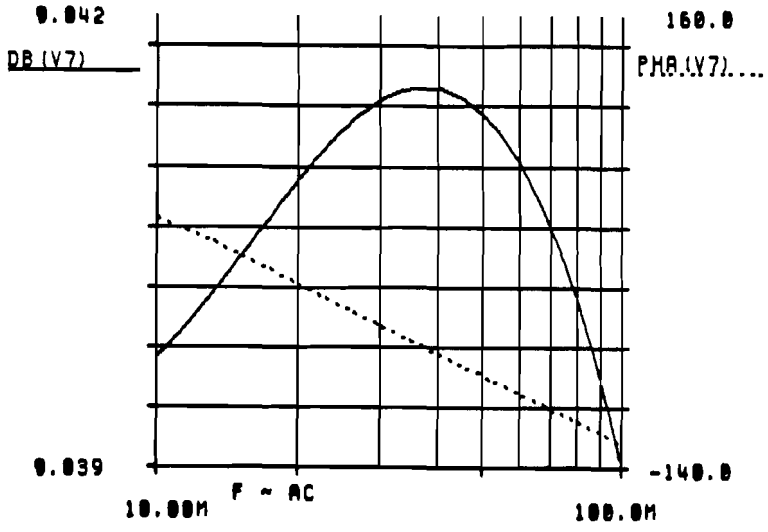
Absolute phase and amplitude output V5:



DB (V5)

PHA (V5)

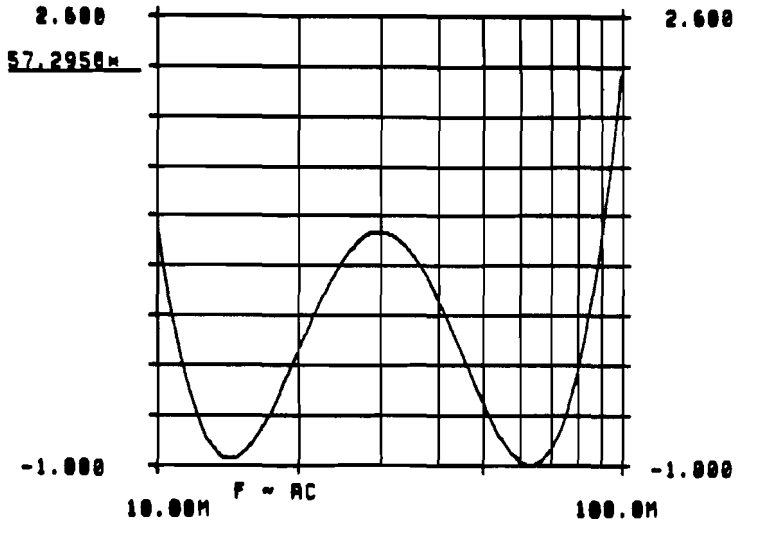
Absolute phase and amplitude output V7:



DB (V7)

PHA (V7)

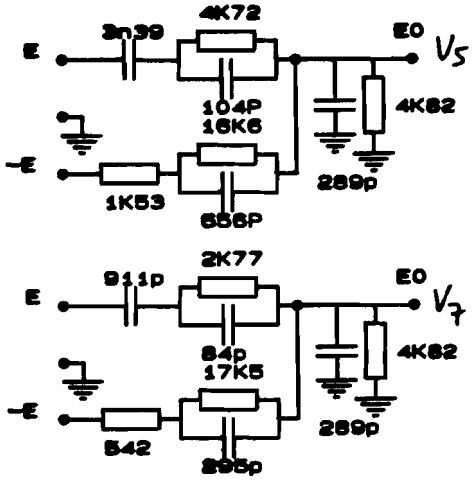
Phase difference between V5 and V7:



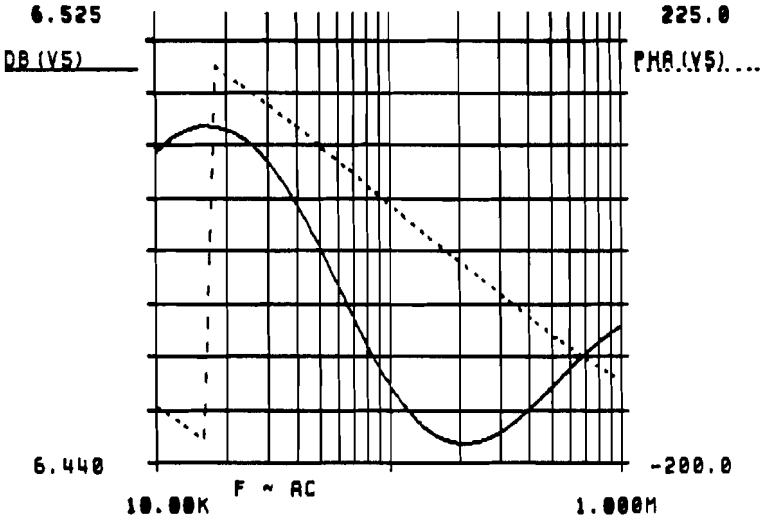
57.2958M (ARG (V5) -ARG

This simulation shows the phase and amplitude of a Weaver RC 90 degree phase shifter. The order of the network is n = 6. This circuit is used for the LF 90 degree phase shifter (10 kHz - 1MHz). Afterwards the phase difference of the output signals is given when the networks are driven by 90 degree shifted input signals (should be 0 degree).

Simulated network:



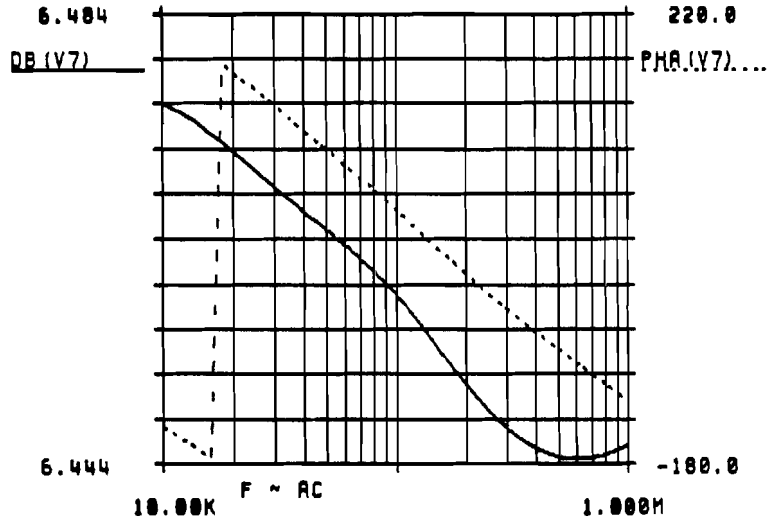
Absolute phase and amplitude output V5:



DB(V5)

PHA(V5)

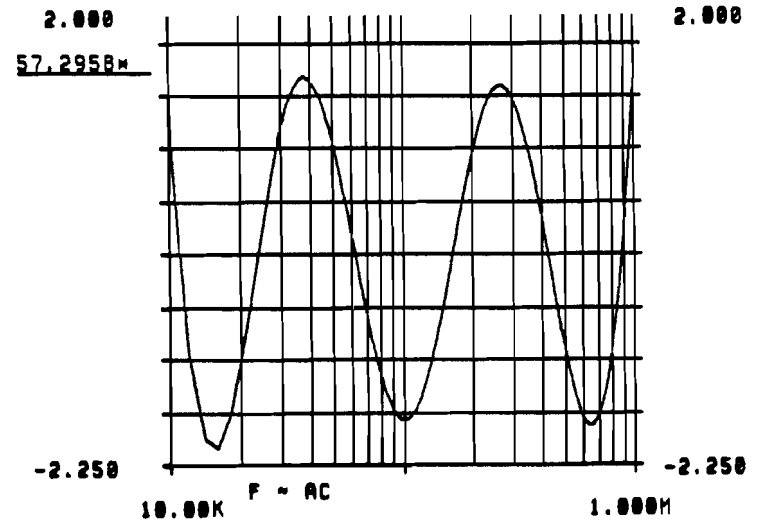
Absolute phase and amplitude output V7:



DB(V7)

PHA(V7)

Phase difference between V5 and V7:



57.2958M (ARG(V5)-ARG

Appendix 2: Fortran program to calculate RC hybrids

```
c Dit programma berekend filters van het WEAVER type.
c Het programma verwerkt de gevallen n = 4 met twee
c variaties en n = 6 met de standaard realizatie.
c Verwezen wordt voor uitvoerige informatie naar
c D.K. Weaver, "Design of wideband 90 degree phase
c difference network",Proc. o.t. I.R.E., pp. 671-676,
c vol. 42, april 1954.
c Stap 1 verondersteld een netwerkcomplexiteit n, die
c bepaald wordt uit de bandbreedte en de toegestane
c rimpel.

      REAL*8 f1,fh,pi,k,l,q,kaccent,qaccent
      INTEGER n,r,keuze,netwerk,uitvoer,IFAIL,N1
      REAL*8 phiA(10),phiB(10),phiAaccent(10),phiBaccent(10)
      REAL*8 sigmaA(10),sigmaB(10),XA(10),pA(10),pB(10)
      REAL*8 fa(10),fb(10),fasub(10),fbsub(10)
      REAL*8 Kc,Kd,Gx,Gy
      REAL*8 Kx1,px1,Cx1,Rx1,Kx2,px2,Rx2,Cx2,Ry0,Ky1,py1,Cy1,Ry1
      REAL*8 Kx0,Ky,famax,fbmax,hulp1,hulp2,hulp3
      REAL*8 RE(10), XZ(10), YZ(10), TOL, X02AAF

      uitvoer = 0

      WRITE (*,'('' Geef de ondergrens(f1) resp. bovengrens(fh) '')')
      WRITE (*,'('' van het gewenste frequentie bereik(Hz).     '')')
      WRITE (*,'('' '')')
      WRITE (*,'('' Input scheiden met komma(,) en nummers moeten '')')
      WRITE (*,'('' een punt(.) bevatten (floating point invoer). '')')

      READ (*,1000) f1,fh
1000  FORMAT(2F20.1)

      WRITE (*,'('' De ondergrens(f1) resp. bovengrens(fh)      '')')
      WRITE (*,'('' in Hz zijn:                                 '')')
      WRITE (*,1000) f1,fh

c stap 2 berekend een aantal constanten nodig voor de berekening
c van het filter.

      pi=3.141592654

      kaccent = f1/fh
      k = sqrt(1.0-(kaccent**2))
      l = 0.5 * ((1.0-sqrt(k))/(1.0+sqrt(k)))
      qaccent = l + 2.0*(l**5) + 15.0*(l**9)
      q = exp ((pi**2)/log(qaccent))

      WRITE (*,1010) kaccent
1010  FORMAT(' kaccent = ',F20.8)
      WRITE (*,1020) k
1020  FORMAT(' k       = ',F20.8)
      WRITE (*,1030) l
1030  FORMAT(',l       = ',F20.8)
      WRITE (*,1040) qaccent
1040  FORMAT(' qaccent = ',F20.8)
      WRITE (*,1050) q
1050  FORMAT(' q       = ',F20.8)

c stap3 berekend een aantal hoeken nodig voor de overdrachtsfunctie
c van het netwerk. Het aantal hoeken wordt bepaald door de netwerk-
c complexiteit n.

      WRITE (*,'('' rimpel diagram.                             '')')
      WRITE (*,'('' De invoer bestaat uit een getal zonder punt.'')')
      WRITE (*,'('' VOORLOPIG DIENT n = 4 of 6 TE ZIJN.         '')')

      READ (*,1060) n
1060  FORMAT(I6)
      WRITE (*,1070) n
1070  FORMAT(' De netwerkcomplexiteit(n) = ',I6)
      WRITE (*,'('' '')')

      IF ( (n.NE.4).AND.(n.NE.6) ) THEN
         WRITE (*,'('' ERROR: n is niet 4 of 6 '')')
         STOP
      ENDIF

      DO r=1,n/2

         phiA(r) = (90.0/(2*n))*(4*r-3)
         WRITE (*,1080) r,phiA(r)
1080     FORMAT(' phiA(r) met index ',I6,' = ',F20.3,' degree')

         phiB(r) = (90.0/(2*n))*(4*r-1)
         WRITE (*,1090) r,phiB(r)
1090     FORMAT(' phiB(r) met index ',I6,' = ',F20.3,' degree')

      ENDDO

      WRITE (*,'('' '')')
      DO r=1,n/2

c radial is een constante om degree (graden) om te rekenen naar radialen
c evenzo dient degree om radialen om te zetten in degree (graden).

         radial = 0.01745329252
         degree = 57.29577951

         phiAaccent(r) = atan( ((q**2-q**6)*sin(4*radial*phiA(r)))
     &                        /(1.0+(q**2+q**6)*cos(4*radial*phiA(r))) )
         WRITE (*,1110) r,phiAaccent(r)*degree
1110     FORMAT(' phiAaccent(r) met index ',I6,' = ',
     &          F20.8,' degree ')

         phiBaccent(r) = atan( ((q**2-q**6)*sin(4*radial*phiB(r)))
     &                        /(1.0+(q**2+q**6)*cos(4*radial*phiB(r))) )
         WRITE (*,1120) r,phiBaccent(r)*degree
1120     FORMAT(' phiBaccent(r) met index ',I6,' = ',
     &          F20.8,' degree ')

      ENDDO
      WRITE (*,'('' '')')

c stap 4 berekend de sigma's in de overdracht functie.

      DO r=1,n/2

         sigmaA(r) = ( 1.0/( sqrt(kaccent) ) )*
     &               tan( (radial*phiA(r))-phiAaccent(r) )
         WRITE (*,1140) r,sigmaA(r)
1140     FORMAT(' sigmaA(r) met index ',I6,' = ',F20.8)

         sigmaB(r) = ( 1.0/( sqrt(kaccent) ) )*
     &               tan( (radial*phiB(r))-phiBaccent(r) )
```

```
1150     FORMAT(' sigmaB(r) met index ',I6,' = ',F20.8)

      ENDDO
      WRITE (*,'('' '')')

c stap 5 bepaald de overdracht functie. Deze is triviaal indien de
c sigma's berekend zijn

c stap 6 bepaald de maximale waarde van K die gebruikt kan worden
c zodanig dat het netwerk alleen met R's en C's gerealizeerd kan
c worden.

c voorlopig kan alleen het geval n=4 of n=6 behandeld worden.

      WRITE (*,'('' VOORLOPIG ALLEEN n=4 of n=6 ! '')')

c (a) bepaling van de waarden van p waarvoor de afgeleide nul is.
c Deze is voorlopig nog met de hand uitgerekend.

      IF (n.EQ.4) THEN

        RE(1) = 2.0*(sigmaA(1)+sigmaA(2))
        RE(2) = 0.0
        RE(3) = -2.0*( (sigmaA(1)**2)*sigmaA(2)+sigmaA(1)*
     &          (sigmaA(2)**2) )

      ENDIF

c Voor het geval n=6 kunnen de maxima en minima van fa(p) en fb(p)
c bepaald worden door de nulpunten van de afgeleide te zoeken. De
c afgeleide is met de hand bepaald en omgerekend tot een polynoom.
c De oplossingen van dit polynoom zijn de gezochte waarden.

      IF (n.EQ.6) THEN

        hulp1 = sigmaA(1)+sigmaA(2)+sigmaA(3)
        RE(1) = 2.0*hulp1
        RE(2) = 0.0
        hulp2 = sigmaA(1)*sigmaA(2)+sigmaA(1)*sigmaA(3)
     &          +sigmaA(2)*sigmaA(3)
        hulp3 = sigmaA(1)*sigmaA(2)*sigmaA(3)
        RE(3) = 6.0*hulp3-2.0*hulp1*hulp2
        RE(4) = 0.0
        RE(5) = 2.0*hulp2*hulp3

      ENDIF

        TOL = X02AAF(0.1D0)
        IFAIL = 1
        N1 = n-1

        CALL C02AEF(RE, N1, pA, YZ, TOL, IFAIL)
c Dit is een naglib routine om nulpunten te vinden van een polynoom.
        DO r=1,n-1
          IF (abs(YZ(r)).GT.0.001) THEN
            WRITE (*,'('' ERROR: er zijn niet reele wortels '')')
            WRITE (*,'('' in afgeleide. (stap 6(a)) '')')
            STOP
            ENDIF
        ENDDO

        IF (N1.NE.1) THEN
          WRITE (*,'('' ERROR: niet alle nulpunten gevonden! '')')
          ENDIF

        IF (IFAIL.NE.0) THEN
          WRITE (*,'('' ERROR: fout in nulpunten zoeken! '')')
          STOP
        ENDIF

        WRITE(*,'('' De minima en maxima van p zijn: '')')

        DO r=1,n-2

          WRITE (*,1180) pA(r)
1180      FORMAT(F20.8)

        ENDDO
        WRITE (*,'('' '')')

c (b) vul de negatieve waarden van p in, in de overdrachtsfunktie.
c Hieruit volgen een aantal getallen groter dan 1. Kies hiervan
c het maximum. De reciproke vormt de gezochte K.

        famax = 0.0
        DO r=1,n-2

          fa(r) = 1.0
          IF (pA(r).LT.0) THEN
            DO index=1,n/2
              fasub(index) =( pA(r)-sigmaA(index) )/
     &                      ( pA(r)+sigmaA(index) )
              fa(r) = fa(r)*fasub(index)
            ENDDO

            WRITE (*,1210) r,fa(r)
1210        FORMAT(' fa(r) met index ',I6,' = ',F20.8)

c Bepaling van het maximum van fa.

            IF (abs(fa(r)).GT.famax) THEN
              famax = abs(fa(r))
            ENDIF

            IF (abs(fa(r)).LT.1) THEN
              WRITE (*,'('' ERROR: de waarden ingevuld in over- '')')
              WRITE (*,'('' dracht funktie voldoen niet. '')')
              STOP
            ENDIF

          ENDIF

          ENDDO

c Stap 6(c), herhaling voor fb(r).

          IF (n.EQ.4) THEN

            RE(1) = 2.0*(sigmaB(1)+sigmaB(2))
            RE(2) = 0.0
            RE(3) = -2.0*( (sigmaB(1)**2)*sigmaB(2)+sigmaB(1)*
     &              (sigmaB(2)**2) )

          ENDIF

c Voor het geval n=6 kunnen de maxima en minima van fa(p) en fb(p)
```

```
c De oplossingen van dit polynoom zijn de gezochte waarden.

        IF (n.EQ.6) THEN

          hulp1 = sigmaB(1)+sigmaB(2)+sigmaB(3)
          RE(1) = 2.0*hulp1
          RE(2) = 0.0
          hulp2 = sigmaB(1)*sigmaB(2)+sigmaB(1)*sigmaB(3)
     &            +sigmaB(2)*sigmaB(3)
          hulp3 = sigmaB(1)*sigmaB(2)*sigmaB(3)
          RE(3) = 6.0*hulp3-2.0*hulp1*hulp2
          RE(4) = 0.0
          RE(5) = 2.0*hulp3*hulp2

        ENDIF

        TOL = X02AAF(0.1D0)
        IFAIL = 1
        N1 = n-1
        CALL C02AEF(RE, N1, pB, YZ, TOL, IFAIL)
c Dit is een naglib routine om nulpunten te vinden van een polynoom.
        DO r=1,n-1
          IF (abs(YZ(r)).GT.0.001) THEN
            WRITE (*,'('' ERROR: er zijn niet reele wortels. '')')
            WRITE (*,'('' in afgeleide. stap 6(b) '')')
            STOP
          ENDIF
        ENDDO

        IF (N1.NE.1) THEN
          WRITE (*,'('' ERROR: niet alle nulpunten gevonden! '')')
          STOP
        ENDIF

        IF (IFAIL.NE.0) THEN
          WRITE (*,'('' ERROR: fout in nulpunten zoeken! '')')
          STOP
        ENDIF

        WRITE(*,'('' De minima en maxima van p zijn: '')')

        DO r=1,n-2

          WRITE (*,1240) pB(r)
1240      FORMAT(F20.8)

        ENDDO
        WRITE (*,'('' '')')

c (b) vul de negatieve waarden van p in, in de overdrachtsfunktie.
c Hieruit volgen een aantal getallen groter dan 1. Kies hiervan
c het maximum. De reciproke vormt de gezochte K.

c Bepaling van het maximum van fb.

        fbmax = 0.0
        DO r=1,n-2

          fb(r) = 1.0
          IF (pB(r).LT.0) THEN
            DO index=1,n/2
               fbsub(index) =( pB(r)-sigmaB(index) )/
```

```
          ENDDO

          WRITE (*,1270) r,fb(r)
1270      FORMAT(' fb(r) met index ',I6,' = ',F20.8)

c Bepaling van het maximum.

          IF (abs(fb(r)).GT.fbmax) THEN
            fbmax = abs(fb(r))
          ENDIF

          IF (abs(fb(r)).LT.1) THEN
            WRITE (*,'('' ERROR: de waarden ingevuld in over- '')')
            WRITE (*,'('' dracht funktie voldoen niet. '')')
            STOP
          ENDIF

        ENDIF

        ENDDO

c ALs n even is (n=4 of n=6) dan moeten de beide waarden famax en
c fbmax (en dus de bijbehorende K's gelijk zijn. Anders moet de
c kleinste K genomen worden (is de grootste van famax en fbmax).

        IF (MOD(n,2).EQ.0) THEN
          IF (abs(famax-fbmax).GT.0.001) THEN
            WRITE (*,'('' ERROR: Verschillende K s '')')
            WRITE (*,'('' in de overdracht. '')')
            STOP
          ENDIF
        ENDIF

        Kc =1.0/ ( max(famax,fbmax) )
        WRITE(*,1290) Kc
1290    FORMAT(' Kc wordt: ',F20.8)
        WRITE(*,'('' '')')

c Stap 7 lost een aantal vergelijkingen op. De oplossingen
c dienen om de impedantie functie's te construeren.
c Bepaald of het geval van 7(a), keuze=0 of 7(b), keuze=1
c uitgerekend wordt. Netwerk dient voor A of B netwerk.

        DO netwerk = 0,1
        WRITE (*,'('' '')')
        IF (netwerk.EQ.0) THEN
          WRITE (*,'('' A NETWERK: '')')
        ENDIF
        IF (netwerk.EQ.1) THEN
          WRITE (*,'('' B NETWERK: '')')
        ENDIF
        WRITE (*,'('' '')')

        DO keuze = 0,1

c De volgende waarden zijn geschikt voor het oplossen van de vergelijking
c f(p)-1 = 0 zoals bij stap 7(a).
c De volgende waarden zijn voor het geval n=4.

1320    IF (n.EQ.4) THEN
        IF (keuze.EQ.0) THEN
          IF (netwerk.EQ.0) THEN
```

```
            RE(3) = Kc*sigmaA(1)*sigmaA(2)-sigmaA(1)*sigmaA(2)
          ENDIF
          IF (netwerk.EQ.1) THEN
            RE(1) = Kc-1.0
            RE(2) = -( Kc*(sigmaB(1)+sigmaB(2))+sigmaB(1)+sigmaB(2) )
            RE(3) = Kc*sigmaB(1)*sigmaB(2)-sigmaB(1)*sigmaB(2)
          ENDIF
        ENDIF
        ENDIF

c De volgende waarden zijn voor het geval n=6, vgl f(p)-1=0

        IF (n.EQ.6) THEN
        IF (keuze.EQ.0) THEN
          IF (netwerk.EQ.0) THEN
            RE(1) = Kc-1.0
            RE(2) = (-Kc-1.0)*( sigmaA(1)+sigmaA(2)+sigmaA(3) )
            RE(3) = (Kc-1.0)*( sigmaA(1)*sigmaA(2)+sigmaA(1)*sigmaA(3)
     &              +sigmaA(2)*sigmaA(3) )
            RE(4) = (-Kc-1.0)*sigmaA(1)*sigmaA(2)*sigmaA(3)
          ENDIF
          IF (netwerk.EQ.1) THEN
            RE(1) = Kc-1.0
            RE(2) = (-Kc-1.0)*( sigmaB(1)+sigmaB(2)+sigmaB(3) )
            RE(3) = (Kc-1.0)*( sigmaB(1)*sigmaB(2)+sigmaB(1)*sigmaB(3)
     &              +sigmaB(2)*sigmaB(3) )
            RE(4) = (-Kc-1.0)*sigmaB(1)*sigmaB(2)*sigmaB(3)
          ENDIF
        ENDIF
        ENDIF

c De volgende waarden zijn geschikt voor het oplossen van de vergelijking
c f(p)+1 = 0 zoals bij stap 7(b).
c De volgende waarden zijn voor het geval n=4.

        IF (n.EQ.4) THEN
        IF (keuze.EQ.1) THEN
          IF (netwerk.EQ.0) THEN
            RE(1) = Kc+1.0
            RE(2) = Kc*(-sigmaA(1)-sigmaA(2))+sigmaA(1)+sigmaA(2)
            RE(3) = Kc*sigmaA(1)*sigmaA(2)+sigmaA(1)*sigmaA(2)
          ENDIF
          IF (netwerk.EQ.1) THEN
            RE(1) = Kc+1.0
            RE(2) = Kc*(-sigmaB(1)-sigmaB(2))+sigmaB(1)+sigmaB(2)
            RE(3) = Kc*sigmaB(1)*sigmaB(2)+sigmaB(1)*sigmaB(2)
          ENDIF
        ENDIF
        ENDIF

c Het volgende geldt voor het geval n=6, vgl f(p)+1=0

        IF (n.EQ.6) THEN
        IF (keuze.EQ.1) THEN
          IF (netwerk.EQ.0) THEN
            RE(1) = Kc+1.0
            RE(2) = (-Kc+1.0)*( sigmaA(1)+sigmaA(2)+sigmaA(3) )
            RE(3) = (Kc+1.0)*( sigmaA(1)*sigmaA(2)+sigmaA(1)*sigmaA(3)
     &              +sigmaA(2)*sigmaA(3) )
            RE(4) = (-Kc+1.0)*sigmaA(1)*sigmaA(2)*sigmaA(3)
          ENDIF
          IF (netwerk.EQ.1) THEN
            RE(3) = (Kc+1.0)*( sigmaB(1)*sigmaB(2)+sigmaB(1)*sigmaB(3)
     &              +sigmaB(2)*sigmaB(3) )
            RE(4) = (-Kc+1.0)*sigmaB(1)*sigmaB(2)*sigmaB(3)
          ENDIF
        ENDIF
        ENDIF

        IF ( (keuze.NE.0).AND.(keuze.NE.1) ) THEN
          WRITE (*,'('' ERROR: variable keuze fout. '')')
          STOP
        ENDIF

        TOL = X02AAF(0.1D0)
        IFAIL = 1

        CALL C02AEF(RE, (n/2)+1, XZ, YZ, TOL, IFAIL)

        DO r=1,(n/2)+1
        IF (abs(YZ(r)).GT.0.001) THEN
          WRITE (*,'('' ERROR: er zijn niet reele wortels. '')')
          WRITE (*,'('' in fa of fb +of-1= 0 '')')
          STOP
          ENDIF
        ENDDO

        DO r=(n/2)+1,10
          XZ(r) = 0.0
        ENDDO

        IF (N1.NE.1) THEN
          WRITE (*,'('' ERROR: niet alle nulpunten gevonden! '')')
          STOP
        ENDIF

        IF (IFAIL.NE.0) THEN
          WRITE (*,'('' ERROR: fout in nulpunten zoeken! '')')
          STOP
        ENDIF

c Aan de hand van het geval 7(a) of 7(b) worden de maxima van de
c wortels van de verschillende vergelijkingen bepaald en toegekend
c aan de sigmaX en sigmaY.
c M01ANF is een naglib sorteer routine.

        IF (keuze.EQ.0) THEN

          CALL M01ANF ( XZ,1,(n/2)+1,IFAIL )
          IF (IFAIL.NE.0) THEN
            WRITE (*,'('' ERROR: met sorteren '')')
            STOP
          ENDIF
          sigma01 = XZ(1)
          sigma23 = XZ(2)
          IF (n.EQ.6) THEN
            sigma32 = XZ(3)
          ENDIF

        ENDIF

        IF (keuze.EQ.1) THEN

          CALL M01ANF (XZ,1,(n/2)+1,IFAIL)
```

```
          STOP
        ENDIF
        sigma12 = XZ(1)
        sigma21 = XZ(2)
        IF (n.EQ.6) THEN
          sigma34 = XZ(3)
        ENDIF

      ENDIF

c Eind van de loop over keuze
      ENDDO

      WRITE (*,1350) sigma01
1350  FORMAT(' sigma01 (sigmay1): ',F20.8)

      WRITE (*,1360) sigma23
1360  FORMAT(' sigma23 (sigmay3): ',F20.8)

      IF (n.EQ.6) THEN
        WRITE (*,1370) sigma32
1370  FORMAT(' sigma32 (sigmax2): ',F20.8)
      ENDIF

      WRITE (*,1380) sigma12
1380  FORMAT(' sigma12 (sigmax1): ',F20.8)

      WRITE (*,1390) sigma21
1390  FORMAT(' sigma21 (sigmay2): ',F20.8)

      IF (n.EQ.6) THEN
        WRITE (*,1400) sigma34
1400  FORMAT(' sigma34 (sigmax3): ',F20.8)
      ENDIF
      WRITE(*,'('' ''')')

      Kd = (1.0-Kc)/(1.0+Kc)

      IF (n.EQ.6) THEN

      XA(1) = sigma01
      XA(2) = sigma23
      XA(3) = sigma32
      XA(4) = sigma12
      XA(5) = sigma21
      XA(6) = sigma34
      DO j=1,6
      X = XA(j)
      Y = Kc * ( X - sigmaA(1) ) * ( X - sigmaA(2) ) *
     &       ( X - sigmaA(3) ) / (
     &       ( X + sigmaA(1) ) * ( X + sigmaA(2) ) *
     &       ( X + sigmaA(3) ) )
      WRITE(*,1405) X,Y
1405  FORMAT(' fpA ( ',F16.8,' ) = ',F16.8)
      ENDDO

      WRITE(*,'('' ''')')

      DO j=1,6
      X = XA(j)
      Y = Kc * ( X - sigmaB(1) ) * ( X - sigmaB(2) ) *
     &       ( X - sigmaB(3) ) / (
```

```
      WRITE(*,1406) X,Y
1406  FORMAT(' fpB ( ',F16.8,' ) = ',F16.8)
      ENDDO

      WRITE(*,'('' ''')')

      ENDIF

c In stap 8 kunnen uit de berekende wortels de impedantie functies
c van de voor geschreven vorm geconstrueerd worden. Verder wordt
c de relatie tussen de Kx en Ky bepaald.

      WRITE(*,1410) Kd
1410  FORMAT(' Kx en Ky verhouden zich als Kx/Ky=Kd: ',F20.8)
      WRITE(*,'('' ''')')

c In stap 9 worden de belastings weerstand en capaciteit berekend
c waarmee het netwerk belast dient te worden. Allereerst de capaciteit.

c Er kan gekozen worden voor twee uitvoer mogelijkheden. De eerste
c betreft de uitvoer zoals voorgesteld in het artikel van Weaver, de
c tweede vorm is meer geschikt voor aanpassing aan bron impedanties.

      IF (n.EQ.4) THEN
        IF (netwerk.EQ.0) THEN
          WRITE (*,'('' ''')')
          WRITE (*,'('' Kies uitvoer (1=met bronimpedantie):''')')
          WRITE (*,'('' ''')')
          READ(*,1425) uitvoer
1425      FORMAT(I3)
        ENDIF
        WRITE (*,1422) uitvoer
1422    FORMAT(' uitvoer = ',I3)
        WRITE (*,'('' ''')')

      ENDIF
      IF (uitvoer.NE.1) THEN
        WRITE (*,1420) 2*Kd
1420    FORMAT(' C1 = 2*Kd/(2*pi*f1*Kx) met 2*Kd:      ',F20.8)
      ENDIF

c In het vervolg van stap 9 wordt ingegaan op de belastings weerstand
c waarmee het netwerk belast dient te worden.

      IF (n.EQ.4) THEN
        Gx = - sigma21
        IF (uitvoer.EQ.1) THEN
          Gx =  sigma21/sigma01
        ENDIF
      ENDIF

      IF (n.EQ.6) THEN
        Gx = - sigma21*sigma34/sigma32
      ENDIF

      WRITE (*,1430) Gx
1430  FORMAT(' Gx = a/Kx   wordt met a: ',E20.8)

      Gy = -Kd*sigma01*sigma23/sigma12
        IF (uitvoer.EQ.1) THEN
          Gy = Kd*sigma23/sigma12
        ENDIF
```

