UNIKASSEL
VERSITÄT

# INTRODUCTION TO INFORMATION THEORY AND CODING "LAB"

Dirk Dahlhaus

Nour Mansour

Communications Laboratory [ComLab]

Faculty of Electrical Engineering and Computer Science
University of Kassel

Winter Semester 2018/2019

# Contents

# Chapter 1

# Introduction

## 1.1 TIMS Overview

- TIMS is a telecommunications modelling system. It models mathematical equations representing electrical signals, or block diagrams representing telecommunications systems.
- TIMS is primarily a hands-on rather than demonstration style teaching system, which combines both the theoretical and practical aspects of implementing systems. TIMS provides the student with a clearer understanding of the concepts behind telecommunications theory.
- Physically, TIMS is a dual rack system. The top rack accepts up to 12 Eurocard sized, compatible black boxes, or modules. The lower rack houses a number of fixed modules, as well as the system power supply.
- The modules are very simple electronic circuits, which function as basic communications building blocks. Each module, fixed or plug-in, has a specific function; functions fall into four general categories:
  - Signal Generation - oscillators, etc
  - Signal Processing - multipliers, filters, etc
  - Signal Measurement - frequency counter
  - Digital Signal Processing - TMS320C50 based (DSP and Advanced Modules are not included in the BASIC TIMS-301)

- Modules are patched together via the front panel sockets using interconnecting leads, to model the system under investigation.

## 1.2 System Conventions

All TIMS modules conform to the following mechanical and electrical conventions.

### 1.2.1 Front Panel Sockets

- Signal interconnections are made via front panel, 4mm sockets.
- Sockets on the LEFT HAND SIDE are for signal INPUTS. (All inputs are high impedance, typically 56 k ohms).
- Sockets on the RIGHT HAND SIDE are for signal OUTPUTS. (All outputs are low impedance, typically 330 ohms).
- YELLOW sockets are only for ANALOG signals. (ANALOG signals are held near the TIMS standard reference level of 4V pk-pk).
- RED sockets are only for DIGITAL signals. (DIGITAL signals are TTL level, 0 to 5 V).
- GREEN sockets are all common, or system GROUND.
- Note that input and output impedances are intentionally mismatched, so that signal connections may be made or broken without changing signal amplitudes at module outputs.

### 1.2.2 Plug-In Modules

- Any plug-in module may be placed in any of the 12 positions of the upper rack.
- All modules use the back plane bus to obtain power supply, only the DSP modules (not part of the BASIC SYSTEM) use the bus to transfer signals.

- The modules are designed so that they may be plugged-in or removed at any time, without turning off the system power.
- The modules are not locked into position and may need to be held while interconnecting leads are removed.

### 1.2.3 Labelling

- All modules are identified as to the function they perform.
- Inputs, outputs, controls and switches are labelled so that a student who has had only a brief introduction to TIMS can use the modules without needlessly referring back to this USER MANUAL.
- It should be noted that no variable controls have calibration marks. This is intentional, as the philosophy behind TIMS is that students setup and adjust systems by observing and measuring signals.
- This assists the student in gaining a much greater understanding, feel and insight into the operation of a communications implementation.

### 1.2.4 Basic Specification

**POWER SUPPLY**

- Input 120, 127, 220 or 240 V AC, 47 Hz to 63 Hz
- Output + 15V, 2.2A DC, -15 V, 2.2A DC
- Protection short circuit, overload, thermal
- Regulation 0.2%

**PHYSICAL**

- Case Dimensions 490(W) x 330(D) x 310(H) mm
- System Weight 10kg
- Plug-in Card Dimensions 160 x 100 mm
- Plug-in Card Bus Connectors 64 way, 2 row, Eurocard

**MODULES**

- Specifications for each module are listed in the appendices

### 1.2.5 Basic Module List

Below are listed all the BASIC SYSTEM MODULES. FIXED modules are located in the lower rack, while PLUG-IN modules can be positioned anywhere in the upper rack.

- Adder - plug-in
- Audio Oscillator - plug-in
- Buffer Amplifiers - fixed
- Dual Analog Switch - plug-in
- Frequency and Event Counter - fixed
- Headphone Amplifier and 3kHz LPF - fixed
- Master Signals - fixed
- Multiplier - plug-in
- Phase Shifter - plug-in
- Quadrature Phase Splitter - plug-in
- Scope Display Selector - fixed
- Sequence Generator - plug-in
- Trunks Panel - fixed
- Tuneable LPF - plug-in
- Twin Pulse Generator - plug-in
- Utilities Module - plug-in
- Variable DC - fixed

- Voltage Controlled Oscillator - plug-in
- 60kHz Lowpass Filter - plug-in

# Chapter 2

# Simple Experiments

This session is for all participants in DC lab. The following time table shows the date of each DC lab separately:

| Module | Lab initial meeting | Number of sessions | Start | End |
|---|---|---|---|---|
| Introduction to Digital Communications | 23.10 | 5 | 29.10.2018 | 26.11.2018 |
| Introduction to Information Theory & Coding | | 3 | 03.12.2018 | 17.12.2018 |
| | | Christmas & new year | | |
| Digital Communication Over Fading Channels | | 5 | 14.01.2019 | 11.02.2019 |

In this chapter, you will work on the fixed modules and use the following basic modules:

1. Audio Oscillator
2. Phase Shifter
3. Sequence Generator
4. Tuneable LPF

## 2.1 Experiment A

1. connect the signals provided by the fixed module Master Signal and try to measure their frequencies using the Frequency Counter
2. use CH-A and CH-B of the oscilloscope to show the signals in time domain
3. try to show a stable signal at the screen by exploiting the trigger properties:
   (a) trigger using channel A
   (b) trigger using channel B
   (c) external trigger
4. change the oscilloscope to the spectrum mode and show the signals in frequency domain

## 2.2 Experiment B

1. use the module Audio Oscillator to generate a sinewave signal
2. tune the Audio Oscillator to 8 kHz sin or/and TTL using Frequency Counter, and use CH-A to show the signals in both time and frequency domains
3. try to change the tune of the module and monitor the corresponding changes using the oscilloscope
4. shift the phase of the sinewave signal using Phase Shifter

## 2.3 Experiment C

1. using 2 kHz clk signal from Master Signal, generate a random TTL message signal using the module Sequence Generator
2. show the clk signal on CH-A while the TTL message on CH-B
3. show the spectrum of the message on CH-A, is the spectrum limited?

4. try to limit the spectrum using the module Tuneable LPF such that the side loops of the sinc are filtered out, show the output of the filter on CH-B

# Chapter 3

# Pulse Code Modulation (PCM)

***ACHIEVEMENTS*** experimental verification of PCM encoder and decoder.

## 3.1 Source Coding Vs. Channel Coding

Two important coding methods are used in almost all communication systems, namely:

- Source coding: is about removing the redundant data or the data which is not that important to be conveyed to the receiver. So in source coding we remove more of a redundant data which is not needed.
- Channel coding: is about adding some redundant bits in the form of parity bits so that you can protect the data from becoming corrupt. The redundant bits can even help in correcting the corrupt data. There are two main types of channel coding: block coding and convolutional coding.

## 3.2 Introduction to PCM

PCM is one of the several analog source coding techniques that are designed to represent the time-domain characteristics of the signal, in other words it

is a temporal waveform coding technique.

- Let $x(t)$ denote a sample function emitted by a source and let $x_n$ denote the samples taken at a sampling rate $f_s \geq 2W$, where $W$ is the highest frequency in the spectrum of $x(t)$.
- In PCM, each sample of the signal is quantized to one of $L = 2^R$ amplitude levels where $R$ is the number of binary digits used to represent each sample. Thus, the rate from the source is $Rf_s$ bit/s.
- The quantized process may be modeled mathematically as

$$\tilde{x}_n = x_n + q_n,$$

where $\tilde{x}_n$ represents the quantized value of $x_n$ and $q_n$ denotes the quantization error, which we treat as an additive noise.