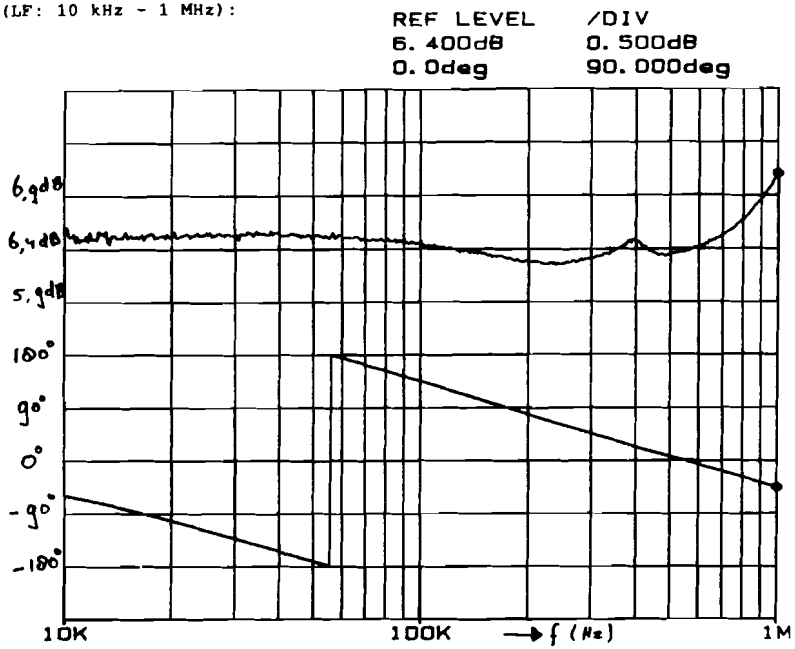
```
1440    FORMAT(' Gy = a/Kx   wordt met a: ',E20.8)

        IF (Gx.LT.Gy) THEN
           g=Gx
                    ELSE
           g=Gy
        ENDIF

        WRITE (*,1450) g
1450    FORMAT(' G1 = 2*g/Kx wordt met g: ',E20.8)
        WRITE (*,'('' ')')

        IF (uitvoer.NE.1) THEN
          WRITE (*,1452) Kd/(pi*f1)
1452      FORMAT(' C1 = a/Kx met a:',E20.8)
        ENDIF

        WRITE (*,1455) 1.0/(2.0*g)
1455    FORMAT(' R1 = a*Kx met a:',E20.8)
        WRITE(*,'('' ')')

        IF (n.EQ.4) THEN

c In stap 10 worden de element waarden bepaald. Dit wordt gedaan door
c Zx en Zy in partial fraction vorm te expanderen. Hier is dit voor
c het geval N=4 voorgerekend.


c Allereerst voor het Zx,in het geval 10(b).

        IF(uitvoer.NE.1) THEN

        Kx1 = 1.0/(1.0-Kd)
        px1 = (-sigma21-g)/(1.0-Kd)
        WRITE (*,1460) Kx1,px1
1460    FORMAT(' Zx1 = a*Kx/p+b      met a resp. b: ',2E20.8)
        WRITE (*,'('' ')')


c Vervolgens kunnen de element waarden bepaald worden.

        Cx1 = 1.0/(2.0*pi*f1*Kx1)
        WRITE (*,1470) Cx1
1470    FORMAT(' Cx1 = a/Kx                  met a: ',E20.8)
        Rx1 = Kx1/px1
        WRITE (*,1480) Rx1
1480    FORMAT(' Rx1 = b*Kx                  met b: ',E20.8)
        WRITE (*,'('' ')')

c Stap 10(d), herhaling van (a), (b) en (c) voor Zy.

        deler = Kd*-sigma01+Kd*-sigma23+sigma12*Kd-g
        Ry0 = 1.0/deler
        Ky1 = -sigma12/deler
        WRITE (*,1490) Ry0,Ky1
1490    FORMAT(' Zy1 = a*Kx + b*Kx/p met a resp. b: ',2E20.8)
        WRITE (*,'('' ')')

        WRITE (*,1500) Ry0
1500    FORMAT(' Ry0 = a*Kx met a:                   ',E20.8)
        Cy1 = 1.0/(2.0*pi*f1*Ky1)
        WRITE (*,1510) Cy1
        WRITE (*,'('' Ry1 = oneindig    ')')
        WRITE (*,'('' ')')
        ENDIF
c Einde van het geval uitvoer=0.

        IF (uitvoer.EQ.1) THEN

        px1 = (-sigma21+sigma01*g)/(1.0-g)
        Kx0 = 1.0/(1.0-g)
        Kx1 = (-px1-sigma01)*Kx0
        WRITE (*,'('' Zx1 = a*Kx + b*Kx/(p+c) met a,b resp. c: ')')
        WRITE (*,1520)Kx0,Kx1,px1
1520    FORMAT(3E20.8)
        WRITE (*,'('' ')')

        WRITE (*,1530)Kx0
1530    FORMAT(' Rx0 = a*Kx met a: ',E20.8)

        WRITE (*,1540)Kx1/px1
1540    FORMAT(' Rx1 = a*Kx met a: ',E20.8)

        WRITE (*,1550)1.0/(Kx1*2.0*pi*f1)
1550    FORMAT(' Cx1 = a/Kx met a: ',E20.8)
        WRITE (*,'('' ')')

        Ky0 = 1.0/(Kd*(1.0-(g/Kd)))
        Ky1 = -sigma12*Ky0

        WRITE (*,'('' Zy1 = a*Kx + b*Kx/p met a resp. b: ')')
        WRITE (*,1560)Ky0,Ky1
1560    FORMAT(3E20.8)
        WRITE (*,'('' ')')

        WRITE (*,1570)Ky0
1570    FORMAT(' Ry0 = a*Kx met a: ',E20.8)

        WRITE (*,1590)1.0/(Ky1*2*pi*f1)
1590    FORMAT(' Cy1 = a/Kx met a: ',E20.8)
        WRITE (*,'('' ')')

        ENDIF
c Einde van het geval uitvoer=1.

        ENDIF
c Einde van het geval n=4.


        IF (n.EQ.6) THEN

c Stap 10(a), (b) en (c) voor Zx en Zx1 in het geval van n=6.

        px2 = ( -sigma21-sigma34-g+(Kd*sigma32) )/(1.0-Kd)
        Kx1 = -sigma32/(px2*(1.0-Kd))
        Kx2 = (1.0/(1.0-Kd))*(1.0+sigma32/px2)
        WRITE (*,'('' Zx1 = a*Kx/p + b*Kx/(p+c) met a,b resp. c: ')')
        WRITE (*,1600) Kx1,Kx2,px2
1600    FORMAT(3E20.8)
        WRITE (*,'('' ')')
        WRITE (*,'('' Rx1 is oneindig ')')
        Rx2 = Kx2/px2
        WRITE (*,1610) Rx2
1610    FORMAT(' Rx2 = a*Kx met a:                   ',E20.8)
```

```
        WRITE (*,1620) Cx1
1620    FORMAT(' Cx1 = a/Kx met a:                ',E20.8)

        Cx2 = 1.0/(2.0*pi*f1*Kx2)
        WRITE (*,1630) Cx2
1630    FORMAT(' Cx2 = a/Kx met a:                ',E20.8)
        WRITE (*,'('' ''')')

c Stap 10(d), herhaling van (a), (b) en (c) voor Zy.

        alpha = Kd*(-sigma01-sigma23)-g+Kd*sigma12
        Ry0 = 1.0/alpha
        py1 = (Kd*sigma01*sigma23+sigma12*g)/alpha
        Ky1 = (-sigma12-py1)/alpha
        WRITE (*,'('' Zy1 = a*Kx + b*Kx/(p+c) met a,b resp. c: ''')')
        WRITE (*,1640) Ry0,Ky1,py1
1640    FORMAT(3E20.8)

        WRITE (*,'('' ''')')
        WRITE (*,1650) Ry0
1650    FORMAT(' Ry0 = a*Kx met a:                ',E20.8)

        Cy1 = 1.0/(2.0*pi*f1*Ky1)
        WRITE (*,1660) Cy1
1660    FORMAT(' Cy1 = a/Kx met a:                ',E20.8)

        Ry1 = Ky1/py1
        WRITE (*,1670) Ry1
1670    FORMAT(' Ry1 = a*Kx met a:                ',E20.8)
        WRITE (*,'('' ''')')

        ENDIF
c Eind van het geval n=6.

c Eind van de loop over netwerk

        ENDDO
        STOP
        END
```

Appendix 3: Measurements on the hybrids

Both the LF hybrid (10 kHz - 1 MHz) and HF hybrid (10 MHz - 90 MHz) are tested. The results show the absolute phase and amplitude and the phase difference. Afterwards the output spectrum of the HF hybrid is given to show the relatively high spurious.

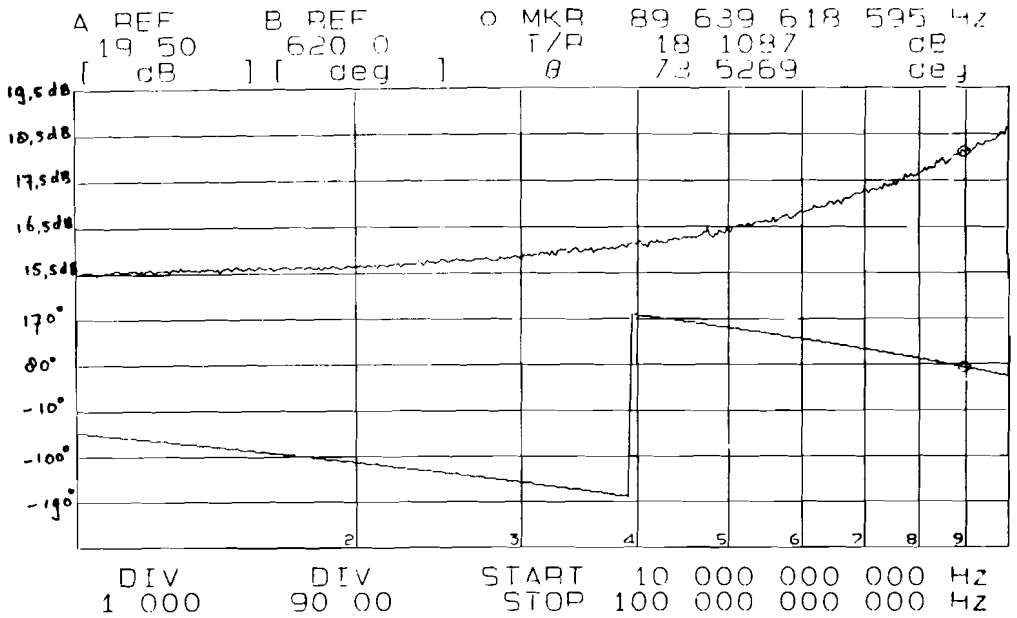
Absolute amplitude and phase RC quad hybrid n=6
(LF: 10 kHz - 1 MHz):



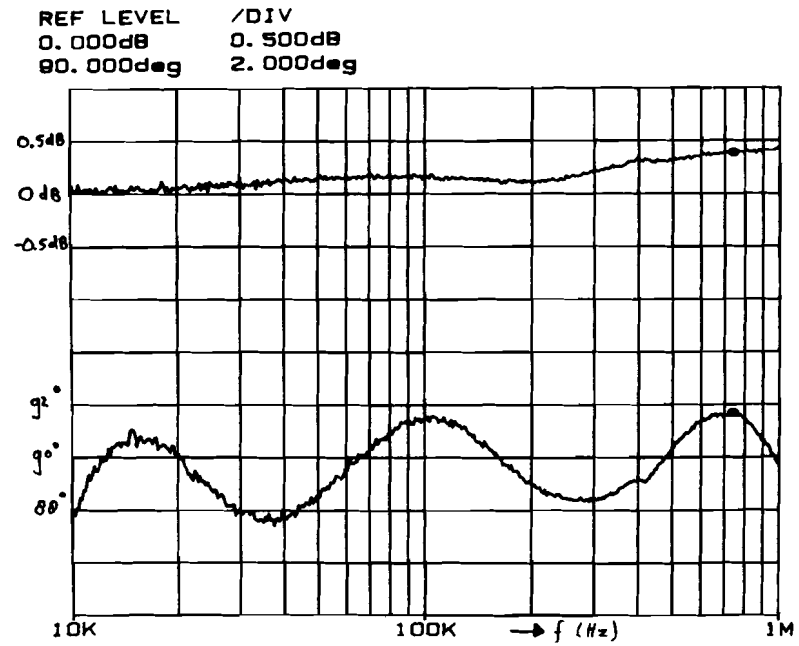
amplitude

phase

Absolute amplitude and phase RC quad hybrid n=4
(HF: 10 MHz - 90 MHz):



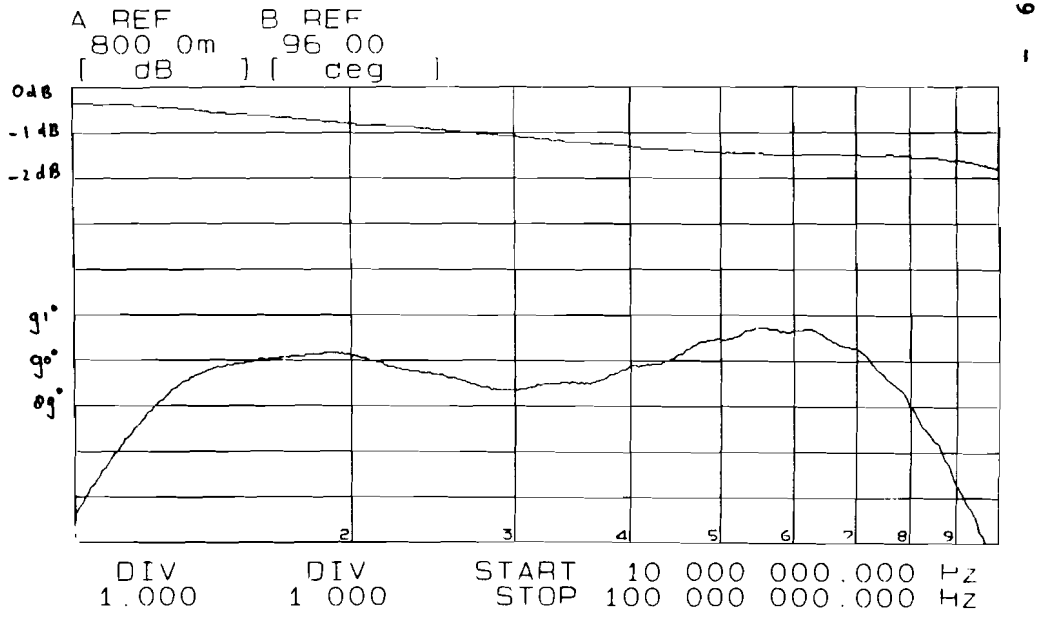
Amplitude difference and phase difference RC quad
hybrid n=6 (LF: 10 kHz - 1 MHz):

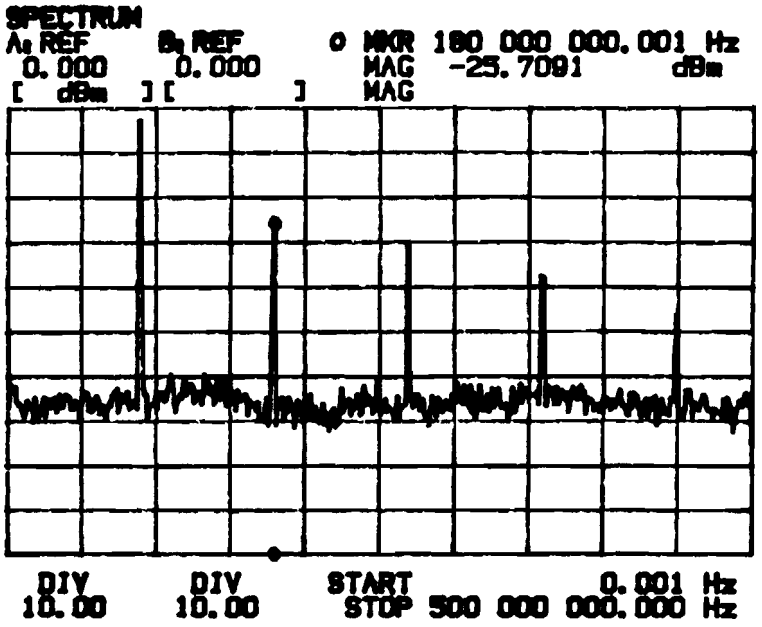


amplitude

phase

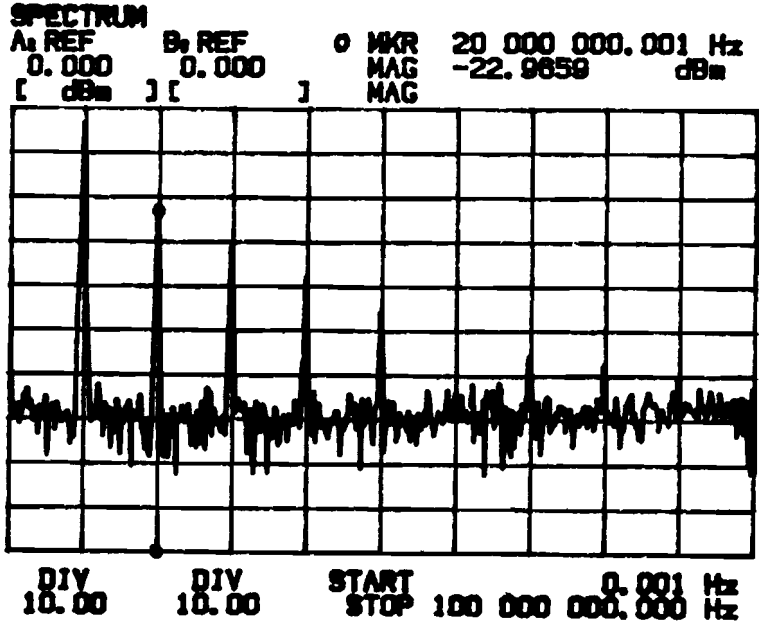
Amplitude difference and phase difference RC quad
hybrid n=4 (HF: 10 MHz - 90 MHz):





Spurious RC quad hybrid (HF: 10 MHz - 90 MHz)
2th harmonic > 20 dB down

1) 10 MHz output = 7 dBm
(- 3 dBm + 10 dB attenuation)



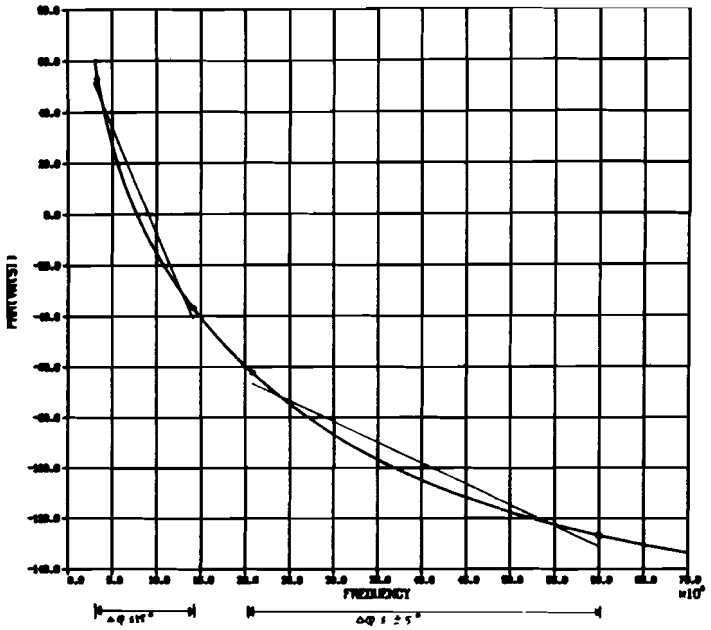
2) 90 MHz output = 7 dBm
(- 3 dBm + 10 dB attenuation)

Appendix 4: Simulations on the hybrids and the IRM

In these simulations the expected phase distortion are given for a LF hybrid with order n = 4 (40 kHz - 600 kHz) and n = 6 (10 kHz - 1 MHz). It is shown that a larger bandwidth is with a certain distortion for lower order networks. Afterwards the expected image rejection is presented for the build image reject mixer.

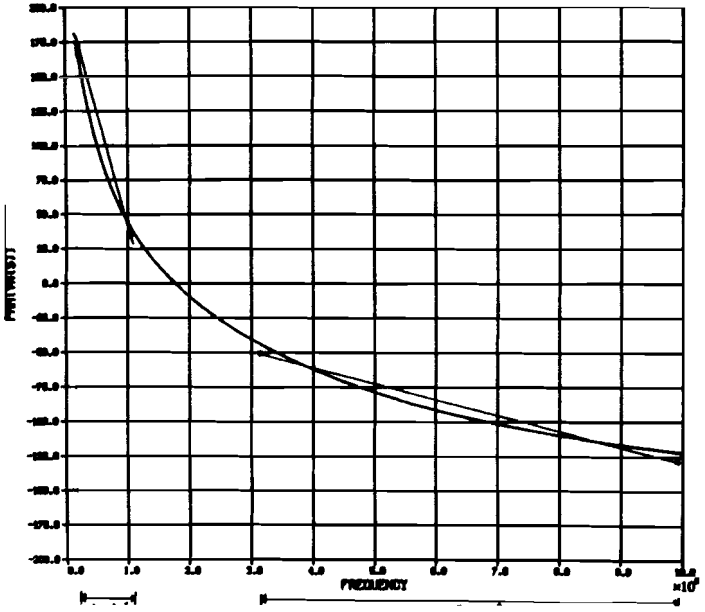
RC HYBRID n=4

Bandwidth with distortion < 5 degree is given.

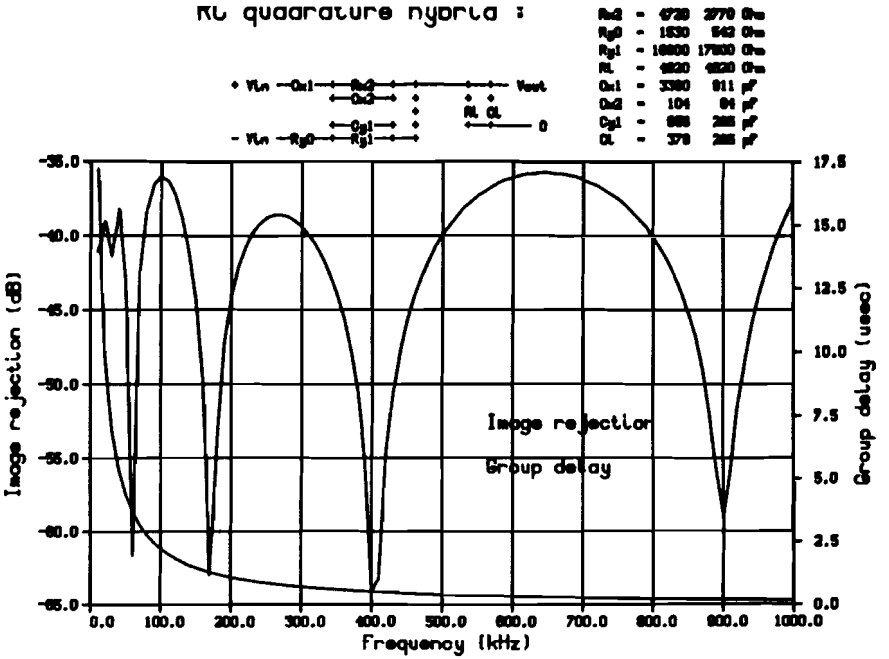


RC HYBRID n=6

Phase of RC hybrid n=6 (LF:10 kHz - 1 MHz) Bandwidth with distortion < 5 degree is given.



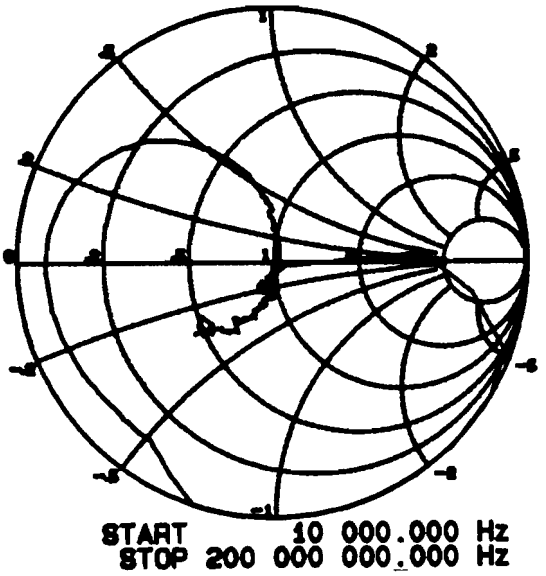
RC quadrature hybrid :

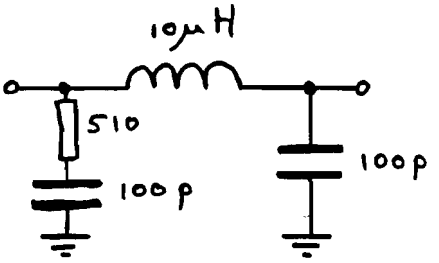


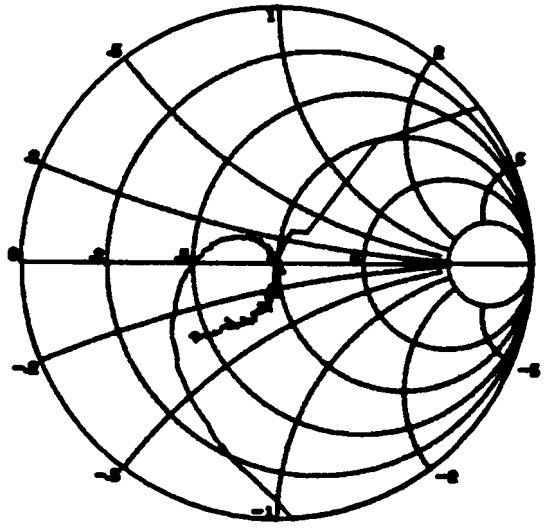
Simulation of image rejection with RC hybrid n=6 (LF: 10 kHz - 1MHz). Also group delay is given.

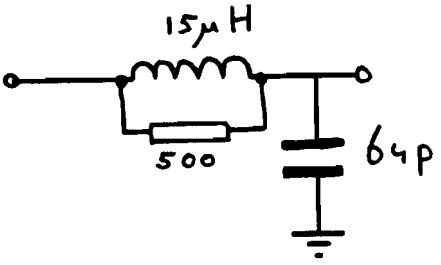
Appendix 5: Filters between mixers and LF hybrid

In order to terminated the used phase detectors (mixers) in the image reject mixer with a suitable impedance (500 Ω), two filters were tested. The imput impedance of the filters connected to the LF hybrids was measured. Both satisfy.









Appendix 6: Measurements on the IRM

Measurement results on the image rejection are presented for various frequencies (10, 21, 50 , 64, 90 MHz). Afterwards the group delay of the LF hybrid is presented in order to estimate phase distortion. Finally the measured intercept of the IRM is given.