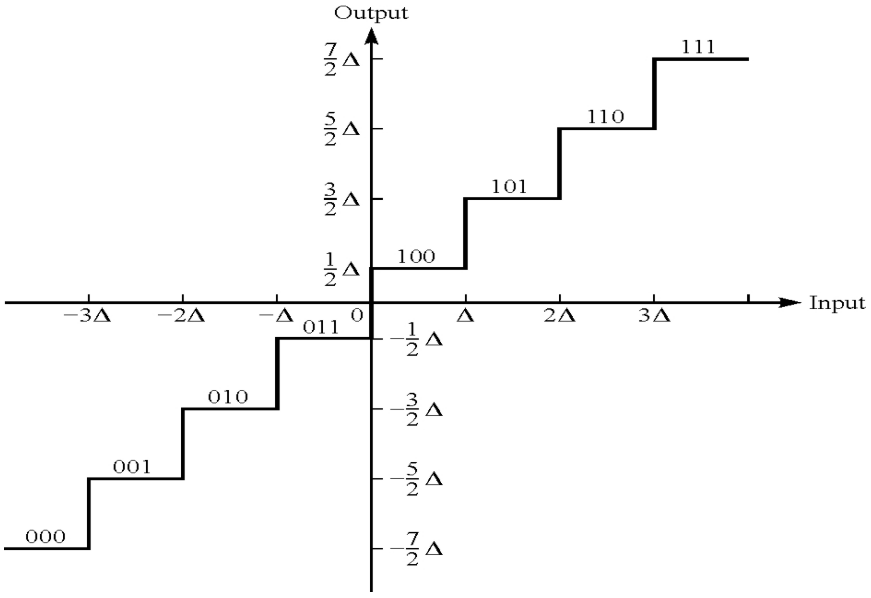



Figure 3.1: Input-output characteristic for a uniform quantizer.

## 3.3 PCM with Non-uniform Quantizer

Many source signals (e.g. speech waveforms) have the characteristic that small signal amplitudes occur more frequently than large ones. However,

a uniform quantizer provides the same spacing between successive levels throughout the entire dynamic range of the signal.

Given that we know what this dynamic range is, a better approach is therefore to employ a nonuniform quantizer. The latter one is usually implemented in two steps:

- pass the signal through a non-linear device that compresses the signal amplitude
- the output of the nonlinearity is fed to a uniform quantizer.

An example of a compressor which can be configured by a parameter $\mu$, is the logarithmic compressor:

$$|y| = \frac{\log(1 + \mu|x|)}{\log(1 + \mu)},$$

where $|x| \leq 1$ is the normalized magnitude of the input, $|y|$ is the magnitude of the output, and $\mu$ is a parameter to give the desired compression characteristic ($\mu = 0$ indicating no compression).

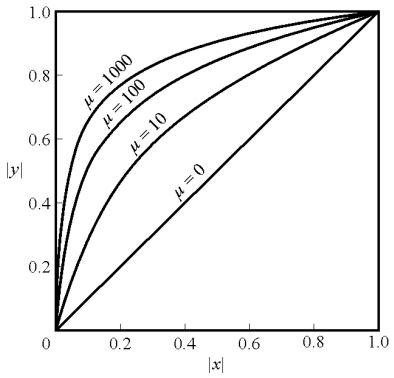



Figure 3.2: Input-output characteristic for a logarithmic compressor.

- In the encoding of speech waveforms, for example, the value of $\mu = 255$ has been adopted as a standard in the United States and Canada, which results in about a 24-dB reduction in the quantization noise power relative to uniform quantization.

- Clearly, at the receiver which reconstructs the signal, we have to carry out an inverse logarithmic mapping in order to expand the signal amplitude. The combined compressor-expander pair is termed compandor.

## 3.4 Experiment A

The only module required for this experiment is a TIMS PCM ENCODER shown in Fig. 3.3. The operation of this module can be described as following:

- the module is driven by an external TTL clock
- the input analog message is sampled periodically. The sampling rate is determined by the frame synchronizing signal (FS)
- the sampling operation is internal to the module. What is held is the amplitude of the analog message at the sampling instant
- each sample amplitude is compared with a finite set of amplitude levels $L$ (system quantizing levels) distributed (uniformly, for linear sampling) within the range $\mp 2.0$ volts
- each quantizing level is assigned a number, starting from zero for the lowest (most negative) level, with the highest number being $L - 1$
- each sample is assigned a digital (binary) code word (CW)

**DC Message**

### 3.4.1 PCM Encoding

1. put the left hand toggle of the on-board switch SW2 "down" and the right hand toggle "up"
2. insert the module into the TIMS frame. Switch the front panel toggle switch to 4-bit linear (i.e., no companding)
3. patch the 8.333 kHz TTL bit CLK from the MASTER SIGNALS module to the CLK input of the PCM ENCODER module

4. connect the $V_{\text{in}}$ input socket to ground of the variable DC module
5. using the FREQUENCY COUNTER, measure the frequency of SYNC MESSAGE from the module
6. connect the frame synchronization signal FS to the oscilloscope ext. synch input
7. on CH1-A display the frame synchronization signal FS while on CH1-B display the CLK signal

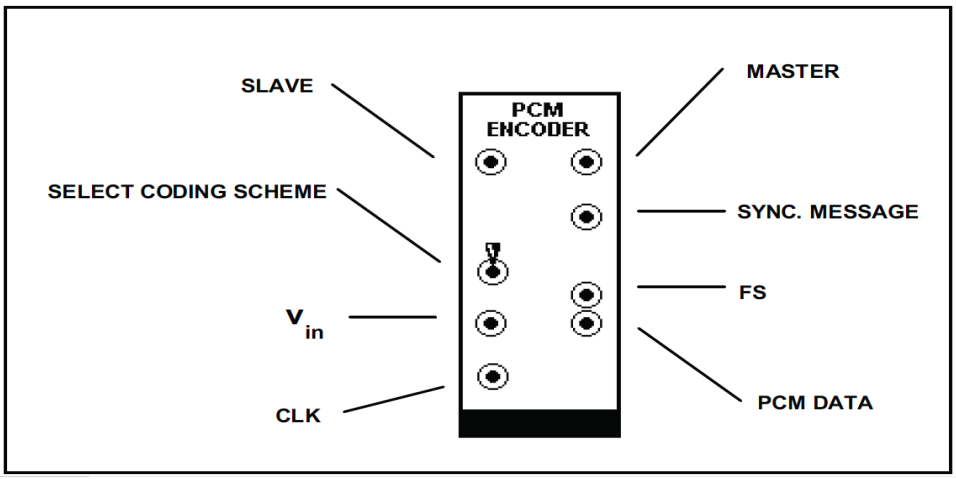



Figure 3.3: PCM encoder

ToDo record the number of clock periods per frame

8. on CH1-B display the PCM DATA from the PCM DATA output socket
9. your display should be similar to that of Fig. 3.4 below knowing:
   (a) the number of slots per frame is 8
   (b) the location of the least significant bit is coincident with the end of the frame
   (c) the binary word length is four bits
   (d) the first three slots are "empty"
10. identify the binary word in slots 4, 3, 2, and 1.