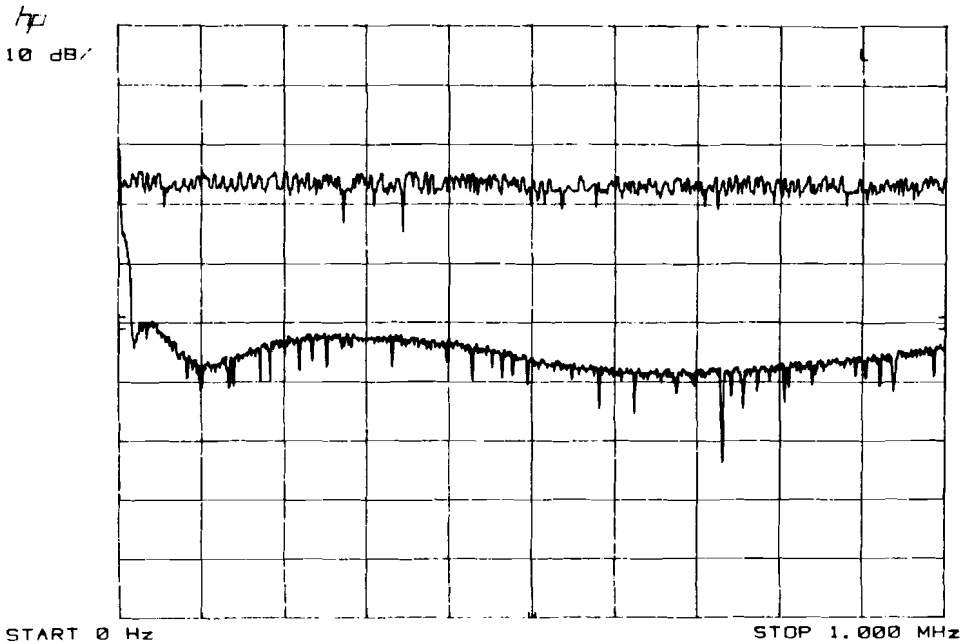


IMAGE REJECTION 50 MHZ

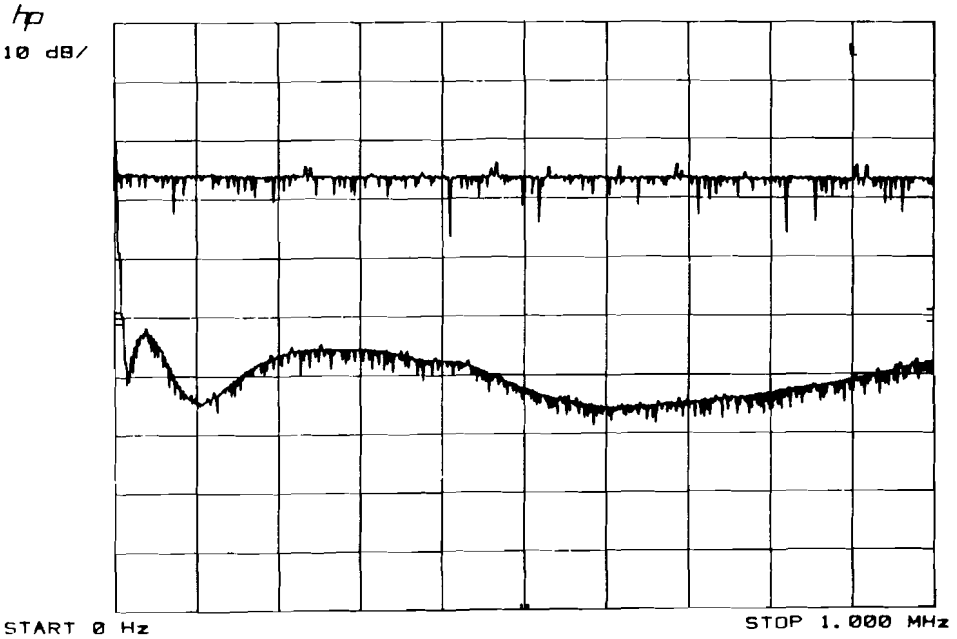


IMAGE REJECTION 21 MHZ

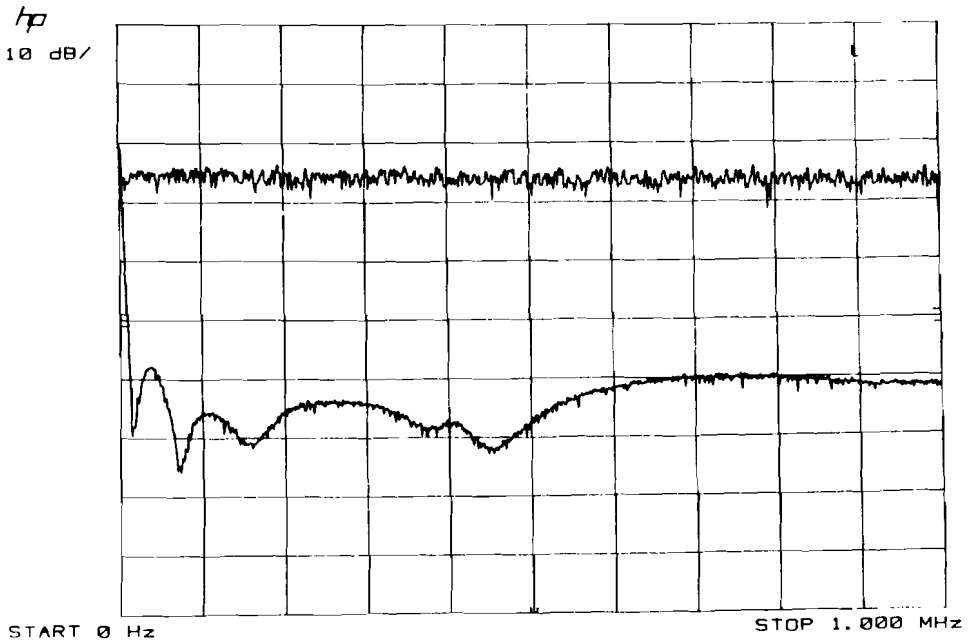
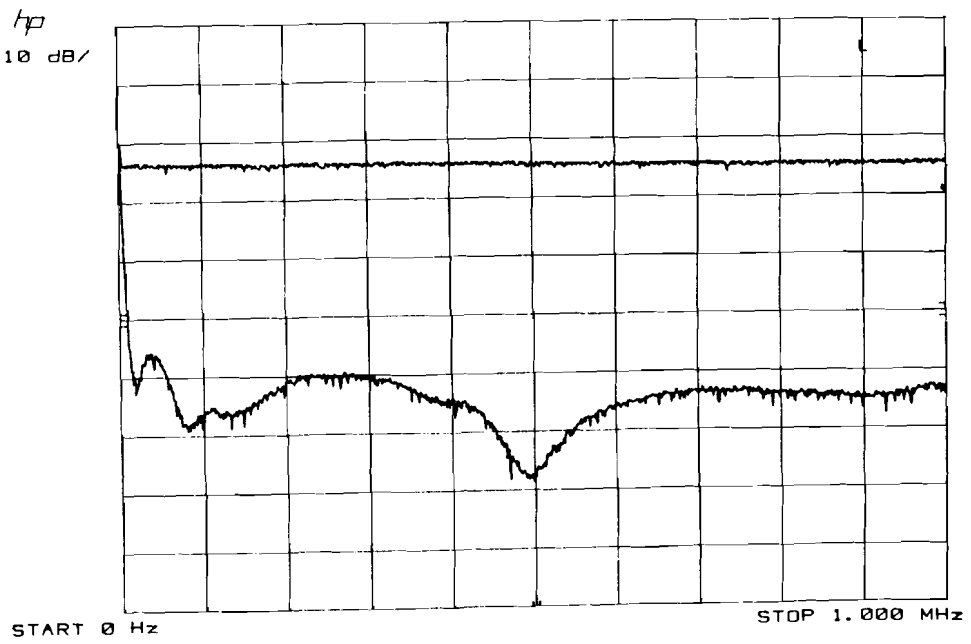
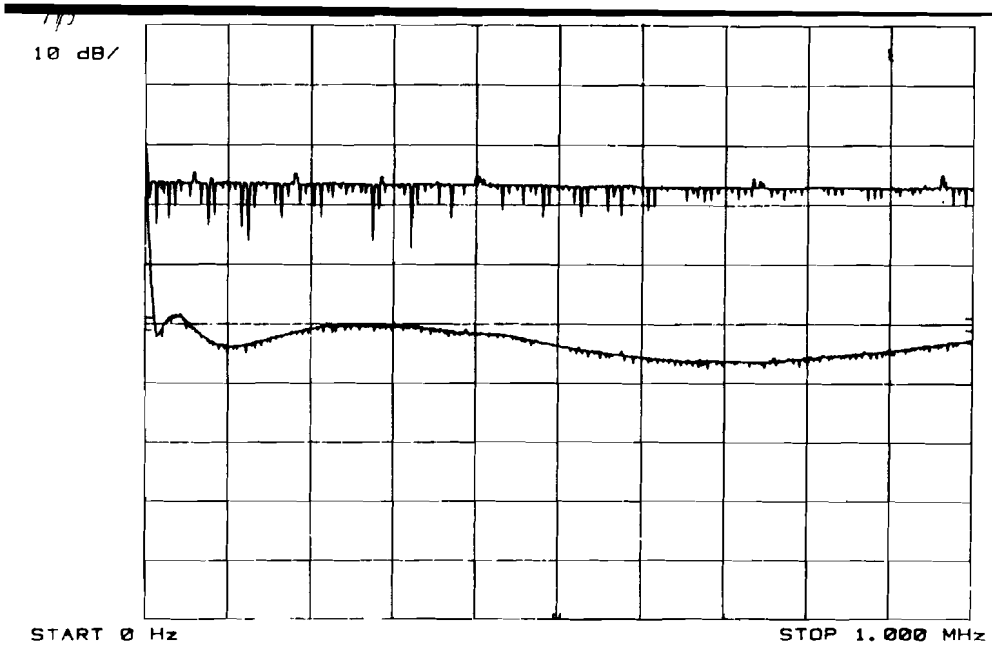
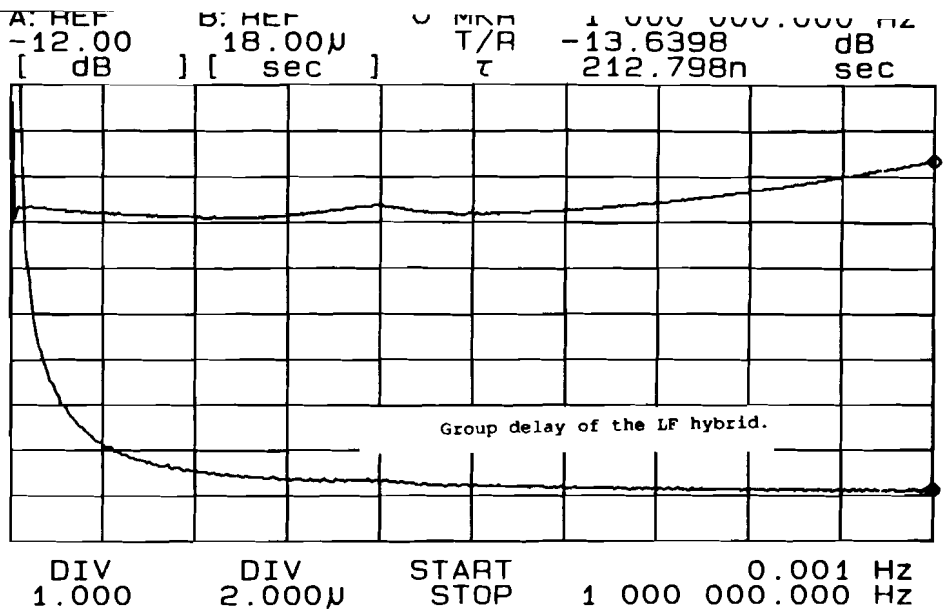
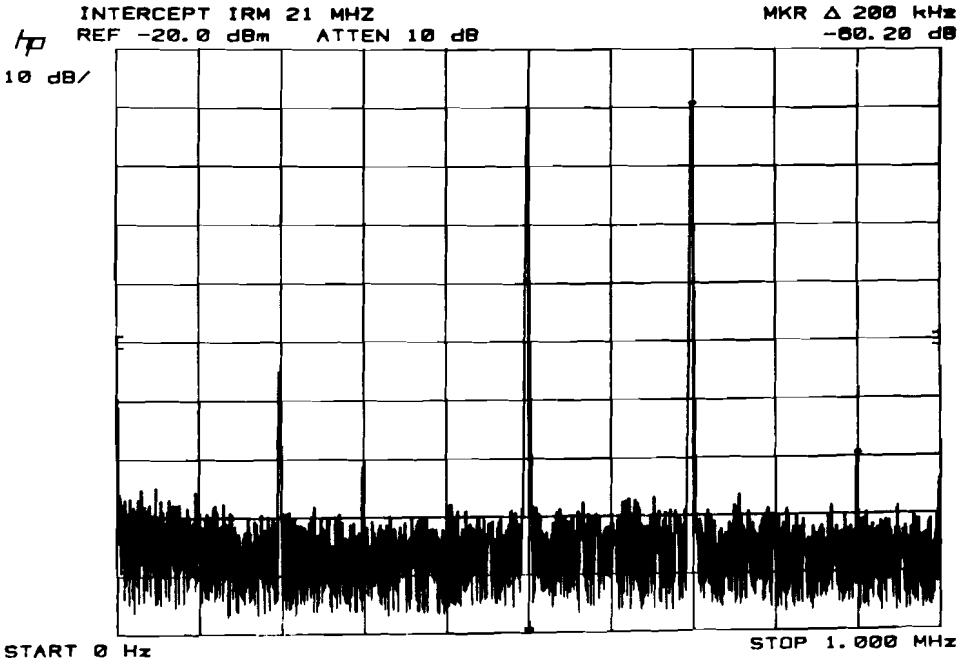


IMAGE REJECTION 64 MHZ



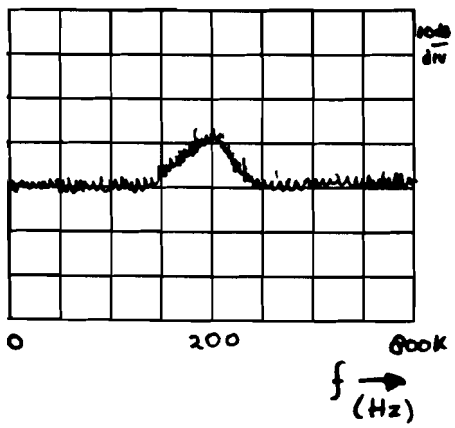

10 dB/
START 0 Hz
STOP 1.000 MHz


A: REF
-12.00
[dB]
B: REF
18.00μ
[sec]
MKR
T/R
τ
-13.6398
dB
212.798n
sec
Group delay of the LF hybrid.
DIV
1.000
DIV
2.000μ
START
STOP
0.001 Hz
1 000 000.000 Hz

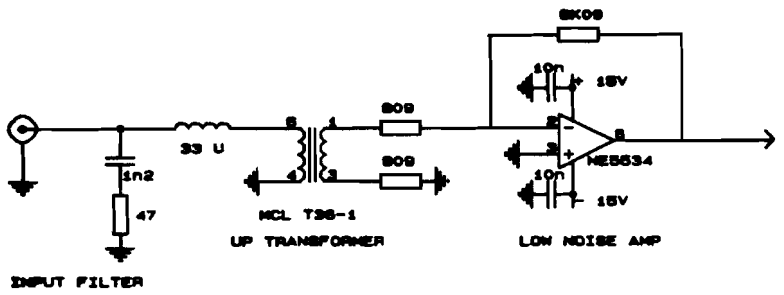

INTERCEPT IRM 21 MHZ
REF -20.0 dBm
ATTEN 10 dB
MKR Δ 200 kHz
-60.20 dB
10 dB/
START 0 Hz
STOP 1.000 MHz

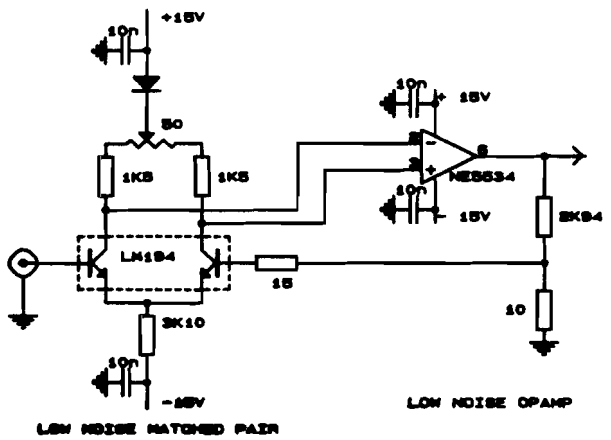
Appendix 7: Noise behavior of the LT1028 and recommendations

Measured noise behavior of the LT 1028 shows deterioration at 200 kHz:



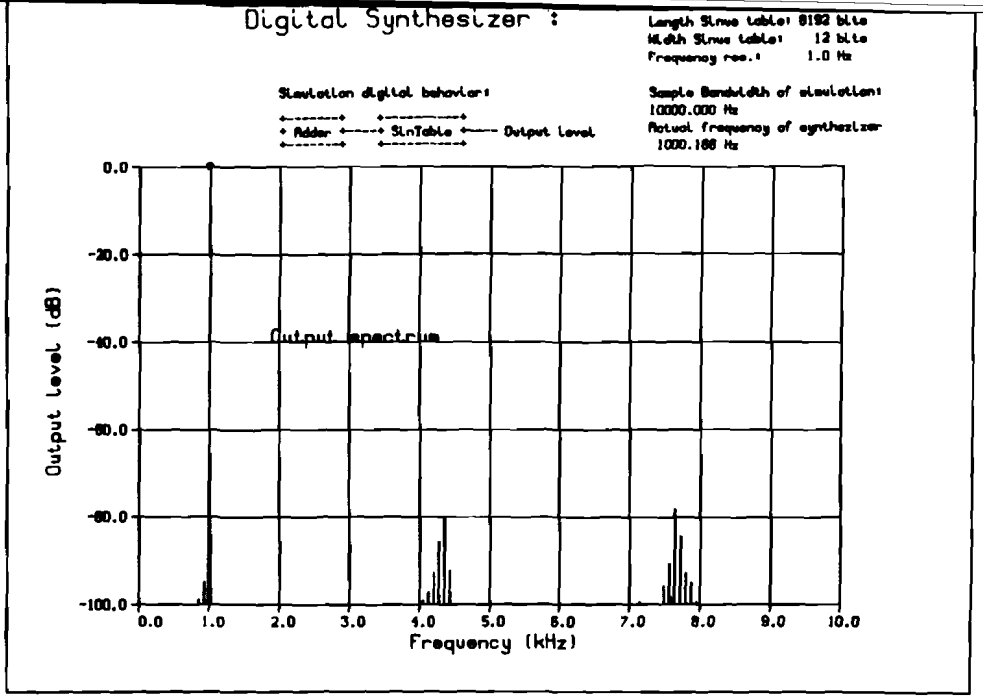
Two ideas to build low noise op amps:

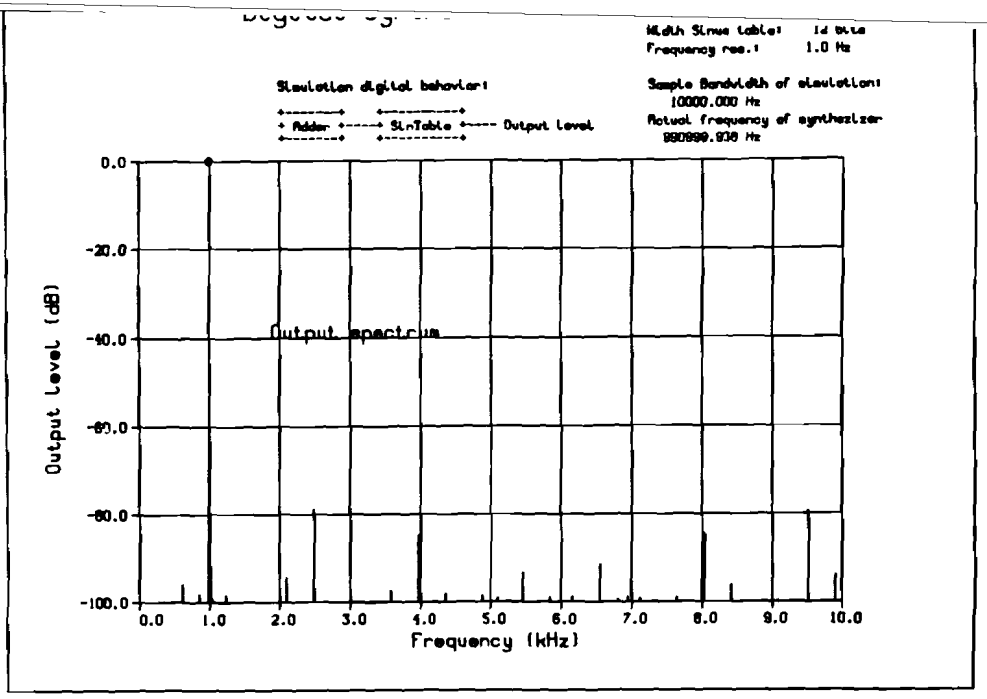


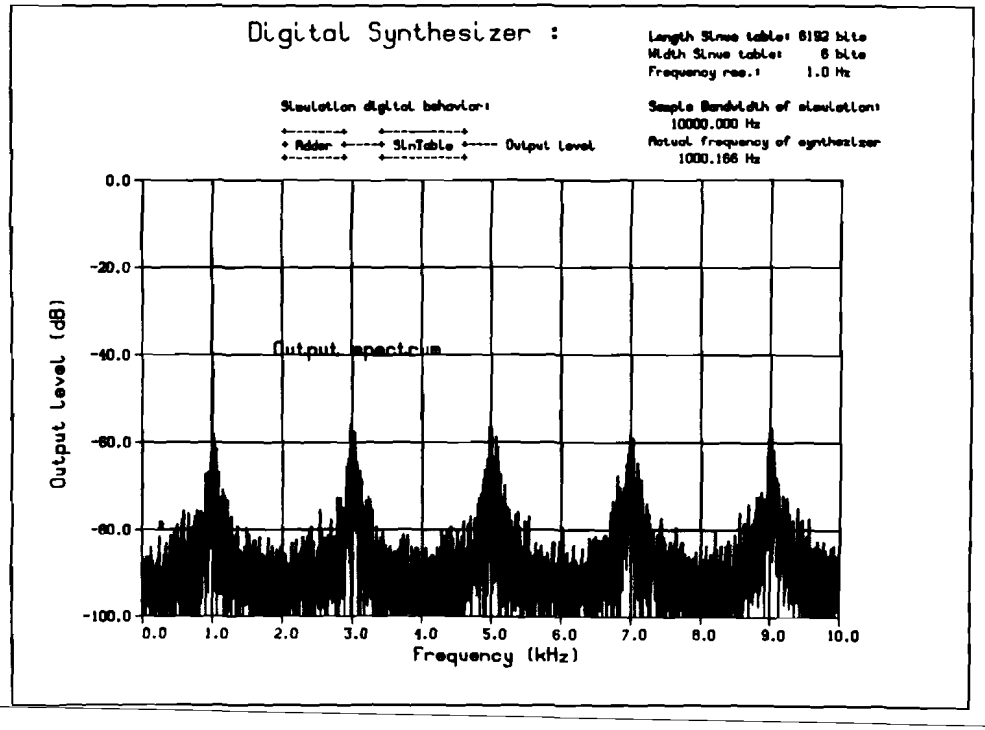


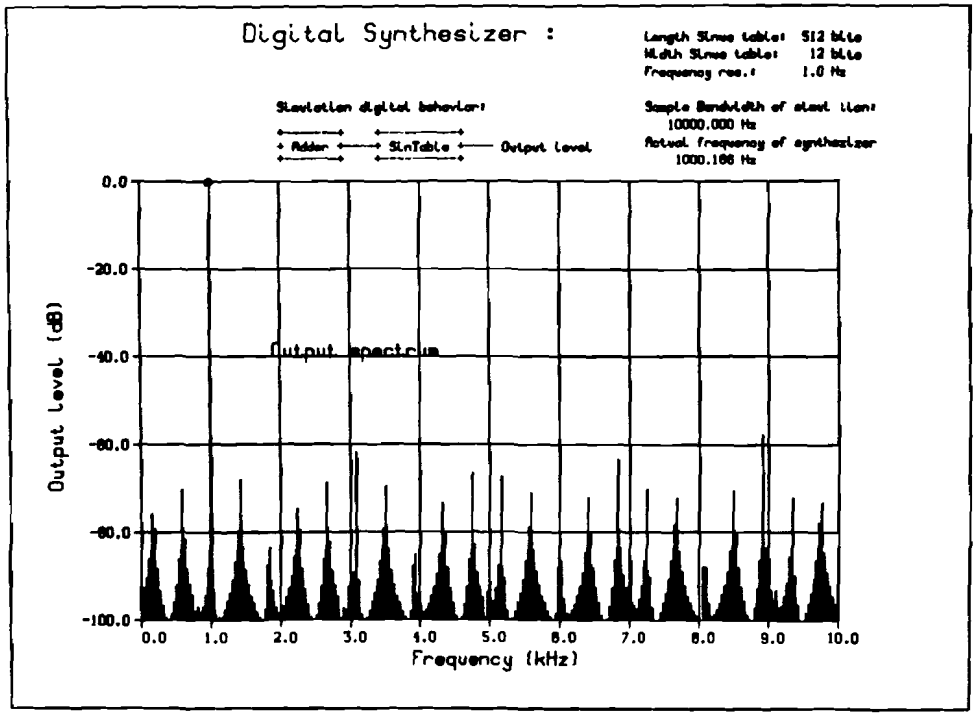
Appendix 8: Simulations on the synthesizer

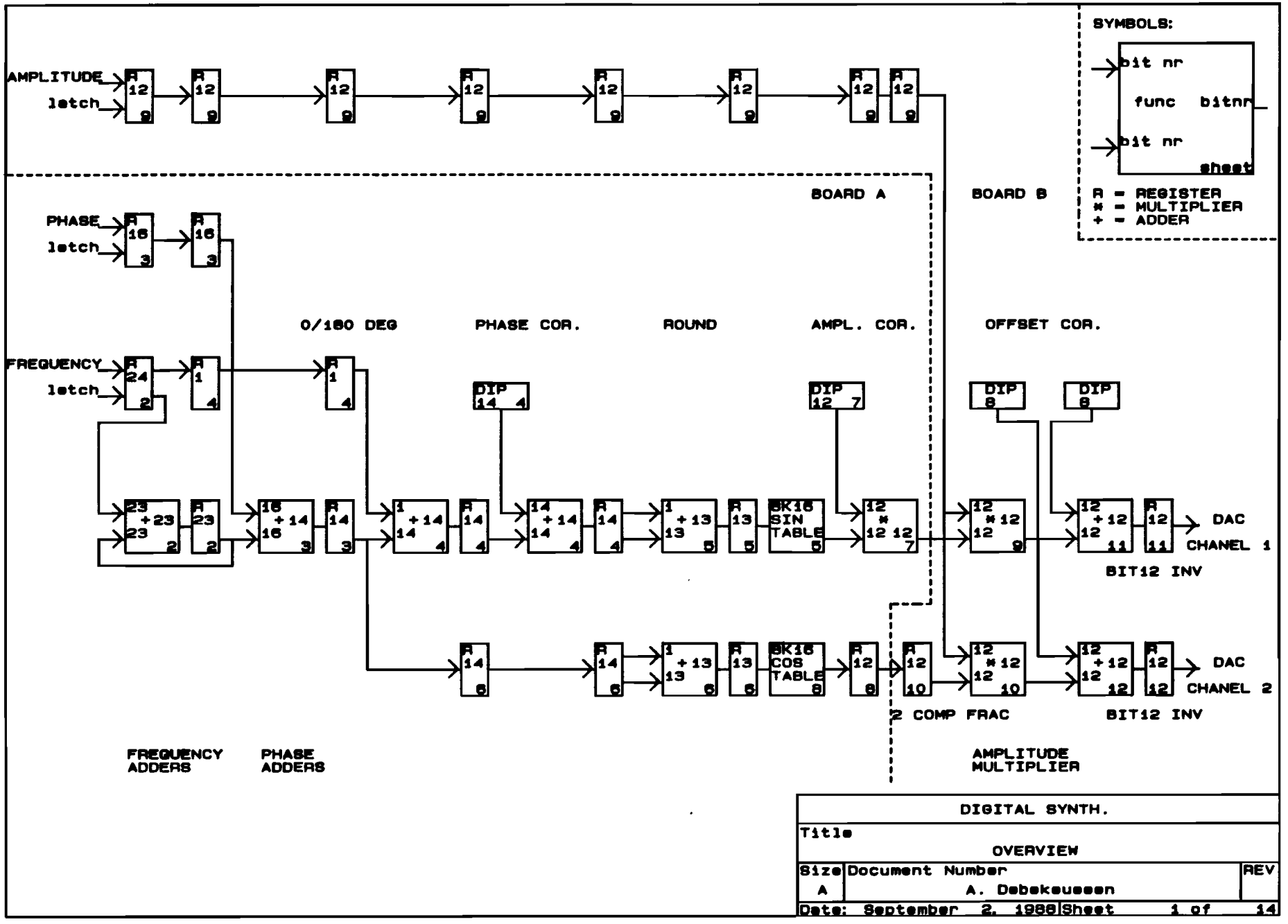
Simulations were carried out to predict the spurious level caused by limited ROM table length and number of DAC bits.

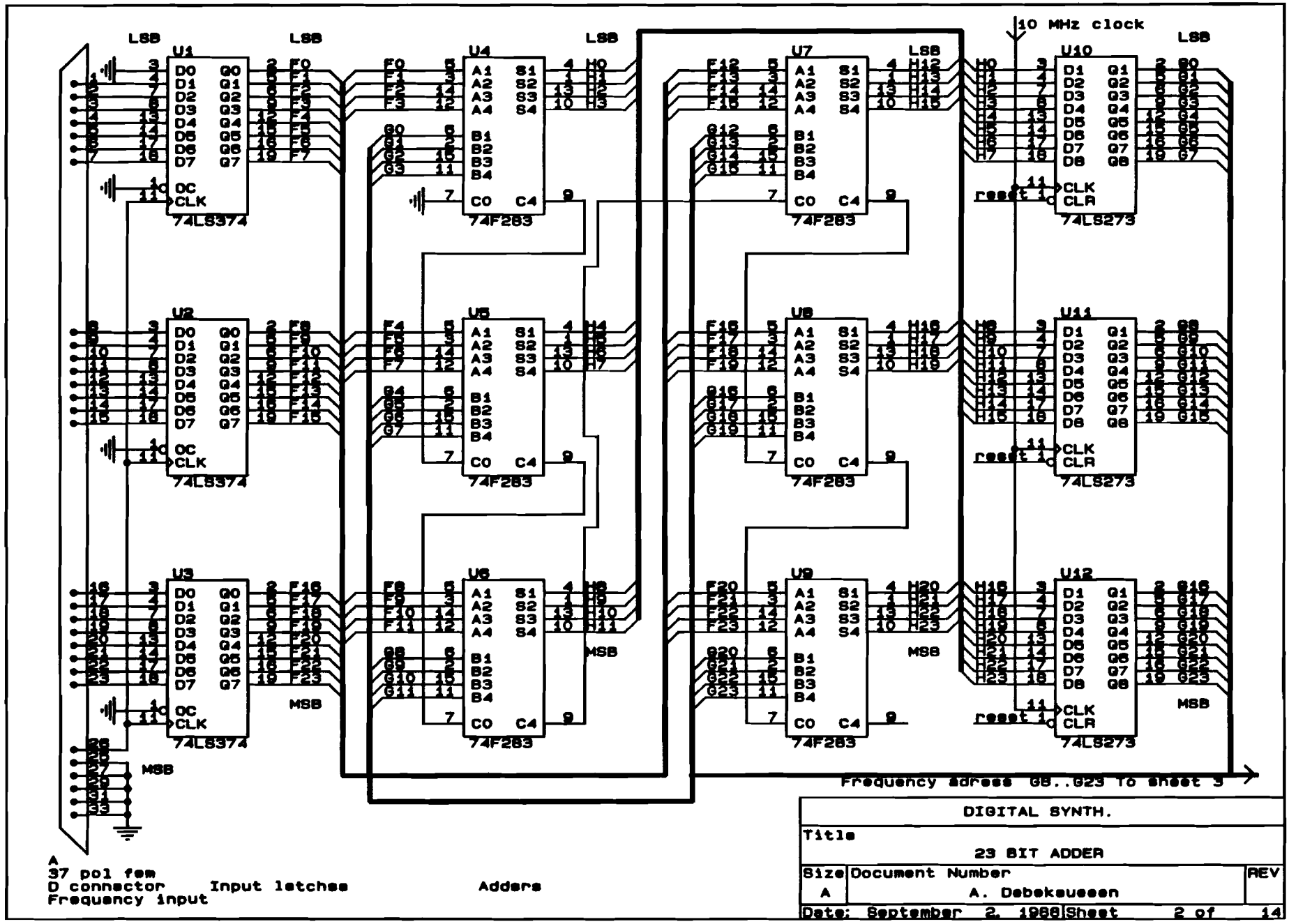

Digital Synthesizer :
Length Sinus table: 8192 bits
Width Sinus table: 12 bits
Frequency res.: 1.0 Hz
Simulation digital behavior:
Adder
SinTable
Output Level
Sample Bandwidth of simulation:
10000.000 Hz
Actual frequency of synthesizer
1000.166 Hz
Output spectrum
Output Level (dB)
Frequency (kHz)

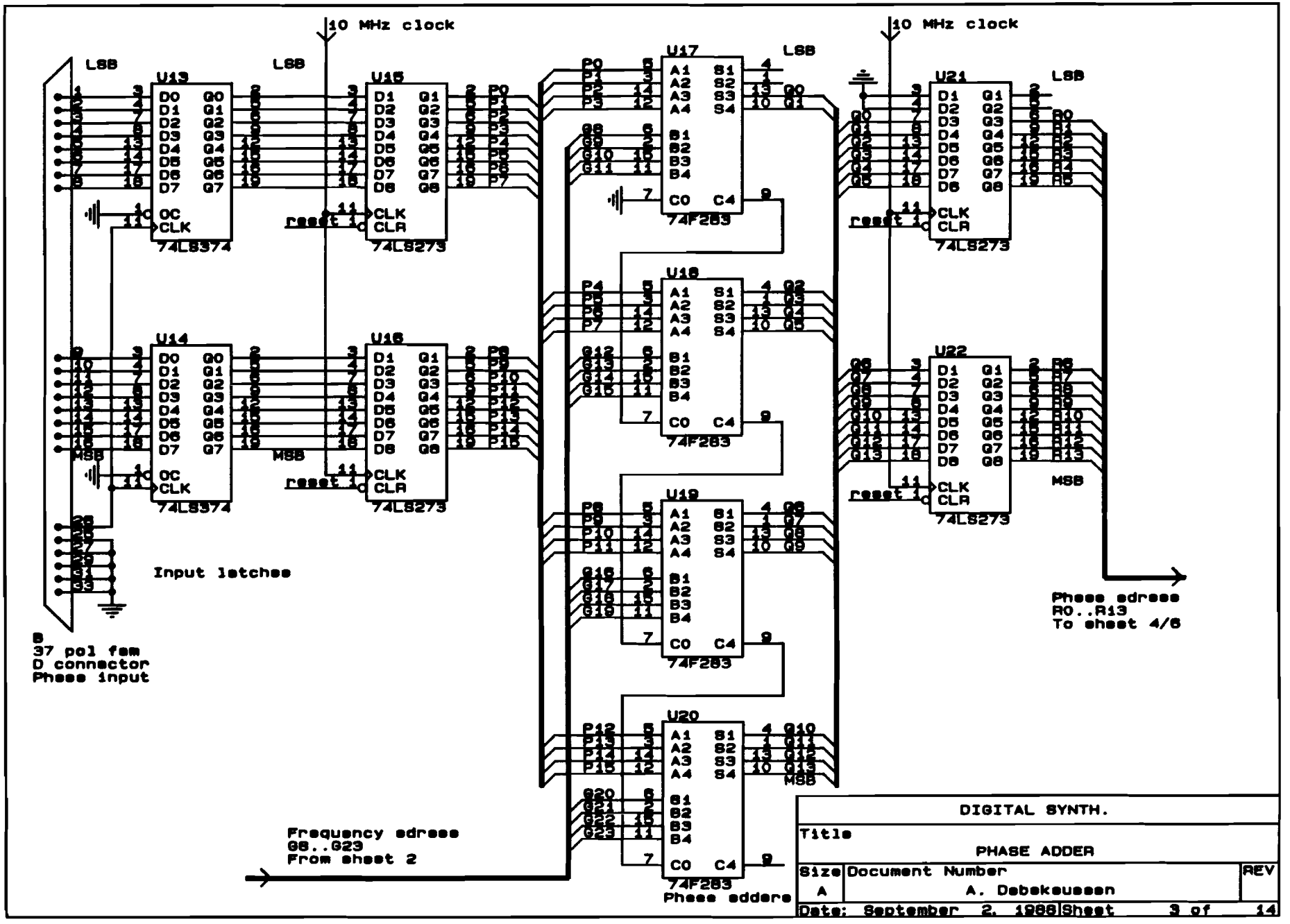

Width Sinus table: 12 bits
Frequency res.: 1.0 Hz
Simulation digital behavior:
Adder
SinTable
Output Level
Sample Bandwidth of simulation:
10000.000 Hz
Actual frequency of synthesizer
990999.938 Hz
Output spectrum
Output Level (dB)
Frequency (kHz)

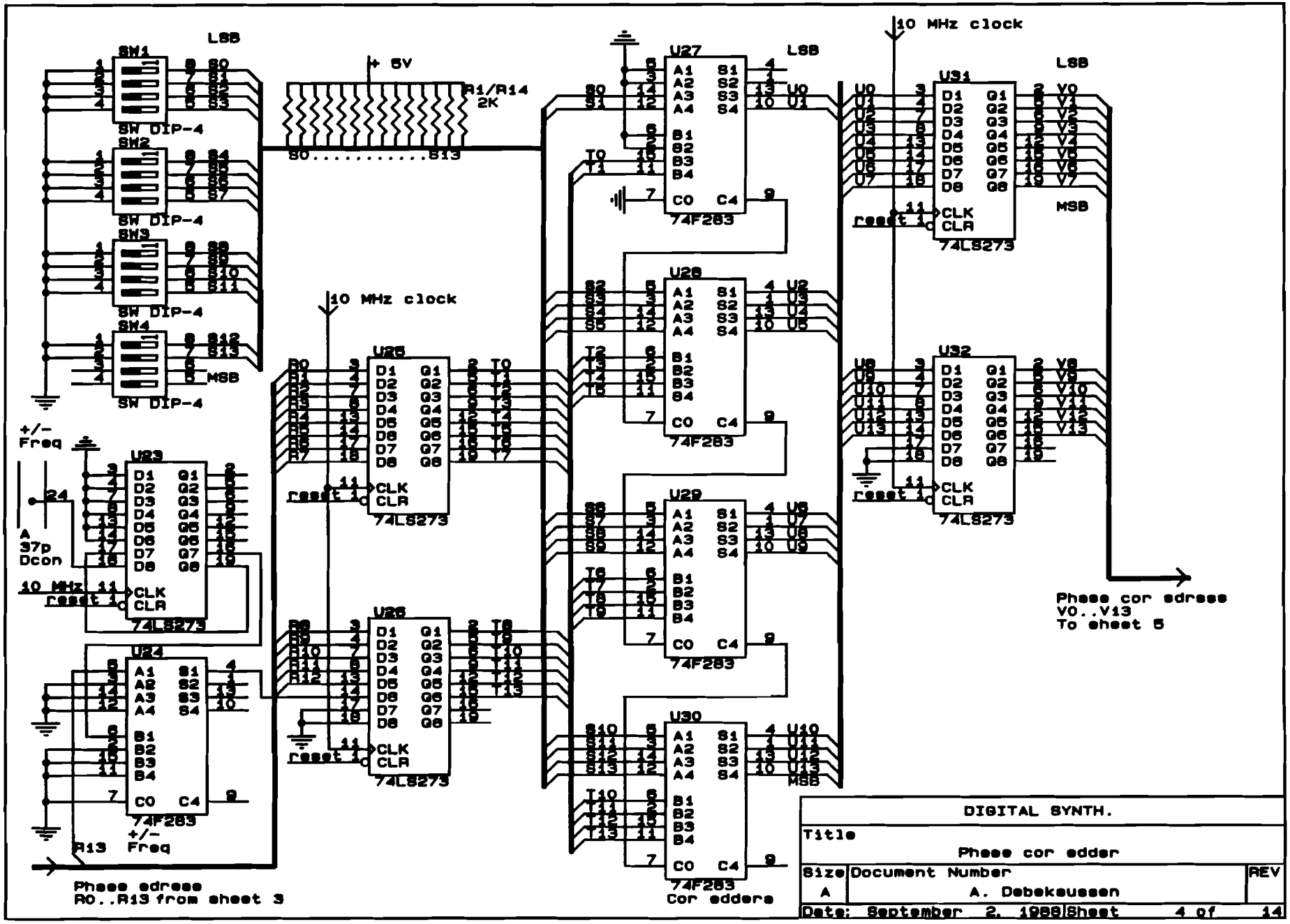

Digital Synthesizer :
Length Sinus table: 8192 bits
Width Sinus table: 8 bits
Frequency res.: 1.0 Hz
Simulation digital behavior:
Adder
SinTable
Output Level
Sample Bandwidth of simulation:
10000.000 Hz
Actual frequency of synthesizer
1000.166 Hz
Output spectrum
Output Level (dB)
Frequency (kHz)

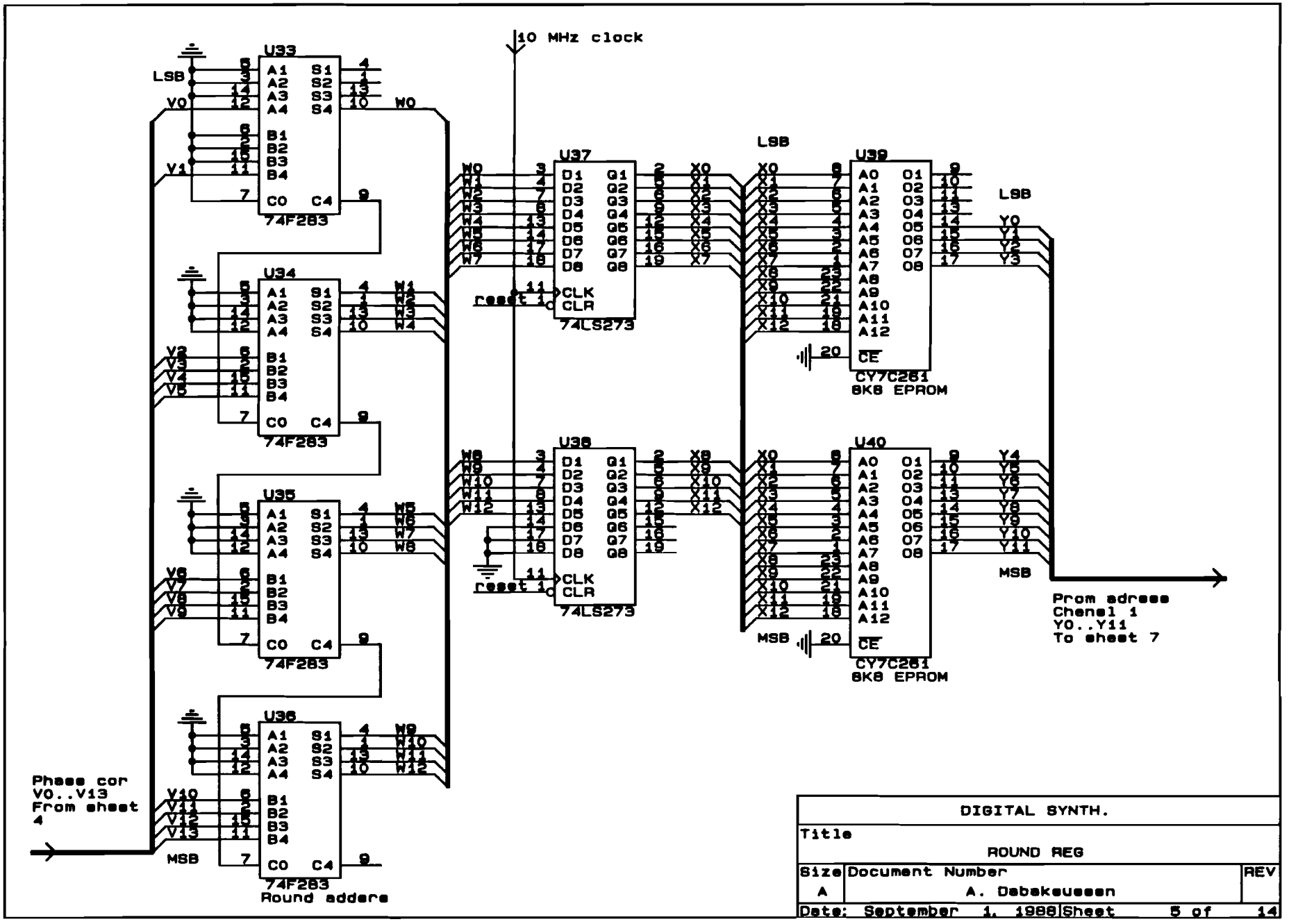

Digital Synthesizer :
Length Sinus table: 512 bits
Width Sinus table: 12 bits
Frequency res.: 1.0 Hz
Simulation digital behavior:
Adder
SinTable
Output Level
Sample Bandwidth of simulation:
10000.000 Hz
Actual frequency of synthesizer
1000.166 Hz
Output spectrum
Output Level (dB)
Frequency (kHz)

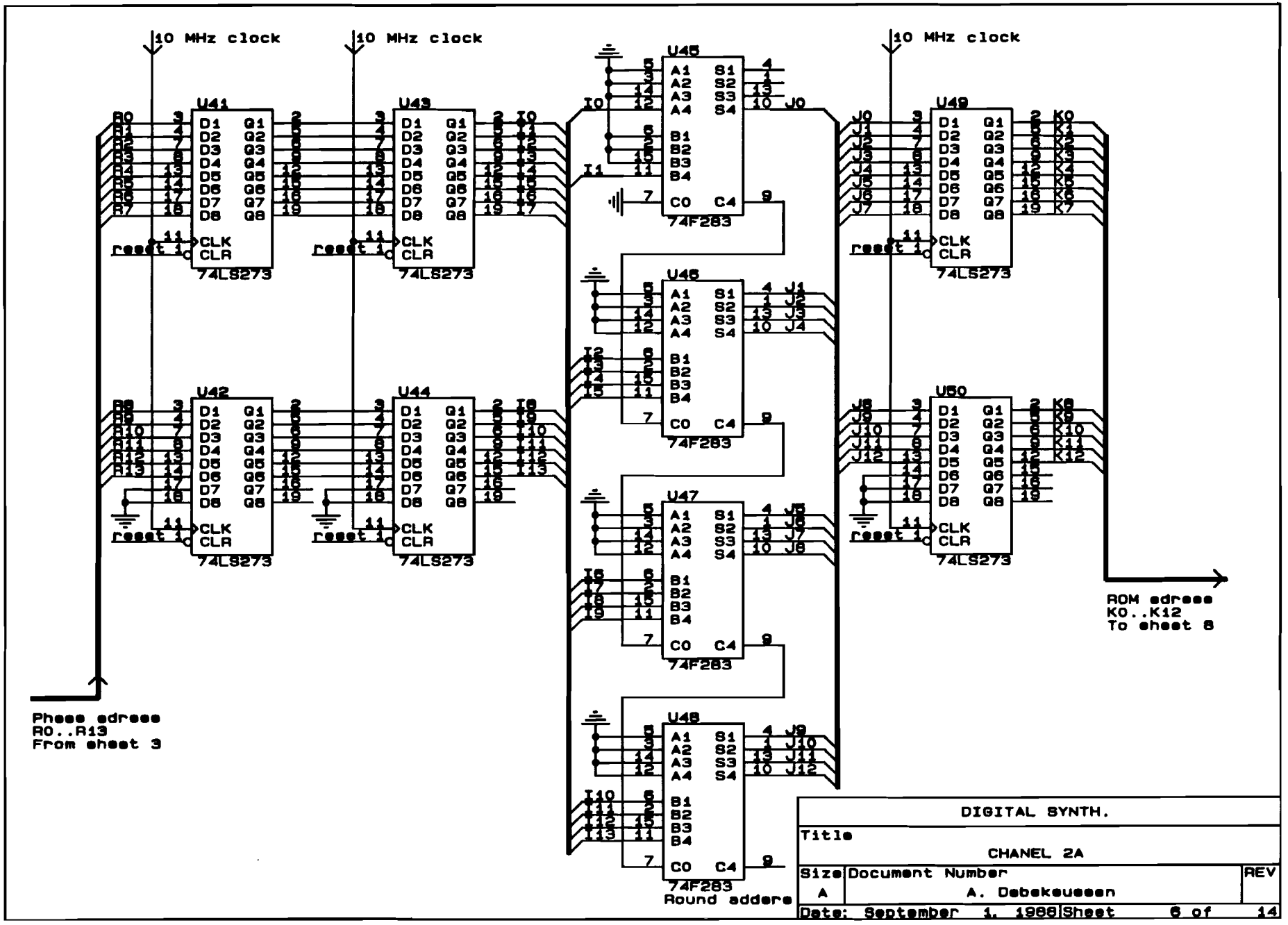

SYMBOLS:
bit nr
func bitnr
bit nr
sheet
R = REGISTER
* = MULTIPLIER
+ = ADDER
BOARD A
BOARD B
AMPLITUDE
latch
PHASE
latch
FREQUENCY
latch
0/180 DEG
PHASE COR.
ROUND
AMPL. COR.
OFFSET COR.
DIP 14 4
DIP 12 7
DIP 8
DIP 8
SIN TABLE
COS TABLE
DAC
CHANEL 1
BIT12 INV
DAC
CHANEL 2
BIT12 INV
2 COMP FRAC
FREQUENCY ADDERS
PHASE ADDERS
AMPLITUDE MULTIPLIER
DIGITAL SYNTH.
Title
OVERVIEW
Size
A
Document Number
A. Debekeuseen
REV
Date: September 2, 1988
Sheet 1 of 14