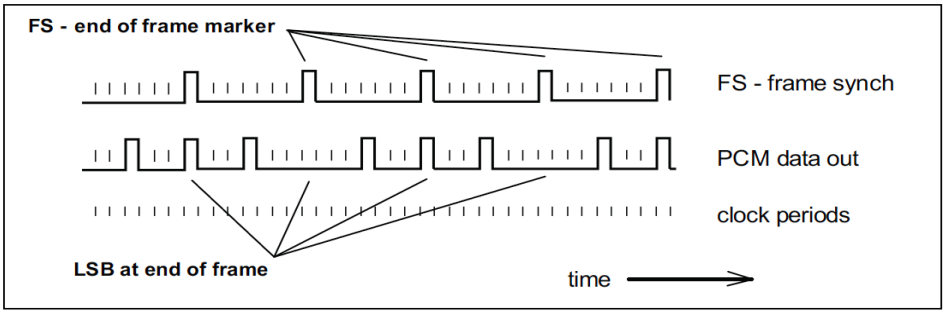



Figure 3.4: 5 frames of 4-bit PCM output for zero amplitude input

11. remove the ground connection, and connect the output of the VARIABLE DC to $V_{\text{in}}$

12. using WIDEBAND TRUE RMS METER module, monitor the DC amplitude at $V_{\text{in}}$

13. sweep the DC voltage slowly from the maximum negative value to the maximum positive value and record the DC voltage at each sudden change to the PCM output signal. Fill the table below:

| Uniform Encoding | |
|---|---|
| DC level | PCM CW |
| .. | .. |
| .. | .. |

14. continue this process over the full range of the DC supply

15. change to 7-bit linear encoding and recognize the changes in the CW

16. change to 4-bit companding by use of the front panel toggle switch ($A_4$ companding law has already been selected)

17. make the necessary measurements to determine the nature of the law $A_4$ by filling the table below:

| Non-uniform Encoding | |
|---|---|
| DC level | PCM CW |
| .. | .. |
| .. | .. |

### 3.4.2 PCM Decoding

The PCM DENCODER shown in Fig. 3.5 is used for decoding. Upon reception the PCM signal, the PCM DECODER

- is driven by an external clock synchronized to that of the transmitter. For this experiment a "stolen" clock will be used
- extracts a frame synchronization signal FS from the data itself (use an FS signal stolen from the transmitter)
- extracts the binary number, which is the coded (and quantized) amplitude of the sample from which it was derived
- identifies the quantization level
- generates a voltage proportional to this amplitude level
- presents this voltage to the output $V_{\text{out}}$

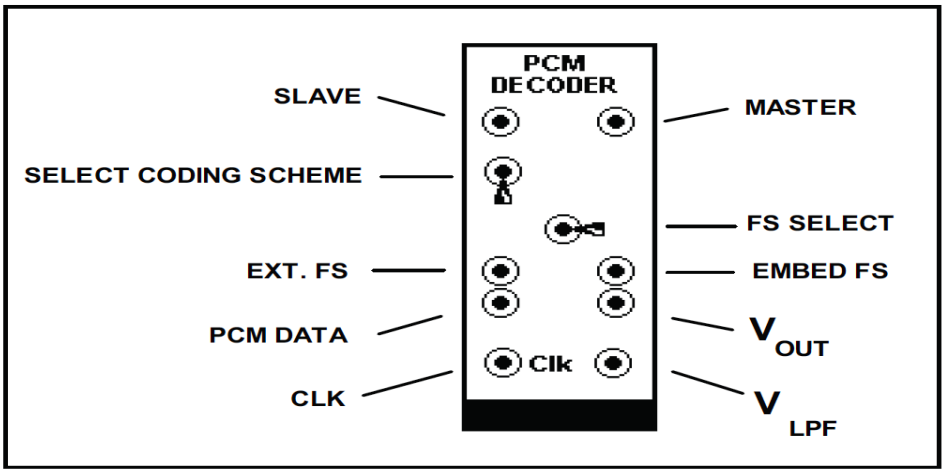



Figure 3.5: PCM decoder

1. steal the FS signal from the transmitter. At the same time ensure that the FS SELECT toggle switch on the receiver is set to EXT. FS
2. connect the PCM DATA output signal from the transmitter to the PCM DATA input of the receiver

3. connect CH1-B to the output of the PCM DECODER

4. vary the DC input data signal and observe the same corresponding changes at the output of the decoder

## 3.5 Experiment B

**Periodic Message**

### 3.5.1 PCM Encoding

1. take a periodic message from the SYNC. MESSAGE socket and observe it on CH-A
   **ToDo** check if these are compatible with the Nyquist criterion

2. observe the PCM DATA output on CH-B

### 3.5.2 PCM Decoding

1. compare between the message and the output of the PCM decoder
   **ToDo** to get rid of the distortion, use TUNABLE LPF

2. change to 7-bit linear encoding and draw your conclusion

# Chapter 4

# Introduction to Block Coding

***ACHIEVEMENTS*** experimental verification of block encoder and decoder, in addition to Hamming code and parity code.

## 4.1 Introduction to Block Coding

- A block code consists of a set of fixed-length vectors called code words of length $n$.
- The elements of a code word are selected from an alphabet of $q$ symbols or elements. When there are only $q = 2$ elements, namely 0 and 1, the code is called binary and the elements are so-called bits.
- There are $2^n$ possible code words in a binary block code of length $n$. From these $2^n$ code words, we may select $M = 2^k$ code words with $k < n$ to form a code. Thus, a block of $k$ bits is mapped into a code word of length $n$ and we refer to the resulting block code as an $(n, k)$ code with code rate $R_c = k/n$.

## 4.2 The Generator Matrix

- Let $\mathbf{X}_m = [x_{m1}, x_{m2}, \ldots, x_{mk}]$ denote the row vector of $k$ information bits entering the encoder. The output is the code word defined by the

$n$-dimensional row vector

$$\mathbf{C}_m = [c_{m1}, c_{m2}, \ldots, c_{mn}] = \mathbf{X}_m \mathbf{G}$$

with the generator matrix of the code defined by

$$\mathbf{G} = \begin{bmatrix} \leftarrow & \mathbf{g}_1 & \rightarrow \\ \leftarrow & \mathbf{g}_2 & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{g}_k & \rightarrow \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ \vdots & \vdots & & \vdots \\ g_{k1} & g_{k2} & \cdots & g_{kn} \end{bmatrix}.$$

- The code words can be written as linear combinations of the row vectors $\{\mathbf{g}_i\}$ of $\mathbf{G}$:

$$\mathbf{C}_m = x_{m1}\mathbf{g}_1 + x_{m2}\mathbf{g}_2 + \ldots + x_{mk}\mathbf{g}_k.$$

- Any generator matrix of an $(n, k)$ code can be reduced by Gaussian elimination, i.e. row operations and column permutations to the systematic form:

$$\mathbf{G} = [\mathbf{I}_k | \mathbf{P}] = \left[\begin{array}{ccccc|ccccc} 1 & 0 & 0 & \ldots & 0 & p_{11} & p_{12} & \ldots & p_{1\,n-k} \\ 0 & 1 & 0 & \ldots & 0 & p_{21} & p_{22} & \ldots & p_{2\,n-k} \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \ldots & 1 & p_{k1} & p_{k2} & \ldots & p_{k\,n-k} \end{array}\right].$$

  Here, $\mathbf{I}_k$ and $\mathbf{P}$ denote a $(k \times k)$-dimensional identity matrix and a $(k \times n-k)$-dimensional matrix containing the $n-k$ redundant bits called parity check bits, respectively.

- Note that a generator matrix of the systematic form generates a linear block code in which the first $k$ bits of each code word are identical to the information bits to be transmitted. The remaining $n-k$ bits of each code word are linear combinations of the $k$ information bits.

## 4.3 Specific Linear Block Codes

### 4.3.1 Hamming Codes

- We only consider binary Hamming codes with

$$(n, k) = (2^m - 1, 2^m - 1 - m),$$

where $m$ is any positive integer. For example, when $m = 3$, we have a $(7, 4)$ code.

- The generator matrix $\mathbf{G}$ of the Hamming code used in the following experimental section is systematic and given by

$$\mathbf{G} = [\mathbf{P}_{4\times3}|\,\mathbf{I}_4] = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

- Hamming codes can detect up to two-bit errors or correct one-bit errors without detection of uncorrected errors

### 4.3.2 Parity Codes

- A parity bit, or check bit, is a bit added to a binary sequence that indicates whether the number of ones in the sequence is even or odd. Parity bits are used as the simplest form of error detecting code. There are two variants of parity bits: even parity bit and odd parity bit.