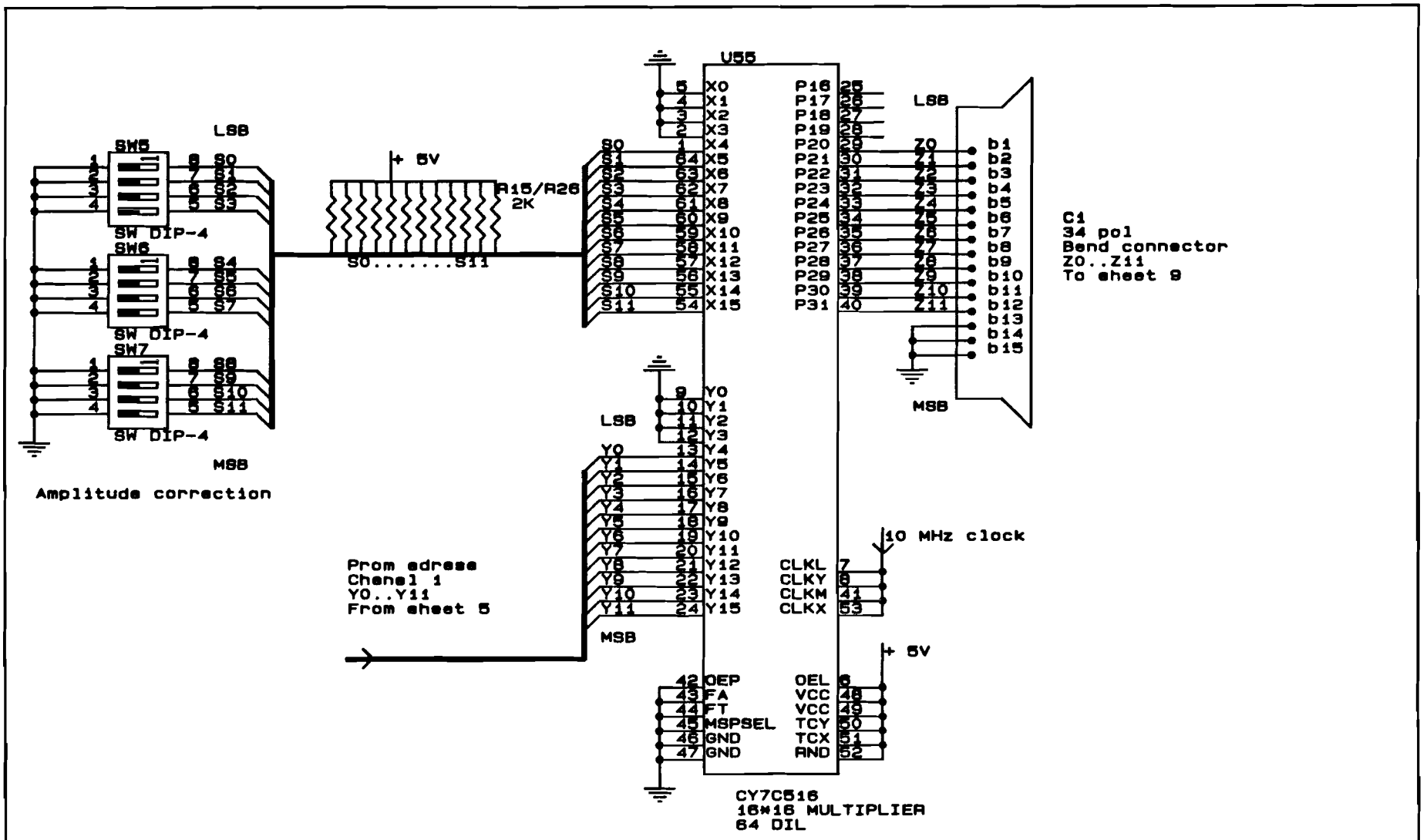

10 MHz clock
LSB
MSB
U1
U2
U3
U4
U5
U6
U7
U8
U9
U10
U11
U12
74LS374
74F283
74LS273
reset 1
CLK
CLR
OC
Frequency adress G8..G23 to sheet 3
A
37 pol fem
D connector
Input latches
Adders
Frequency input
DIGITAL SYNTH.
Title
23 BIT ADDER
Size
A
Document Number
A. Debeksussen
REV
Date: September 2, 1988
Sheet 2 of 14


10 MHz clock
LSB
MSB
U13
U14
U15
U16
U17
U18
U19
U20
U21
U22
74LS374
74LS273
74F283
Input latches
P
37 pol fem
D connector
Phase input
Frequency adresse
G8..G23
From sheet 2
Phase adders
Phase adresse
R0..R13
To sheet 4/6
DIGITAL SYNTH.
Title
PHASE ADDER
Size
A
Document Number
A. Debebeussen
REV
Date: September 2, 1988
Sheet 3 of 14

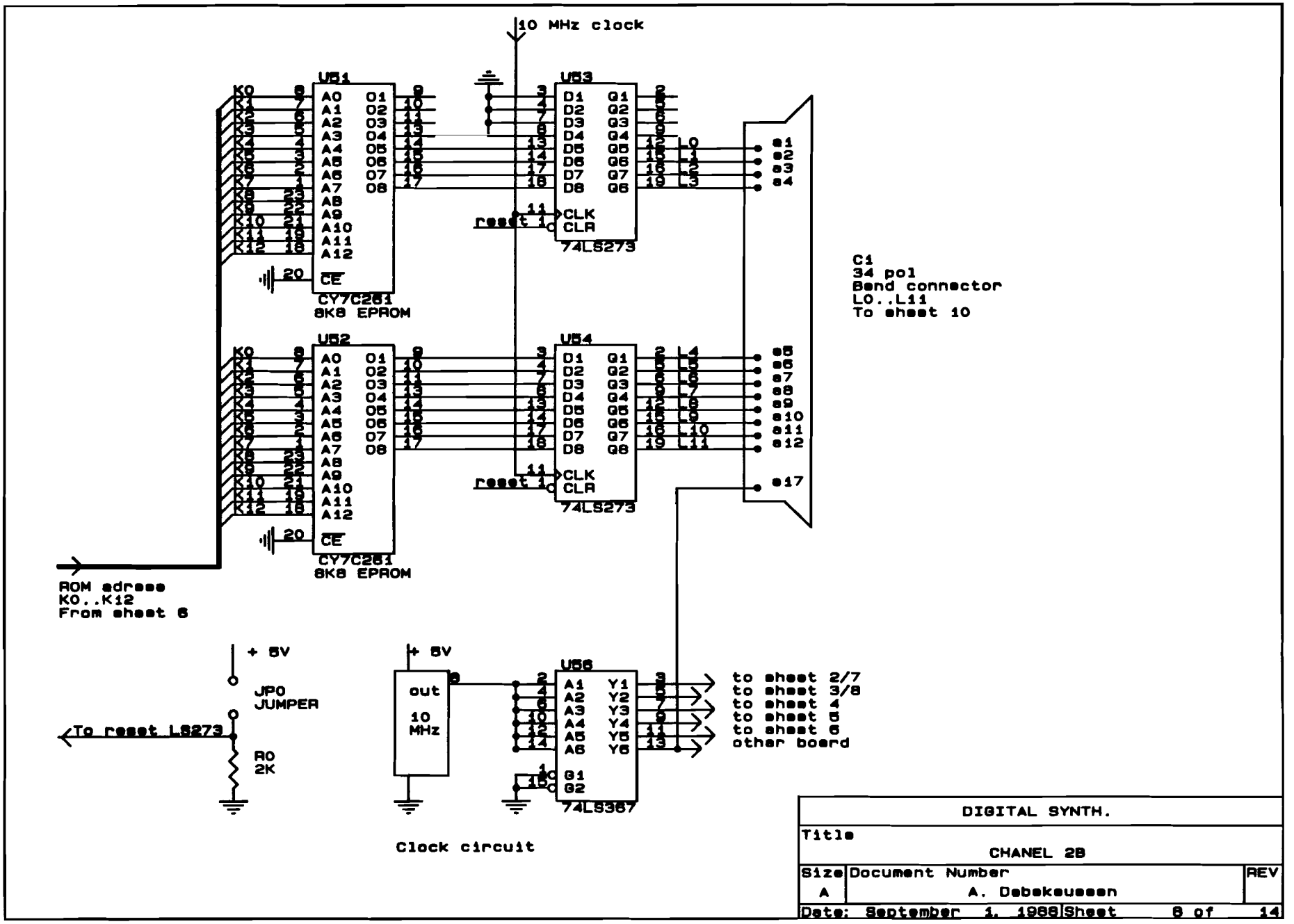

LSB
SW1
SW DIP-4
SW2
SW DIP-4
SW3
SW DIP-4
SW4
SW DIP-4
MSB
+ 5V
R1/R14
2K
S0.....S13
+/-
Freq
24
A
37p
Dcon
10 MHz
reset
U23
U24
74F283
+/-
Freq
R13
Phase adress
R0..R13 from sheet 3
10 MHz clock
U25
U26
74LS273
U27
U28
U29
U30
74F283
Cor adders
U31
U32
74LS273
Phase cor adress
V0..V13
To sheet 5
DIGITAL SYNTH.
Title
Phase cor adder
Size
A
Document Number
A. Debekeussen
REV
Date: September 2, 1988
Sheet 4 of 14

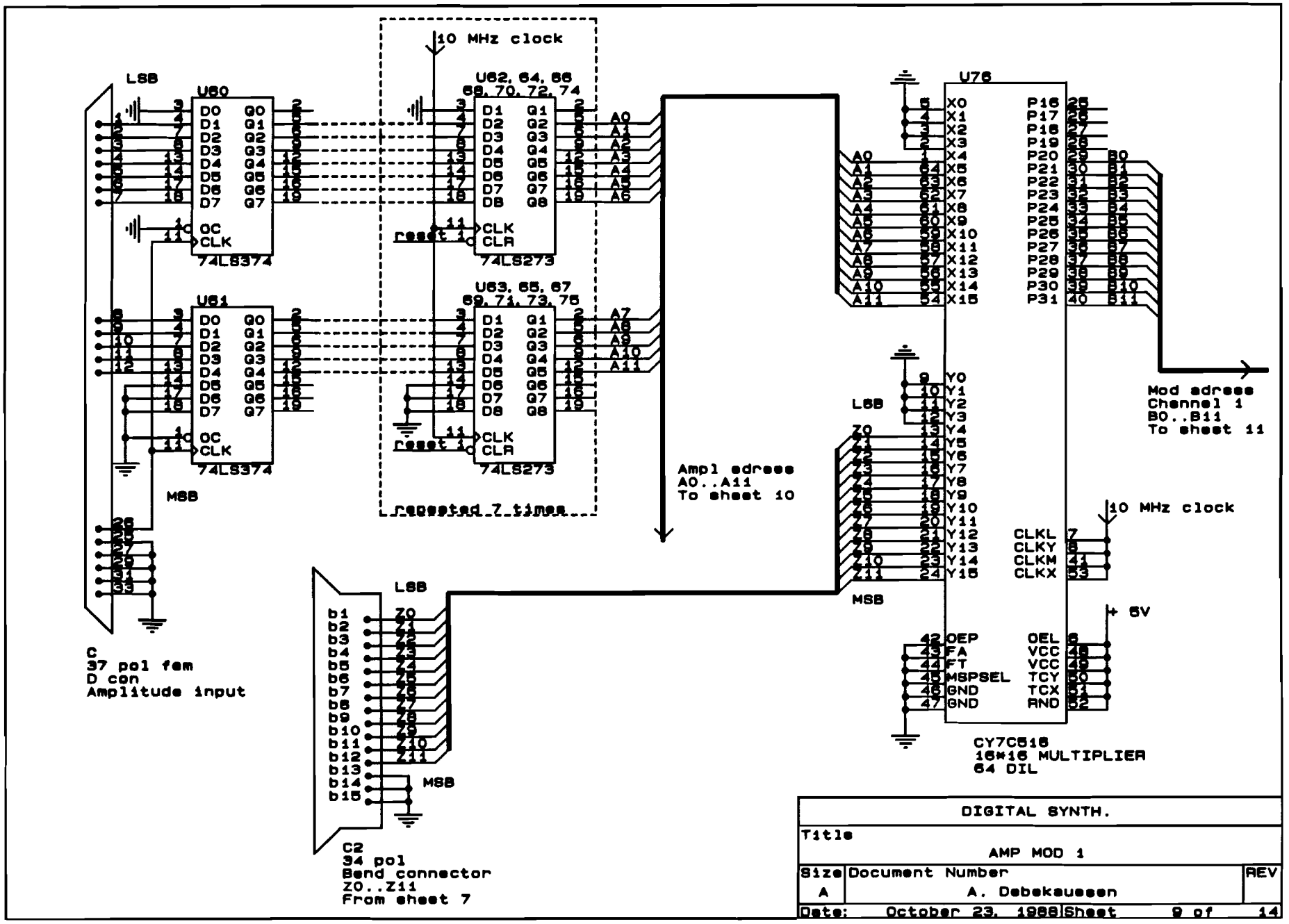

10 MHz clock
U33
U34
U35
U36
74F283
Round addere
LSB
MSB
V0
V1
V2
V3
V4
V5
V6
V7
V8
V9
V10
V11
V12
V13
W0
W1
W2
W3
W4
W5
W6
W7
W8
W9
W10
W11
W12
Phase cor
V0..V13
From sheet
4
U37
U38
74LS273
reset 1
CLK
CLR
U39
U40
CY7C261
8K8 EPROM
CE
Y0
Y1
Y2
Y3
Y4
Y5
Y6
Y7
Y8
Y9
Y10
Y11
Prom adress
Chenel 1
Y0..Y11
To sheet 7
DIGITAL SYNTH.
Title
ROUND REG
Size
A
Document Number
A. Dabakeuseen
REV
Date: September 1, 1988
Sheet 5 of 14


10 MHz clock
10 MHz clock
10 MHz clock
U41
U42
U43
U44
U45
U46
U47
U48
U49
U50
74LS273
74F283
reset
CLK
CLR
Phase adress
R0..R13
From sheet 3
ROM adress
K0..K12
To sheet 8
Round adders
DIGITAL SYNTH.
Title
CHANEL 2A
Size
A
Document Number
A. Debakeussen
REV
Date: September 1, 1988
Sheet 6 of 14

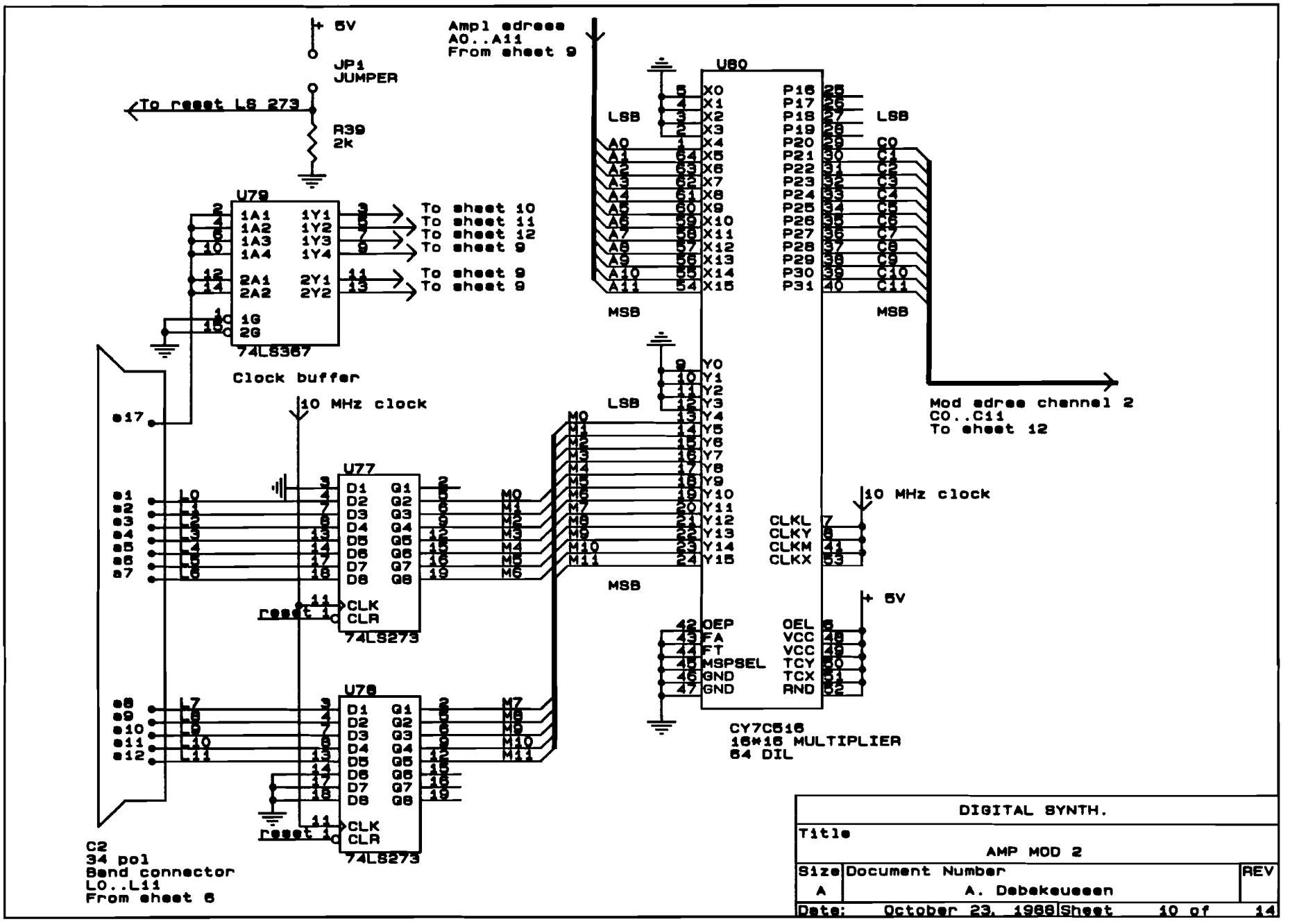


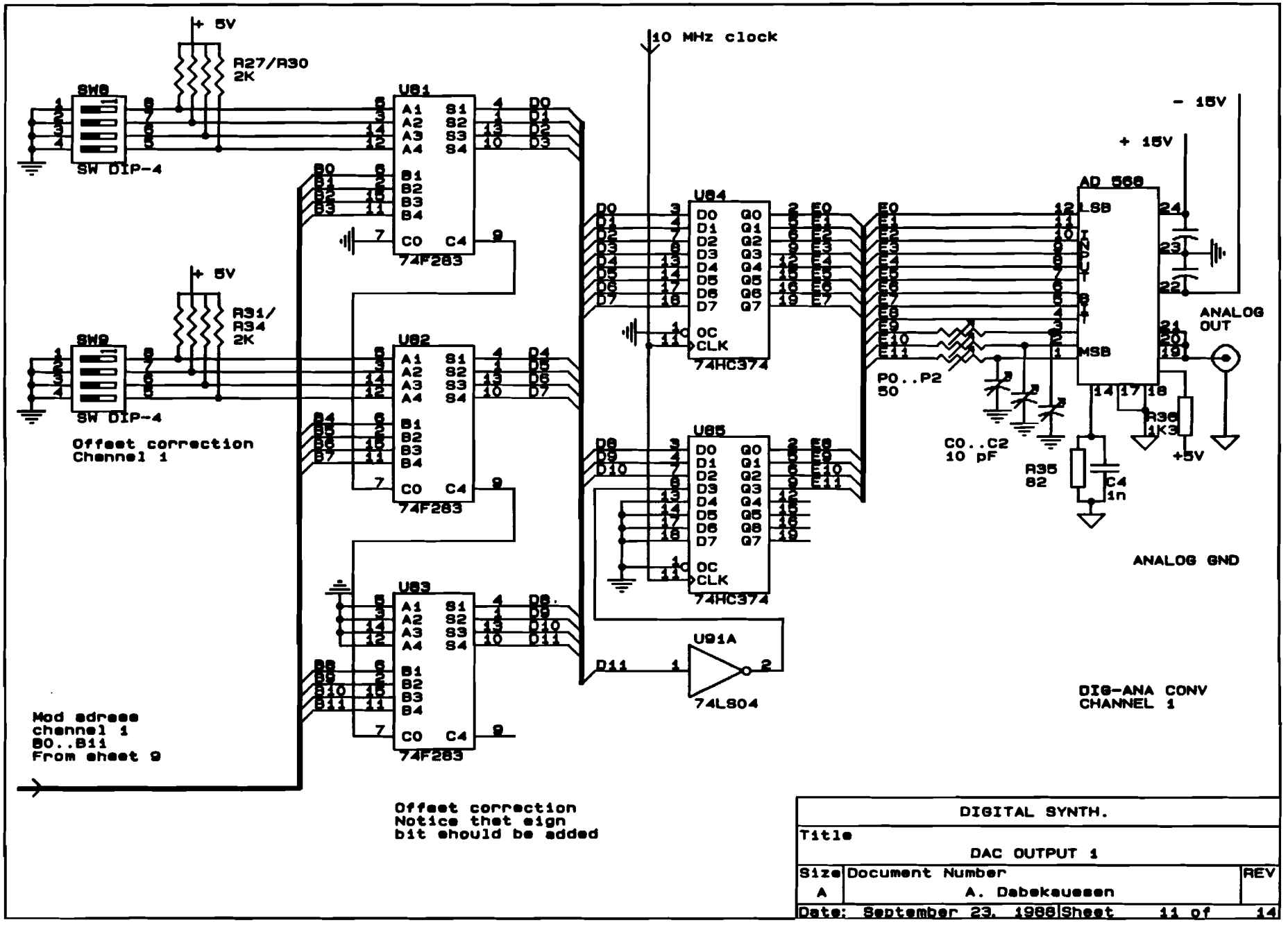
DIGITAL SYNTH.		
Title: AMPLITUDE COR		
Size: A	Document Number: A. Debekeuseen	REV
Date: October 23, 1988	Sheet 7 of 14	

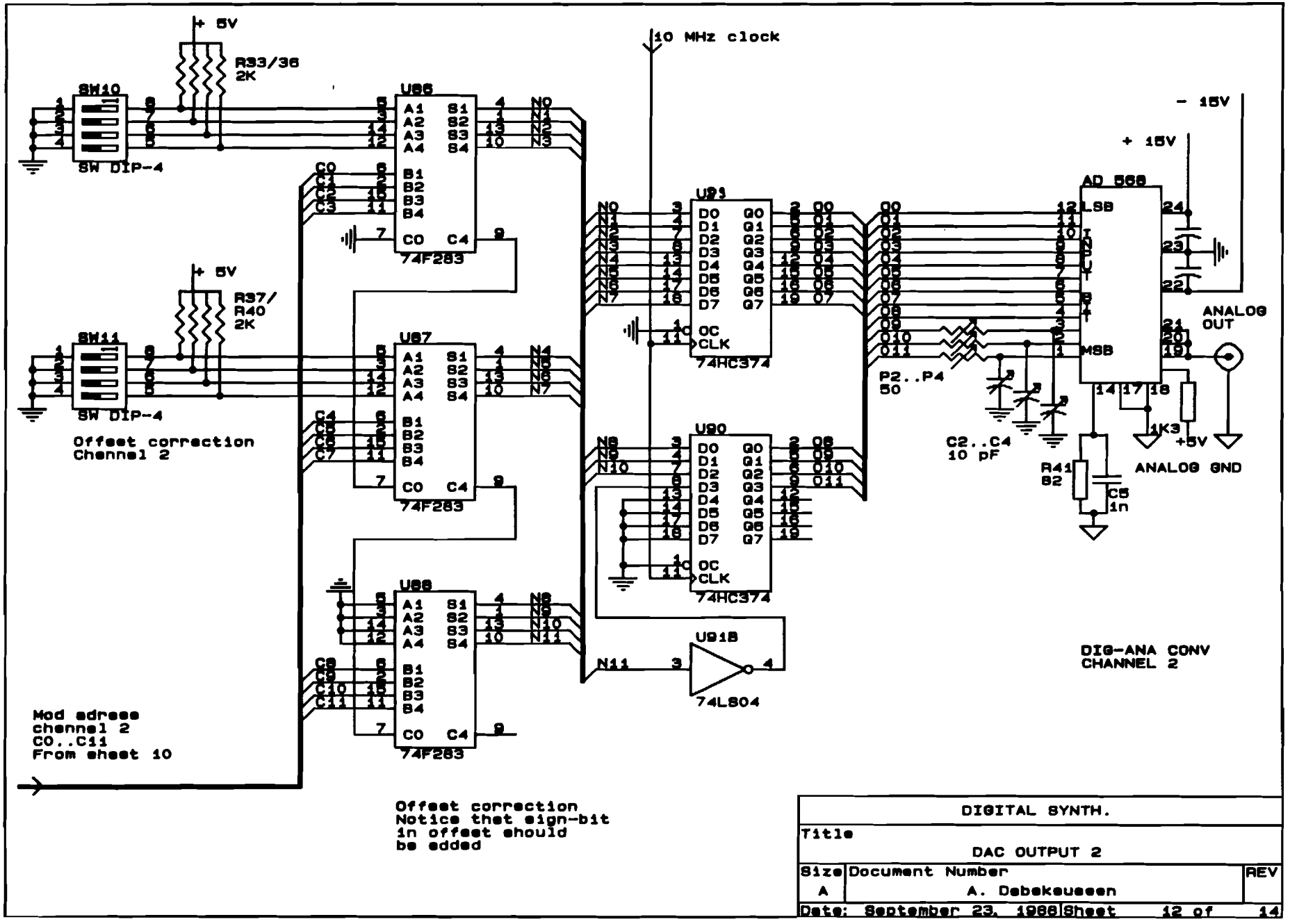

10 MHz clock
U51
U53
U52
U54
CY7C261
8K8 EPROM
74LS273
reset 1
CLK
CLR
CE
C1
34 pol
Bend connector
LO..L11
To sheet 10
ROM adress
K0..K12
From sheet 6
+ 5V
JP0
JUMPER
To reset LS273
R0
2K
out
10
MHz
U56
74LS367
to sheet 2/7
to sheet 3/8
to sheet 4
to sheet 5
to sheet 6
other board
Clock circuit
DIGITAL SYNTH.
Title
CHANEL 2B
Size
A
Document Number
A. Debakeussen
REV
Date: September 1, 1988
Sheet 8 of 14

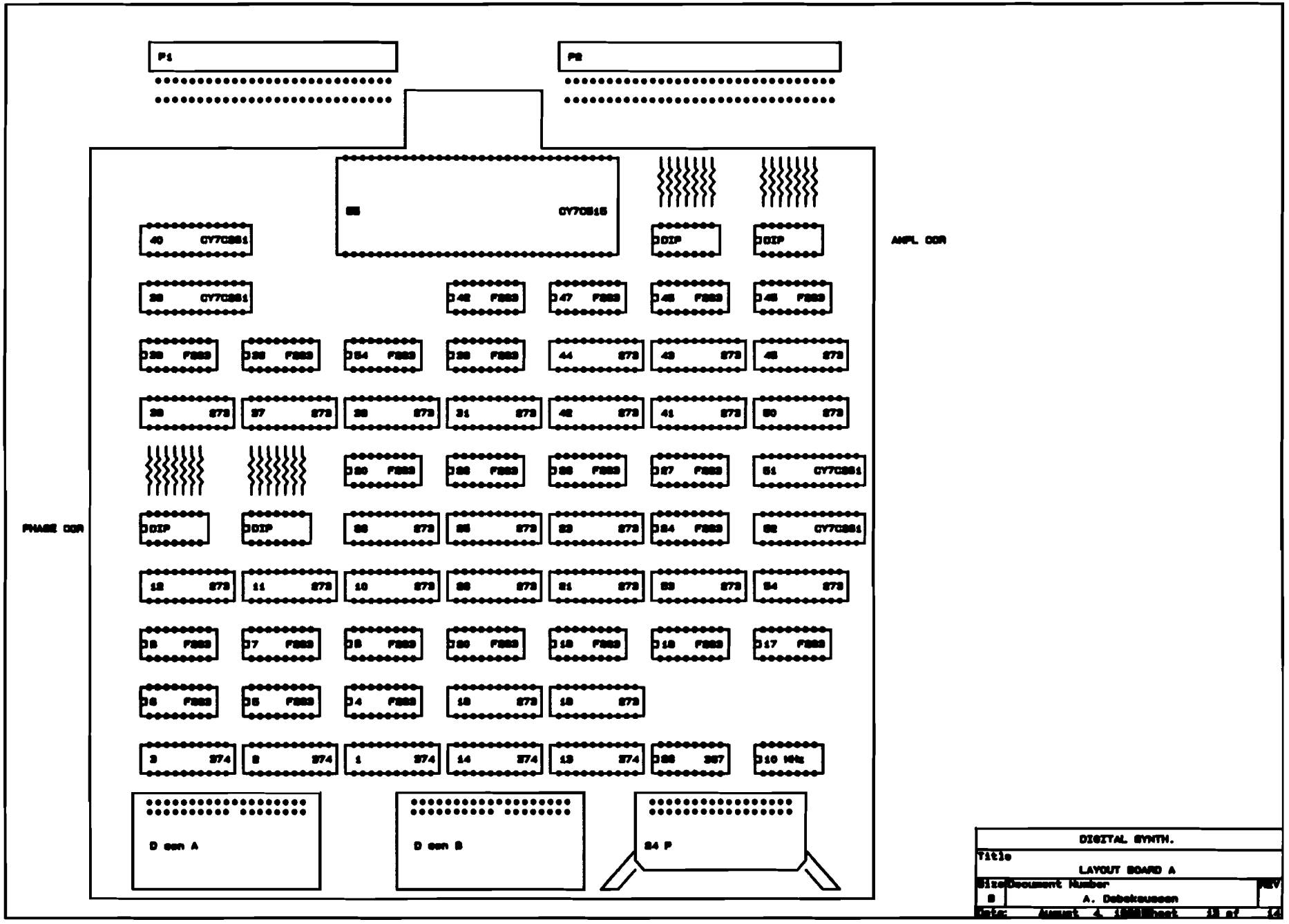


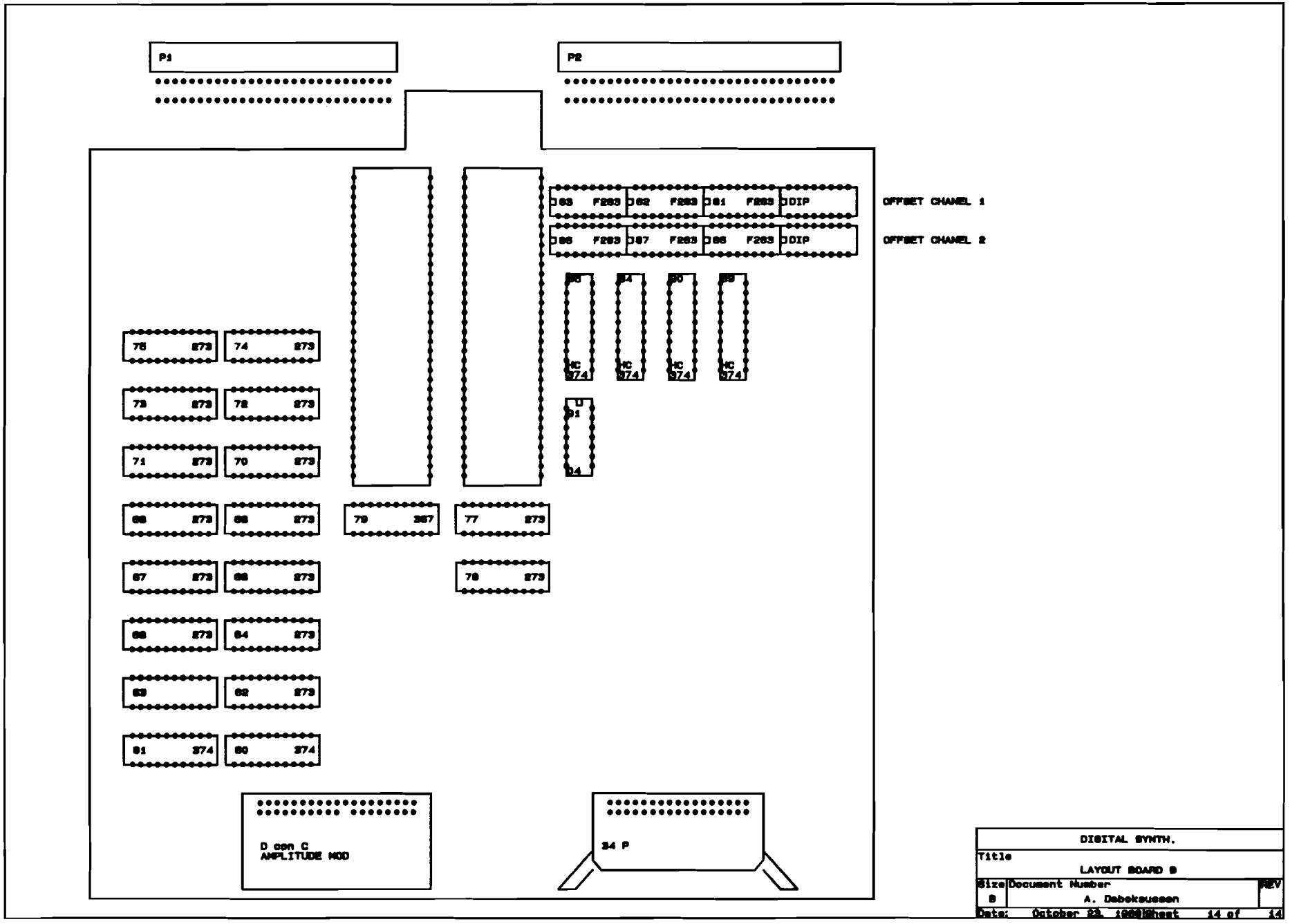
DIGITAL SYNTH.		
Title: AMP MOD 1		
Size: A	Document Number: A. Debeksussen	REV
Date: October 23, 1988	Sheet 9 of 14	


+ 5V
JP1
JUMPER
To reset LS 273
R39
2k
Ampl edrese
A0..A11
From sheet 9
U79
74LS367
Clock buffer
To sheet 10
To sheet 11
To sheet 12
To sheet 9
To sheet 9
To sheet 9
10 MHz clock
U80
LSB
MSB
Mod edrse channel 2
C0..C11
To sheet 12
U77
U78
74LS273
reset 1
C2
34 pol
Bend connector
L0..L11
From sheet 6
CY7C516
16*16 MULTIPLIER
64 DIL
DIGITAL SYNTH.
Title
AMP MOD 2
Size
A
Document Number
A. Debekeuseen
REV
Date: October 23, 1988
Sheet 10 of 14


+ 5V
R27/R30
2K
SW0
SW DIP-4
U81
74F283
10 MHz clock
- 15V
+ 15V
AD 568
LSB
MSB
ANALOG
OUT
U84
74HC374
+ 5V
R31/
R34
2K
SW0
SW DIP-4
Offset correction
Channel 1
U82
74F283
P0..P2
50
C0..C2
10 pF
R35
82
R36
1K3
+5V
C4
1n
U85
74HC374
ANALOG GND
U83
74F283
U91A
74LS04
Mod adress
channel 1
B0..B11
From sheet 9
Offset correction
Notice that sign
bit should be added
DIG-ANA CONV
CHANNEL 1
DIGITAL SYNTH.
Title
DAC OUTPUT 1
Size
A
Document Number
A. Dabekaussen
REV
Date: September 23, 1988
Sheet 11 of 14

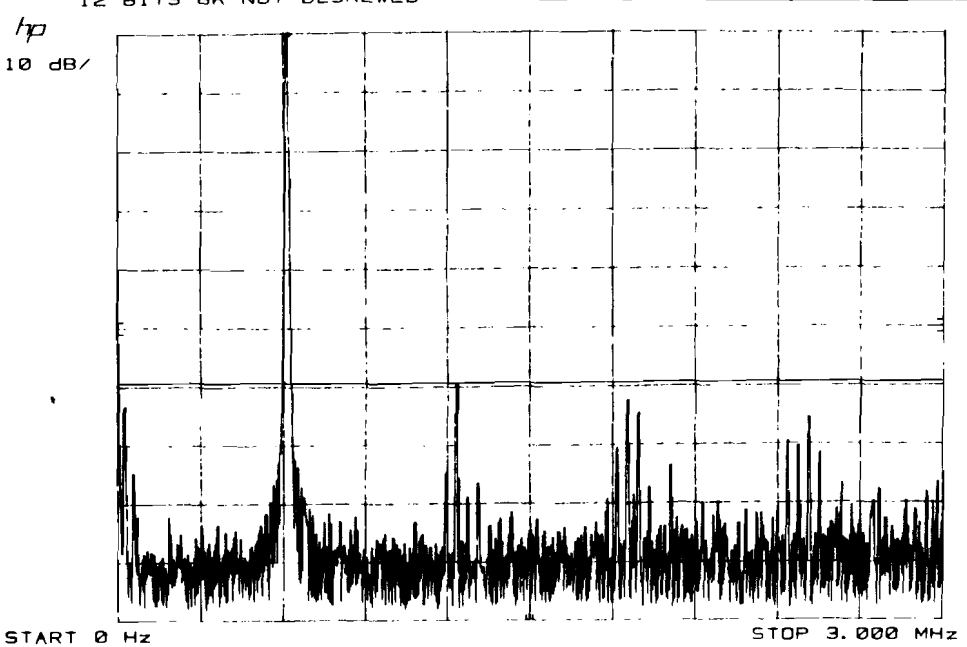

+ 5V
R33/36
2K
SW10
SW DIP-4
U86
74F283
10 MHz clock
- 15V
+ 15V
AD 568
ANALOG
OUT
U91
74HC374
P2..P4
50
C2..C4
10 pF
R37/
R40
2K
SW11
SW DIP-4
Offset correction
Channel 2
U87
74F283
1K3
+5V
ANALOG GND
R41
82
C5
1n
U90
74HC374
U91B
74LS04
DIG-ANA CONV
CHANNEL 2
U88
74F283
Mod adress
channel 2
C0..C11
From sheet 10
Offset correction
Notice that sign-bit
in offset should
be added
DIGITAL SYNTH.
Title
DAC OUTPUT 2
Size
A
Document Number
A. Debakeuseen
REV
Date: September 23, 1988
Sheet 12 of 14

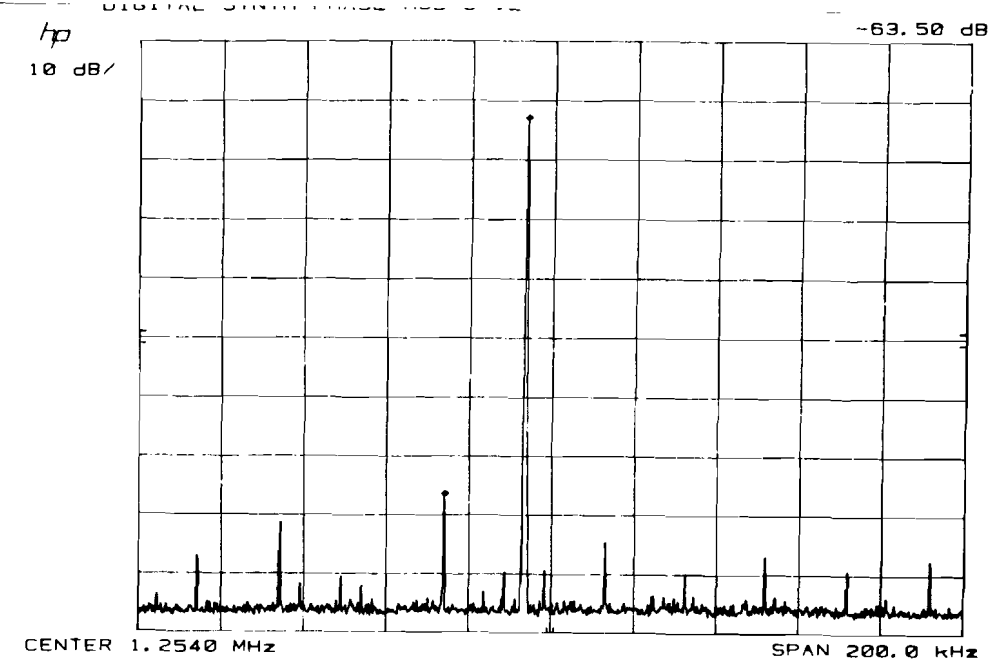

P1
P2
U5
CY7C515
40 CY7C261
28 CY7C261
42 F283
47 F283
46 F283
46 F283
DIP
DIP
AMPL CON
39 F283
39 F283
54 F283
39 F283
44 273
43 273
45 273
38 273
37 273
38 273
31 273
42 273
41 273
50 273
50 F283
56 F283
56 F283
27 F283
51 CY7C261
DIP
DIP
PHASE CON
58 273
55 273
23 273
24 F283
52 CY7C261
12 273
11 273
10 273
56 273
21 273
53 273
54 273
8 F283
7 F283
6 F283
50 F283
18 F283
18 F283
17 F283
6 F283
5 F283
4 F283
18 273
18 273
3 374
2 374
1 374
14 374
13 374
68 367
10 MHz
D con A
D con B
24 P
DIGITAL SYNTH.
Title
LAYOUT BOARD A
Size
B
Document Number
A. Debeteuseen
REV
Date: August 4, 1988 Sheet 13 of 14