- In the case of even parity, for a given set of bits, the occurrences of bits whose value is 1 is counted. If that count is odd, the parity bit value is set to 1. If the count of ones in a given set of bits is even, the parity bit's value is 0.

- In the case of odd parity, the coding is reversed. For a given set of bits, if the count of bits with a value of 1 is even, the parity bit value is set to 1. If the count of bits with a value of 1 is odd, the parity bit's value is 0.

- The generator matrix $\mathbf{G}$ of the parity code used in the following experimental section is even and given by

$$\mathbf{G} = [\mathbf{P}_{4\times3}|\,\mathbf{I}_4] = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix},$$

which can be considered as $(5, 4)$ because the last two bits are unused.

## 4.4 Block Code Format

The BLOCK CODE ENCODER module is designed for input blocks of length eight slots, where some of these slots are empty. The data frame is illustrated in Fig. 4.1 where

- the message bits are shown as $D_3, D_2, D_1, D_0$ where $D_3$ is the most significant bit of the message
- the slots $C_2, C_1, C_0$ is used by the BLOCK ENCODER for code bits
- the frame synchronization bit is shown as FS

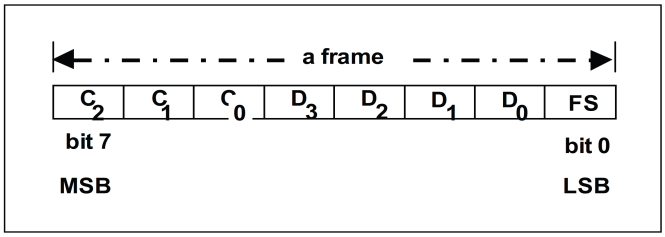



Figure 4.1: A data frame of eight slots, one per clock period

- BLOCK CODE ENCODER module receive always three digital signals
  - TTL binary data in an 8-bit wide frame where the data from the PCM ENCODER must occupy the slots 4, 3, 2, and 1
  - 2.083 kHz TTL CLK, to which the incoming data is synchronized
  - TTL frame synchronization signal FS

## 4.5 Lab Experiment

**Perfect Channel**

The block digram of this experiment is shown in Fig. 4.2

1. the BLOCK CODE ENCODER requires a TTL clock near 2 kHz which can be obtained by dividing 8.333 kHz TTL signal from the MASTER SIGNALS module by 4 using the LINE-CODE ENCODER module

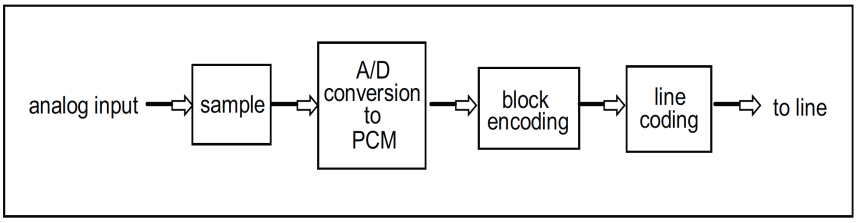



Figure 4.2: The system using block encoder

2. patch up the model in Fig. 4.3 which is equivalent to Fig. 4.2
3. switch the front panel toggle switch of the PCM ENCODER to 4-bit linear
4. change the scope sweep such that three frames can be seen on the screen
5. firstly select the parity check coding
6. add the signal FS as a reference signal
7. on CH1-A display the PCM encoder output while on CH1-B block encoded CW

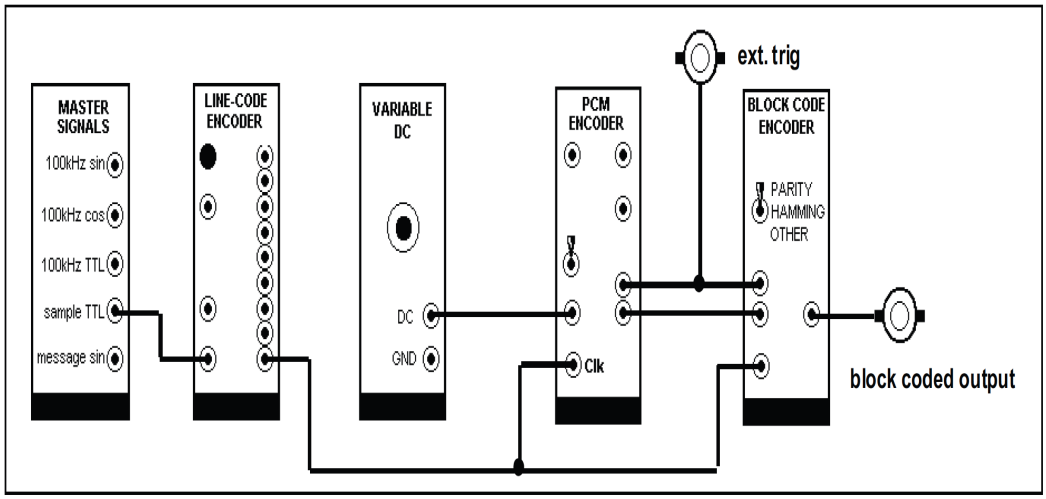



Figure 4.3: The system using block encoder

8. sweep the DC voltage slowly from the maximum negative value to the maximum positive value and record the DC voltage at each sudden change to the PCM output signal. Fill the table below:

| Parity Check Coding | |
|---|---|
| PCM CW | Block Coded CW |
| .. | .. |
| .. | .. |

**ToDo** ensure that the parity check CWs you got from the above table are the same as $\mathbf{c} = \mathbf{dG}$ where

$\mathbf{c} = [C_2\ C_1\ C_0\ D_3\ D_2\ D_1\ D_0]$ Block CW

$\mathbf{d} = [D_3 D_2 D_1 D_0]$ PCM CW

$$\mathbf{G} = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$ The generating matrix of parity check

**ToDo** after filling up the table, conclude how the parity check coding works

9. secondly select the Hamming coding and fill up the

| Hamming Coding | |
|---|---|
| PCM CW | Block Coded CW |
| .. | .. |
| .. | .. |

**ToDo** ensure that the Hamming CWs you got from the above table

are the same as $\mathbf{c} = \mathbf{dG}$ where

$$\mathbf{c} = [C_2\ C_1\ C_0\ D_3\ D_2\ D_1\ D_0] \qquad \text{Block CW}$$

$$\mathbf{d} = [D_3 D_2 D_1 D_0] \qquad \text{PCM CW}$$

$$\mathbf{G} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad \text{The generating matrix of Hamming}$$

10. patch up the BLOCK CODE DECODER according to Fig. 4.4 with using the "stolen" FS from the transmitter

11. verify that successful decoding back to the original PCM is possible for all code words

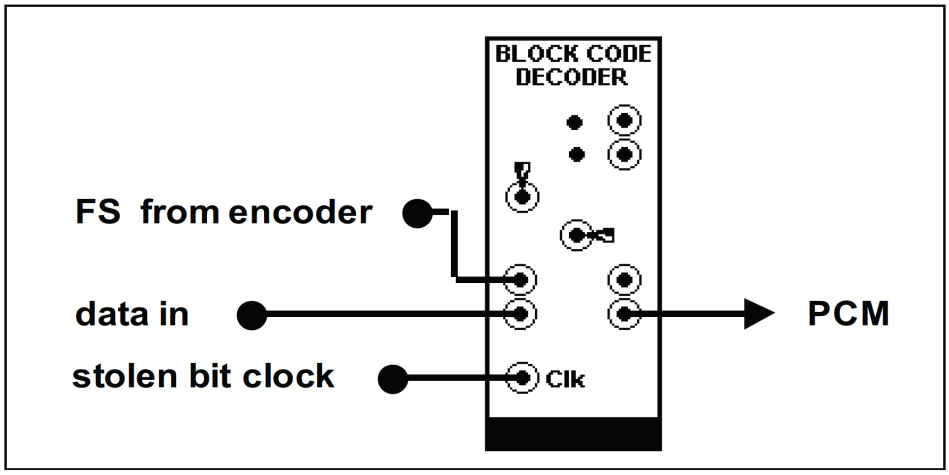



Figure 4.4: The bolck code decoding

# Chapter 5

# Error Detection and Correction for Block Coding

***ACHIEVEMENTS*** Error detection and correction with noisy channel using Hamming code and parity code.

## 5.1 Hamming Distance

- Suppose that $\mathbf{C}_i$ and $\mathbf{C}_j$ are any two code words in an $(n, k)$ block code. A measure of the difference between the code words is the Hamming distance denoted as $d_{ij}$.
- Clearly, we have $0 \leq d_{ij} \leq n$.
- The smallest value of the set $\{d_{ij}\}$ with $i \neq j$ is called the minimum distance $d_{\min}$.
- The minimum distance of a code determines both its error-detecting ability and error-correcting ability
- Linear codes:
  - Suppose that $\alpha_1$ and $\alpha_2$ are any two elements selected from the alphabet.

- Then, the code is said to be linear if $\alpha_1 \mathbf{C}_i + \alpha_2 \mathbf{C}_j$ is also a code word. This implies (by choosing $\alpha_1 = \alpha_2 = 0$) that a linear code contains the all-zero code word. Consequently, a constant-weight code is non-linear.
- Let $\mathbf{C}_i$ with $i = 1, \ldots, M$ denote the $M$ code words of a linear code with $\mathbf{C}_1 = [00 \cdots 0]$ denoting the all-zero code word. Furthermore, $w_r$ is the weight of the $r$-th code word which is obviously the Hamming distance between the code words $\mathbf{C}_r$ and $\mathbf{C}_1$, i.e., $d_{1r} = w_r$.
- In general, the distance $d_{ij}$ between code words $\mathbf{C}_i$ and $\mathbf{C}_j$ is the weight of the difference between $\mathbf{C}_i$ and $\mathbf{C}_j$.
- Since the code is linear, the difference between $\mathbf{C}_i$ and $\mathbf{C}_j$ is also a code word with weight included in the set $\{w_r\}$.
- That is

$$\boxed{d_{\min} = \min_{r, r \neq 1} w_r}$$

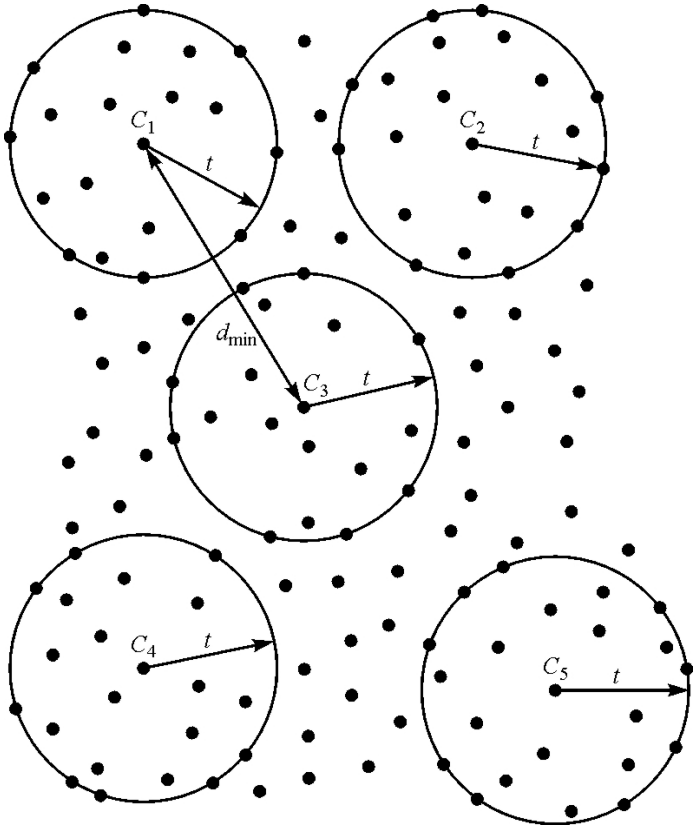
## 5.2 Error Detection

- From the discussion above, when the syndrome is the zero vector, the received word is one of the $2^k$ possible transmitted code words.
- Since the minimum separation between a pair of code words is $d_{\min}$, it is possible for an error pattern of weight $d_{\min}$ to transform one of these $2^k$ code words in the code into another code word.
- In this case, we have an undetected error.
- If the actual number of errors is less than $d_{\min}$, the syndrome will have a nonzero weight. When this occurs, we have detected the presence of one or more errors on the channel.
- Clearly, the $(n, k)$ block code is capable of detecting $d_{\min} - 1$ errors. Note that detecting $d_{\min} - 1$ errors does not mean that the number of errors would be known at the receiver. It simply means that a non-valid code word has been recognized by the receiver.

- Error detection may be used in conjunction with an automatic repeat-request (ARQ) scheme for retransmission of the code word in error.

## 5.3 Error Correction

- The error correction capability of the code also depends on the minimum distance. However, the number of correctable error patterns is clearly limited by the number of possible syndromes (or coset leaders in the standard array).
- We interpret the $2^k$ possible code words as elements in an $n$-dimensional vector space, where each code word is viewed as the centre of a sphere of radius $t$ (all distance measures represent Hamming distances). The value of $t$ is the largest value so that the spheres are non-overlapping.
- As a result, $t = \left\lfloor \frac{1}{2}(d_{\min} - 1) \right\rfloor$, where $\lfloor x \rfloor$ denotes the largest integer contained in $x$. Within each sphere lie all the possible received words of distance less than or equal to $t$ from the valid code word.
- Consequently, any received code vector that falls within a sphere is decoded into the valid code word at the centre of the sphere.
- We say that the $(n, k)$ code with minimum distance $d_{\min}$ is capable to correct $t = \left\lfloor \frac{1}{2}(d_{\min} - 1) \right\rfloor$ errors.




A representation of code words as centers of spheres of radius $t = \left\lfloor \frac{1}{2}(d_{\min} - 1) \right\rfloor$.

- Clearly, to correct $t = \left\lfloor \frac{1}{2}(d_{\min} - 1) \right\rfloor$ errors, we first have to detect them. For example, for a code with $d_{\min} = 7$, we could define spheres of radius $t = 3$ around each code word. If a received code word falls within a certain sphere, we decode the centre of the sphere as the transmitted code word.

- Can we increase the error detection capability of the code? Obviously, if we keep $t = 3$, it is not possible to do so, since as soon as a received code word is inside a certain sphere, the syndrome in the corresponding set would decode the vector to the word in the sphere centre.

- However, if we reduce the radius of the spheres, there will be code words between the spheres which, if received, will make the receiver to ask for a retransmission. As an example, assume $d_{\min} = 7$ and $t = 2$. Obviously, around each sphere centre, we have *orbits* of radius 3 or radius 4 which do not overlap with the spheres around neighbored

code words. If more than four errors occur they will not be detected if they fall into another sphere.

- Similarly, we can decrease the sphere radius to $t = 1$ and detect five errors.
- In general, a code with minimum distance $d_{\min}$ can detect $e_d$ errors and correct $e_c$ errors, where

$$e_d + e_c \leq d_{\min} - 1 \qquad \textbf{and} \qquad e_c \leq e_d$$

- A perfect code has the property that all spheres of Hamming distance $t = \left\lfloor \frac{1}{2}(d_{\min} - 1) \right\rfloor$ around the $M = 2^k$ possible code words are disjoint and every received code word falls in one of the spheres.

## 5.4 Lab Experiment

The overall block digram of this experiment is shown in Fig. 5.1. It is recommended to implement the system step by step (mentioned below) checking the output of each block using the oscilloscope .