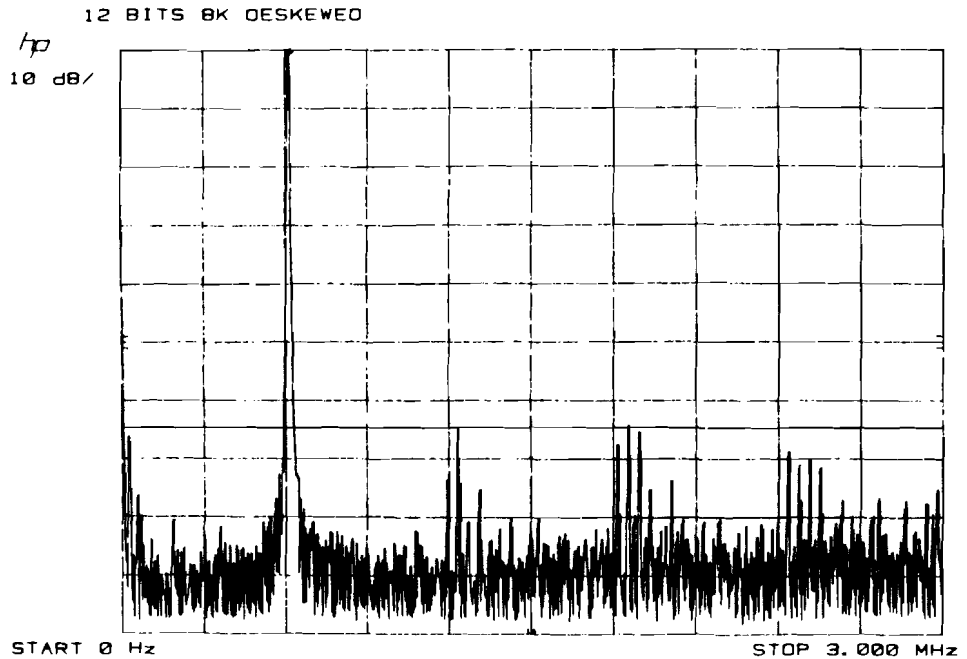

P1
P2
OFFSET CHANEL 1
OFFSET CHANEL 2
D con C
AMPLITUDE MOD
34 P
DIGITAL SYNTH.
Title
LAYOUT BOARD B
Size
B
Document Number
A. Dabekaussen
REV
Date: October 23, 1998 Sheet 14 of 14

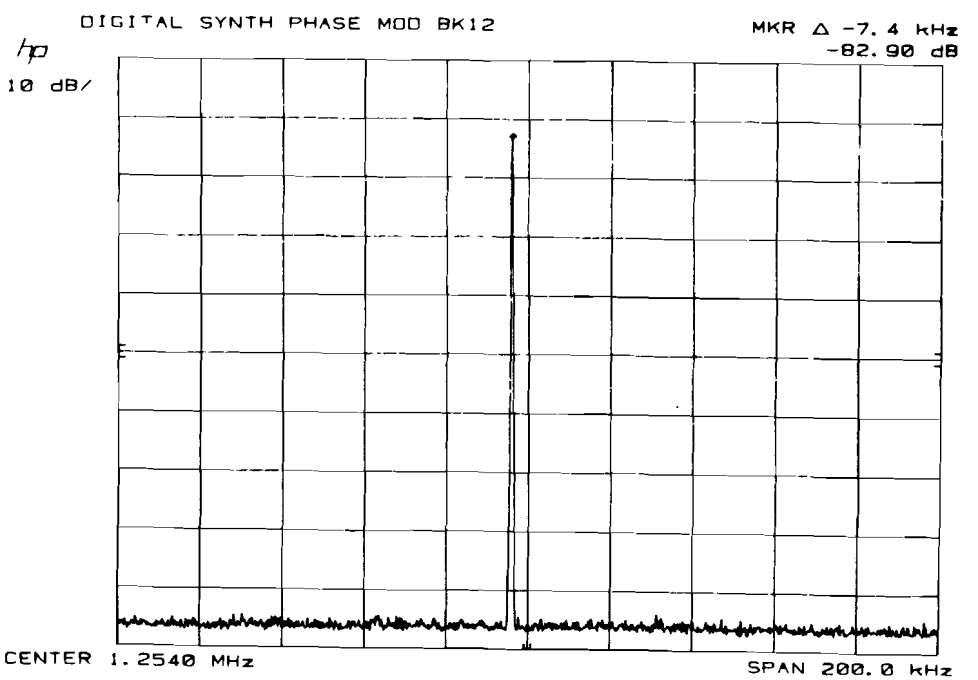
Apppendix 10: Measurement results of the synthesizer

First the influences of deskewing the DAC bits are presented. The glitch peaks far from the carrier and afterward near by the carrier. Finally the influence of a limited ROM table length and width are presented to compare these results with the simulations.


hp
10 dB/
START 0 Hz
STOP 3.000 MHz

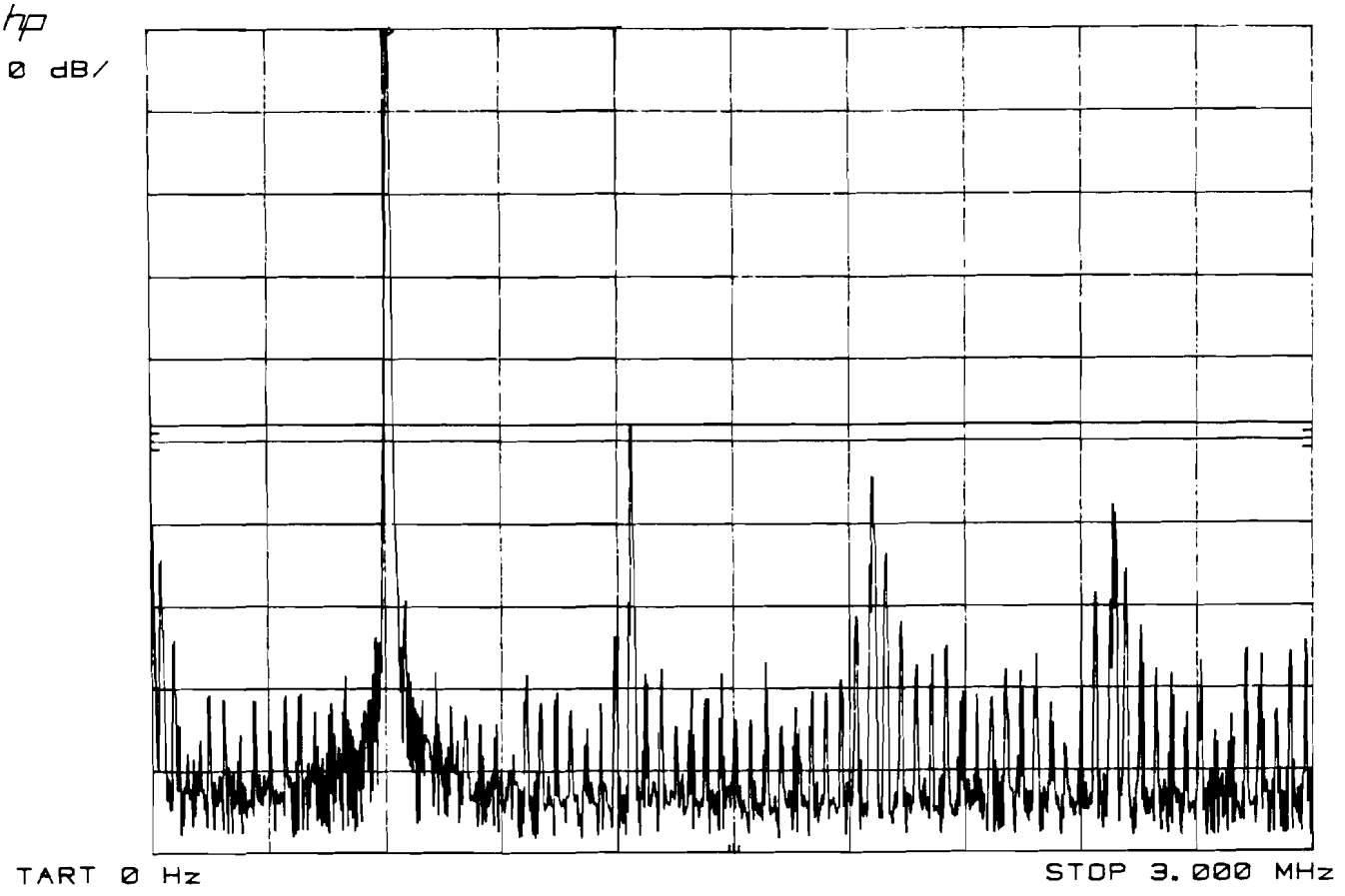

hp
10 dB/
-63.50 dB
CENTER 1.2540 MHz
SPAN 200.0 kHz


12 BITS 8K OESKEWED
hp
10 dB/
START 0 Hz
STOP 3.000 MHz

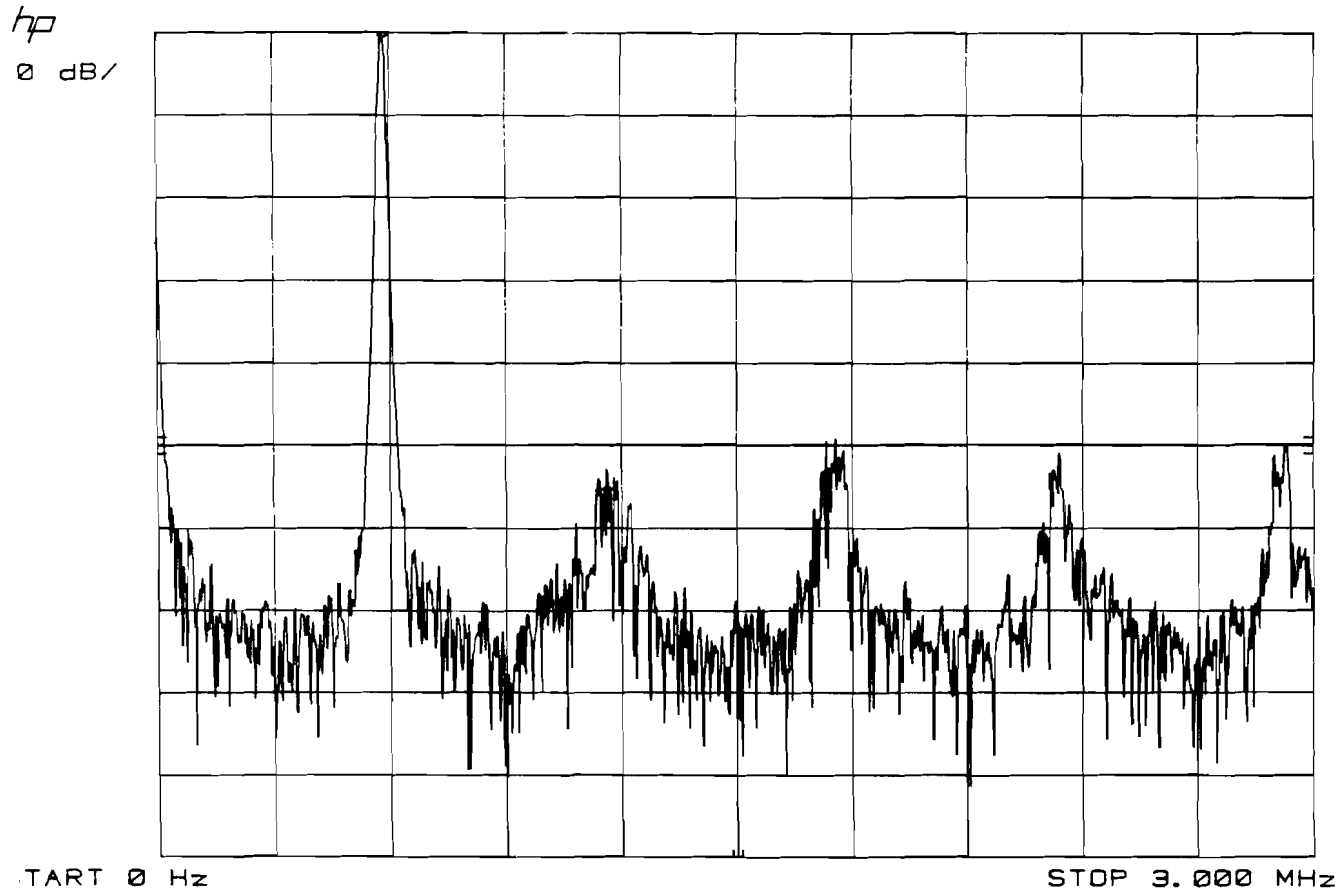

DIGITAL SYNTH PHASE MOD 8K12
MKR Δ -7.4 kHz
-82.90 dB
hp
10 dB/
CENTER 1.2540 MHz
SPAN 200.0 kHz

96

12 BITS 256 WORST CASE

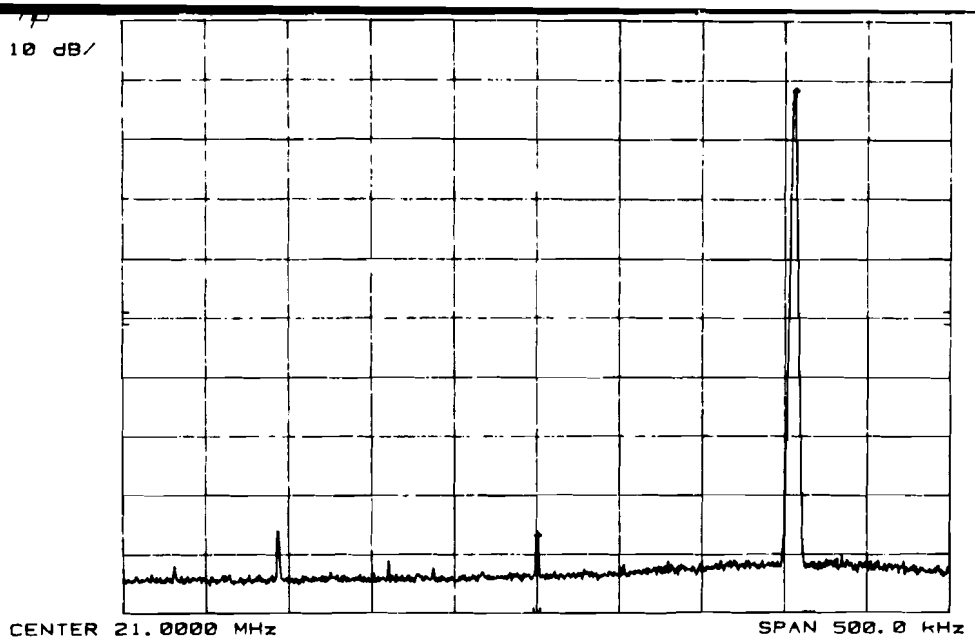


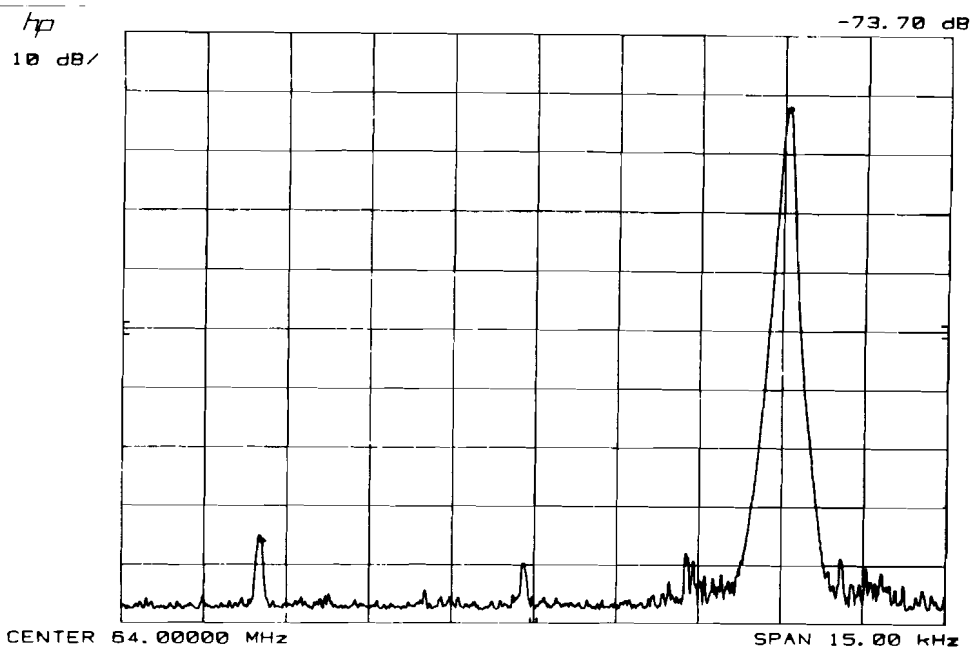
6 BITS 8K

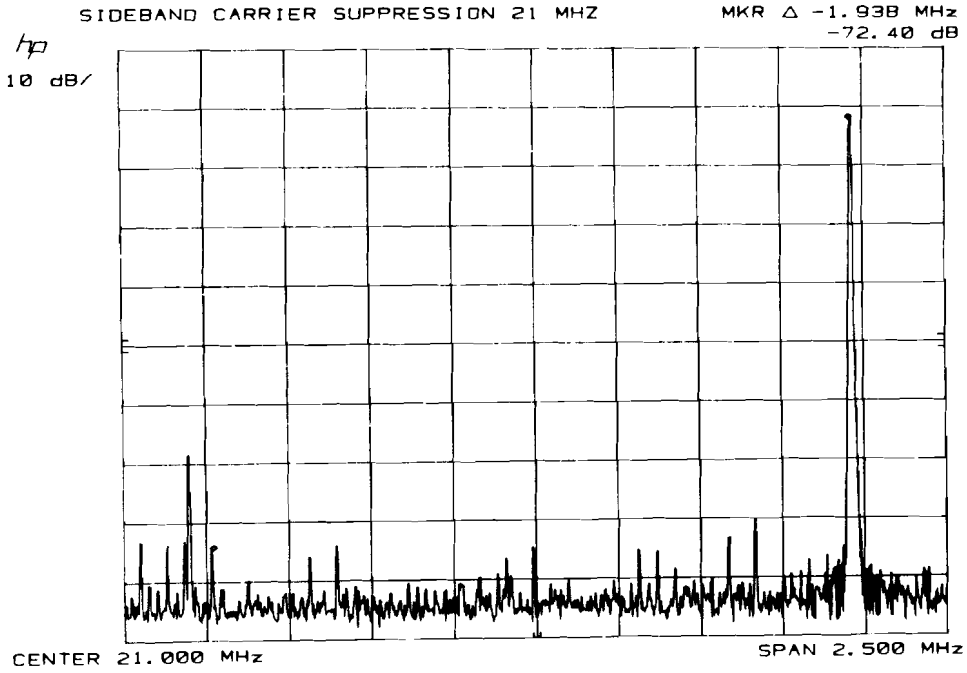


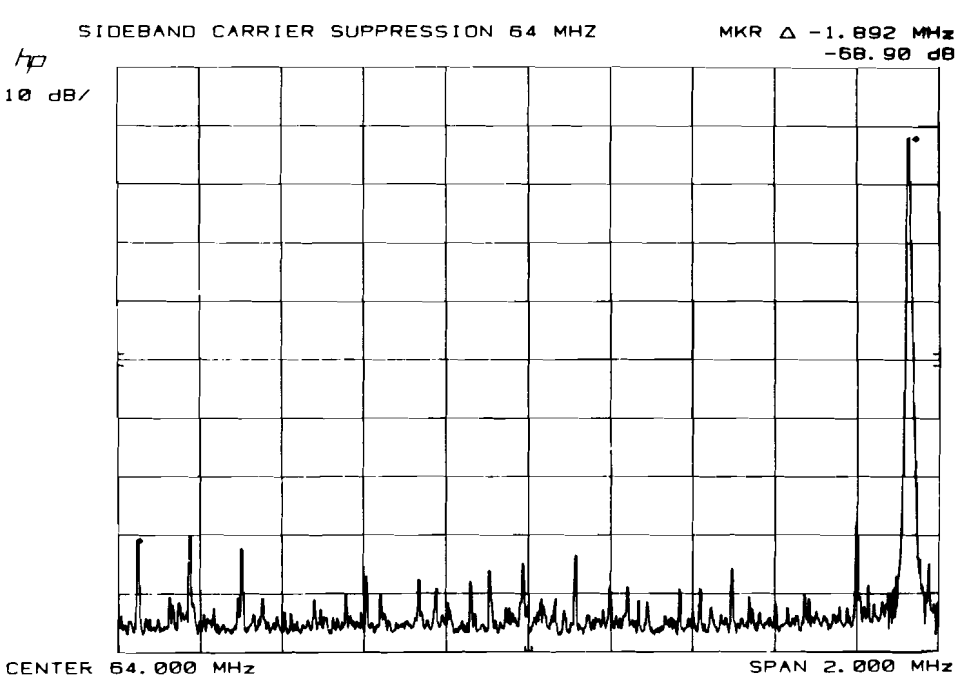
Appendix 11: Measurement results of the SSB modulator

The achieved sideband and carrier suppression for various RF frequencies and synthesizer frequencies are given after optimizing the correction factors.


10 dB/
CENTER 21.0000 MHz
SPAN 500.0 kHz


-73.70 dB
10 dB/
CENTER 64.00000 MHz
SPAN 15.00 kHz


SIDEBAND CARRIER SUPPRESSION 21 MHZ
MKR Δ -1.938 MHz
-72.40 dB
10 dB/
CENTER 21.000 MHz
SPAN 2.500 MHz


SIDEBAND CARRIER SUPPRESSION 64 MHZ
MKR Δ -1.892 MHz
-68.90 dB
10 dB/
CENTER 64.000 MHz
SPAN 2.000 MHz