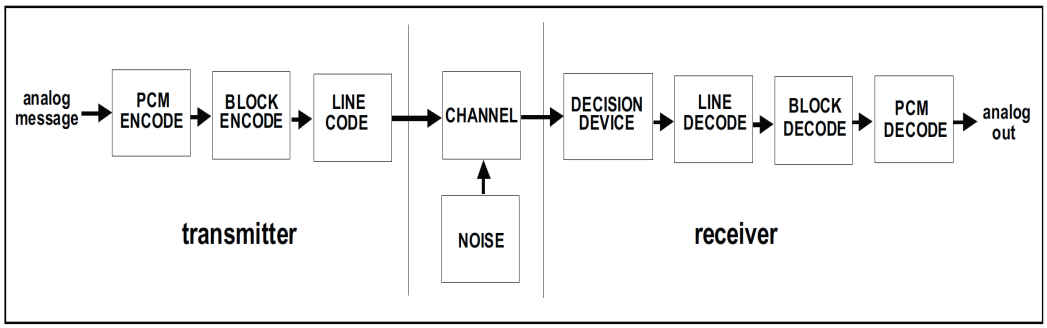



Figure 5.1: The system block diagram

- the DECISION MAKER has a CLK limited to the vicinity of 2 kHz
- for 8 bits frame the sampling rate will be 260 samples/sec = 2083/8
- the maximum message frequency will be limited to 130 Hz (Nyquist)

***Transmitter:***

1. patch up the model of Fig. 5.2 which is the transmitter (left part) in Fig. 5.1
2. select the 4-bit LINEAR PCM code, and PARITY block coding
3. change the scope sweep such that three frames can be seen on the screen
4. add the signal FS as a reference signal
5. on CH-A check the input of each block while on CH-B check its output

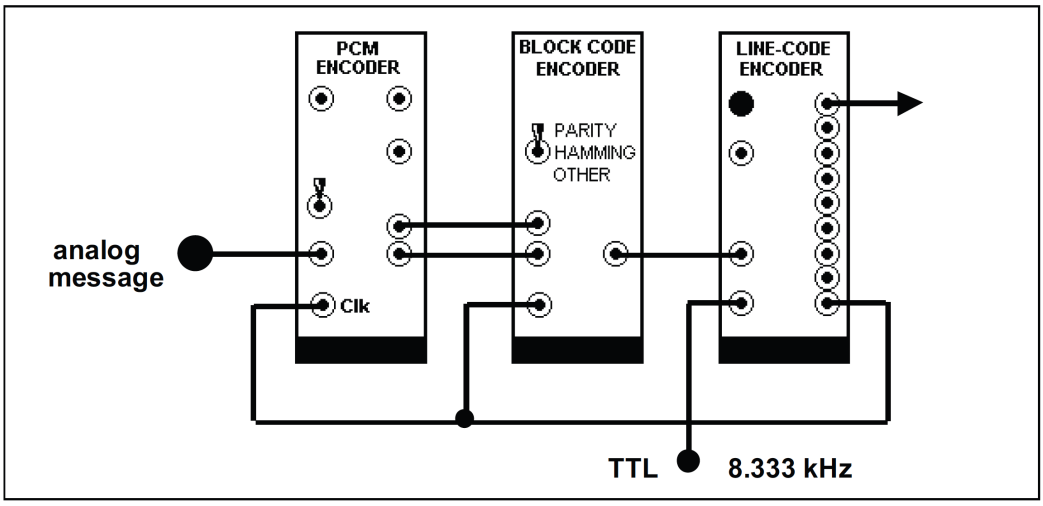



Figure 5.2: The transmitter

***Channel:***

1. patch up the noisy channel in Fig. 5.3 using TUNEABLE LPF for the bandlimitation and connect the output signal of Fig. 5.2 to be the input signal of the noisy channel
2. amplify the signal at the output of the channel using buffer amplifier
3. patch both the oscilloscope and the WIDEBAND TRUE RMS METER to observe the signal at the channel output (in this case the output of the channel is the output of the buffer amplifier)

4. observe the wave shape at the channel output for full channel bandwidth, then tune the filter until there is obvious bandlimiting

5. reduce the signal amplitude to zero with the "G" gain control of the ADDER

6. set the attenuator of the NOISE GENERATOR to the nominal read 14 dB. Change the gain of the ADDER such that the reading of WIDEBAND TRUE RMS METER is 0.5 volt rms at the OUTPUT ADDER ($N = 0.5$ volt rms amplitude)

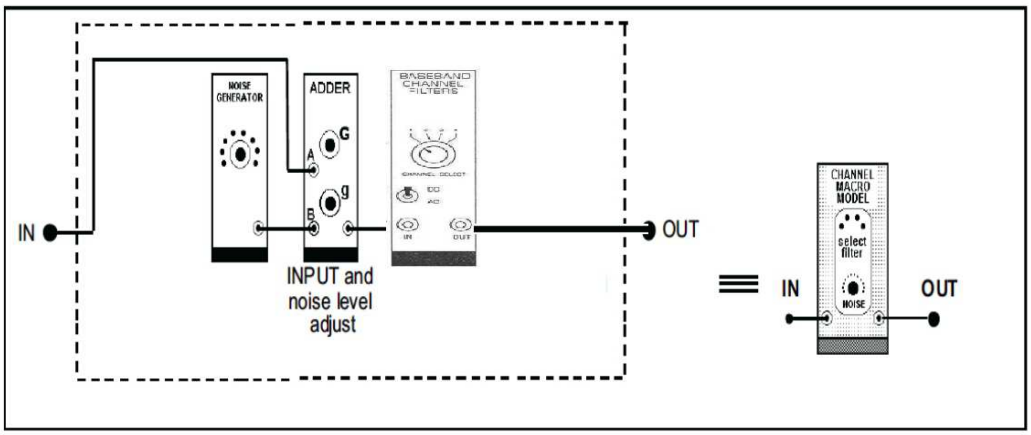



Figure 5.3: The macro CHANNEL MODEL module

7. remove the noise by unplugging the patch cord from the ADDER

8. introduce some signal with the "G" control of the channel ADDER until the rms meter is reading the same as the previous noise reading. Record this reading ($S = 0.5$ volt rms amplitude)

9. reconnect the noise again. Do *NOT* disturb the ADDER gain settings from now on!

10. the SNR is now set up to a reference value $10 \log_{10} \left[ \frac{S^2}{N^2} \right] = 0$

11. to have higher SNR, reduce the noise by the full available attenuation of the NOISE GENERATOR front panel attenuator e.g. to 0 dB

**the signal is now ready for demodulation, presumably without errors, since the SNR should be well above 0 dB (≈ 14 dB)**

***Receiver:***

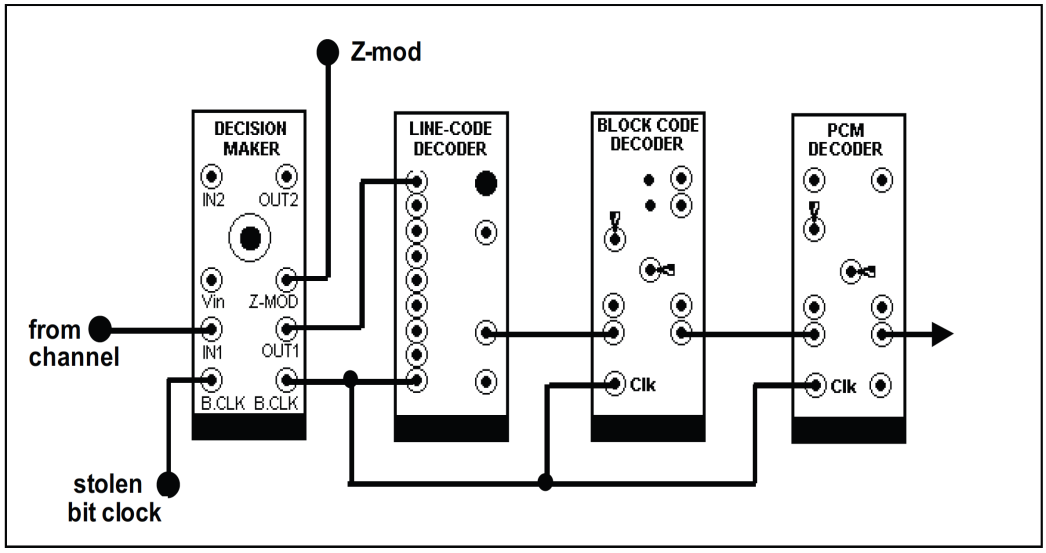



Figure 5.4: The receiver

1. patch up the model of Fig. 5.4
2. ensure that the line codes selected at both transmitter and receiver are matched
3. at the BLOCK CODE DECODER select frame synchronization using the embedded frame synch. signal, and PARITY block coding
   **ToDo** use the oscilloscope persistence mode to show the eye diagram at the output of the channel and find the best sampling instance (taken at z-mode of the DECISION MAKER) to be very close to the middle of the eye
4. at the PCM DECODER select 4-bit LINEAR decoding and frame synchronization using the embedded frame synch signal
5. on CH-A check the input of each block while on CH-B check its output

6. on CH-A check the data transmitted signal while on CH-B check the output of the PCM DECODER
   **ToDo** take a periodic message from the SYNC. MESSAGE socket and repeat step 6 for parity codes

7. on CH-A connect FS while on CH-B connect the error detection INDICATOR and monitor 13 frames

8. decrease the SNR substantially and draw your conclusion by monitoring the error detection INDICATOR

9. change the type of coding by choosing the Hamming code (at both the transmitter and the receiver and repeat steps 6 & 7 & 8. Here, the error correction INDICATOR is to be used for the Hamming code
   **ToDo** define at which SNR you can detect the signal error-free using both Hamming and parity codes, which one is better? justify your answer!

# Appendix A

# FIXED MODULES USER INSTRUCTIONS

## A.1 Buffer Amplifiers

Two independent variable gain amplifiers are provided.

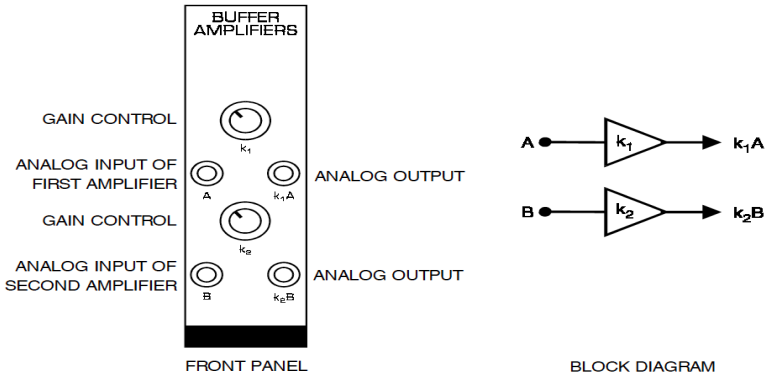



Figure A.1: Buffer amplifiers module

### A.1.1 Use

These buffers may be used to amplify small signals or attenuate large signals. Each amplifier has its own gain control on the front panel. Care should be taken to ensure that later modules are not overloaded due to excessive gain. Overload causes no damage but means non-linear operation. If overload

occurs, turn the gain control counter clockwise.

### A.1.2 Basic Specifications

- Bandwidth DC to approx 1MHz
- Gain 0 to 10

## A.2 Frequency Counter

The TIMS counter is an 8 digit, 10MHz frequency and event counter.

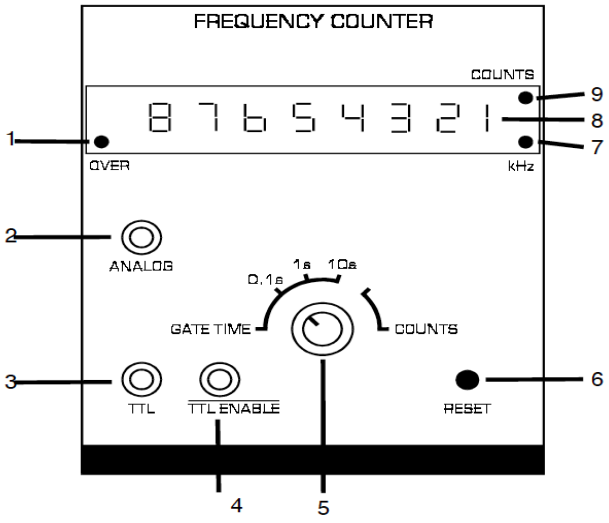



Figure A.2: Frequency Counter module

### A.2.1 Basic Specifications

1. OVERflow indication LED
2. ANALOG input:
   - Bandwidth 40Hz to 1 MHz
   - Sensitivity 250mV typically, at 100kHz
   - Maximum input ± 12V
3. TTL Input:

- Bandwidth: DC to 10MHz
- Input: TTL level signals only

4. TTL ENABLE: may be used to gate the TTL input signal, it has the same specifications of the TTL input.
5. Mode and Range rotary switch:
   - Frequency counter mode: Gate time selection of 0.1s, 1s or 10s with reading in kHz
   - Event counter mode: displays number of pulses counted since the last RESET
6. RESET Push Button: resets the count of the Event Counter to zero
7. kHz LED: is lit when counter is in FREQUENCY COUNTER mode
8. 8 digit, 7 segment display of frequency or pulse counts; maximum display 99999999
9. COUNTS LED: is lit when counter is in EVENT COUNTER mode

## A.3 Master Signals

Five synchronized analog and digital signals are available, ranging from 2kHz to 100kHz. The function and frequency of each signal is indicated on the front panel.

### A.3.1 Use

Signals are labelled as follows:

- CARRIER signals are 100kHz, which for modelling purposes is sufficiently far from the audio channel bandwidth of 3kHz.
- SAMPLE CLOCK of 8.3kHz, which may be used to sample bandwidth-limited (3kHz) audio message signals.
- MESSAGE provides an audio frequency signal which is synchronized to a sub-multiple of the carrier to enable text-book like displays of simple modulation schemes to be achieved.

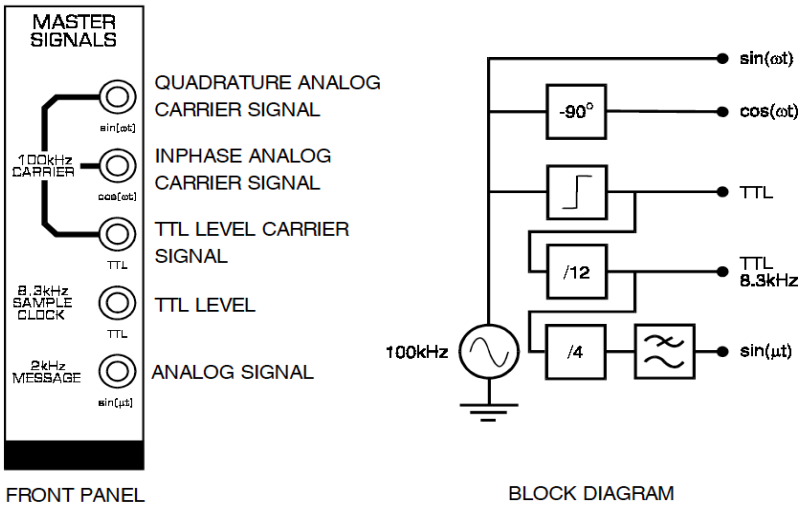



Figure A.3: Master signals module

The five signals are derived from a master crystal oscillator resulting in low frequency drift. Their frequencies are fixed internally. The output levels are also fixed. To vary the amplitude, the signals may be applied to the neighboring buffers. The analog signals are sinusoidal in shape, having a distortion of less than 0.1%. Digital signals are all standard TTL level, with rise times of better than 80nsec.

### A.3.2 Basic Specifications

- Output Frequencies 100kHz (carrier), 8.333kHz (sample clock), 2.083kHz (audio, carrier sub-multiple)
- Output Levels 4V pk-pk for analog and TTL level for digital
- Distortion $< 0.1\%$, analog outputs only

## A.4 Scope Selector

It allows 2 of 4 different signals to be viewed simultaneously on a 2 channel oscilloscope. A third input labeled TRIG is ideal for connecting a trigger signal to the oscilloscopes external trigger input.

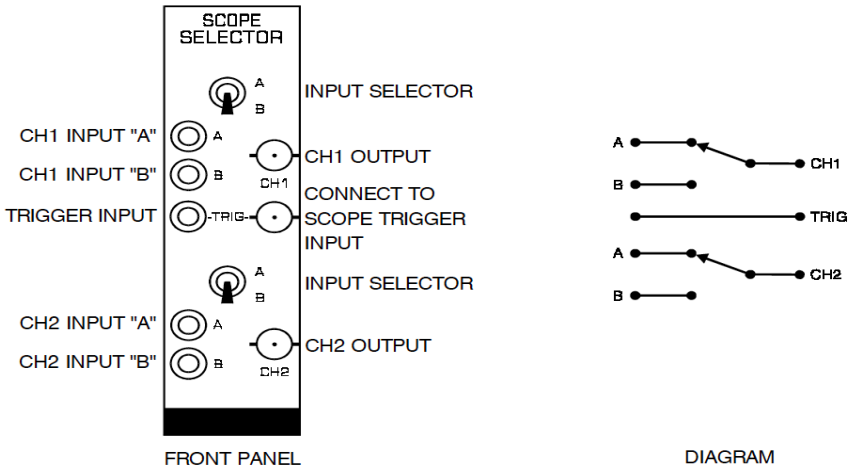



Figure A.4: Scope selector module

### A.4.1 Use

Connection to the oscilloscope is via BNC sockets. Inputs are standard 4mm sockets. Although the input sockets are yellow (analog), either analog or digital signals may be examined.

## A.5 Variable DC

It is a stable, bipolar DC source.

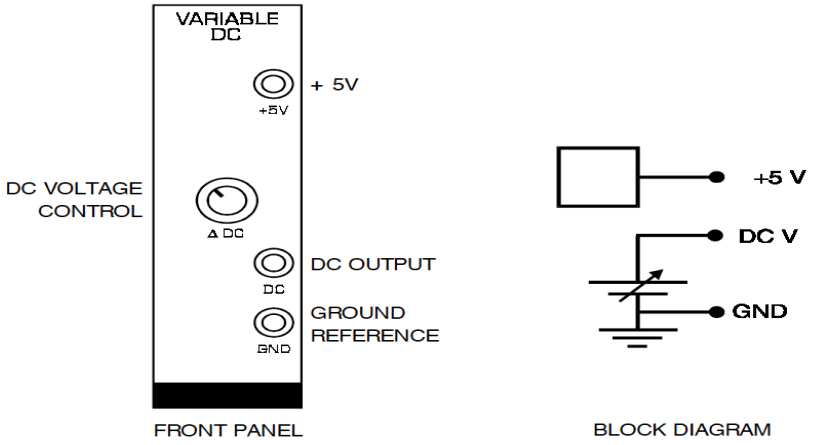



Figure A.5: Variable DC module

### A.5.1 Use

The DC voltage output varies from about -2.5V when the control is fully counter clockwise through zero to + 2.5V when control is turned fully clockwise. If greater resolution or wider range is required, then one of the buffer amplifiers can be used in conjunction with this module.

### A.5.2 Basic Specifications

- Voltage Range $\pm$ 2.5V DC
- Short-term Stability $<$ 2mV/hr
- Resolution approx 20mV
- Output Current $<$ 5mA

# Appendix B

# BASIC MODULES USER INSTRUCTIONS

## B.1 Adder

Two analog input signals $A(t)$ and $B(t)$ may be added together, in adjustable proportions $G$ and $g$. The resulting sum is presented at the output.

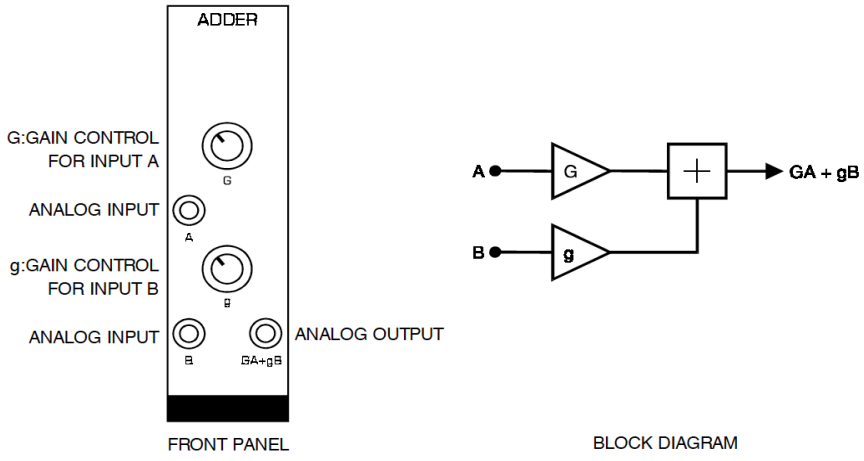



Figure B.1: Adder module

### B.1.1 Use

When adjusting the gains, avoid overloading the following modules. Overloading causes no damage but means non-linear operation. The ADDER is

capable of delivering a signal well in excess of the standard reference level, 4V pk-pk, given a standard level input.

The ADDER can also be used as a normal amplifier by using only one input and turning the gain of the other input to minimum. Note that gains G and g are negative. All inputs and outputs are DC coupled.

### B.1.2 Basic Specifications

- Gain Range $0 < G < 2; 0 < g < 2$;
- Bandwidth approx 1MHz
- Output DC Offset $<$ 10mV, open circuit inputs

## B.2 Audio Oscillator

A low distortion tuneable frequency sinewave source with a frequency range from 500Hz to 10kHz. Three outputs are provided. Two outputs are sinusoidal, with their signals in quadrature. The third output is a digital TTL level signal.

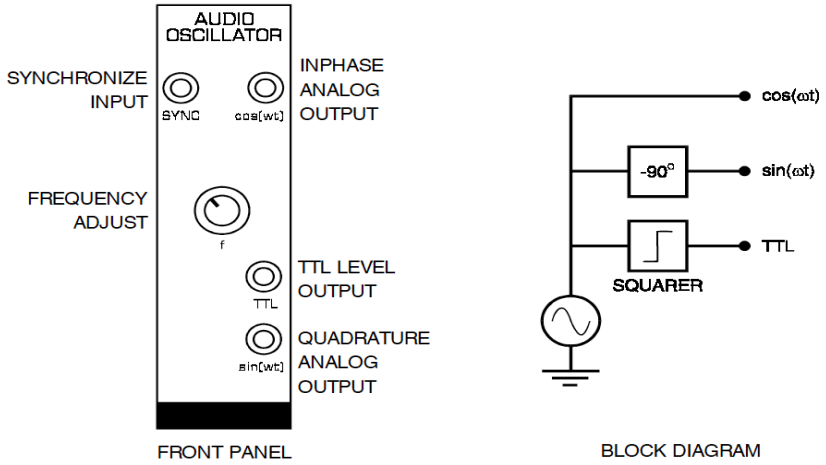



Figure B.2: Audio oscillator module

### B.2.1 Use

The frequency of each of the three outputs is the same and is varied by the front panel $\Delta f$ control. Both the in-phase and quadrature analog output signals have fixed amplitude. Their shape is sinusoidal, having a distortion of less than 0.1%.

This module may be synchronized to an external periodic signal by connecting such a signal to the front panel SYNC input. A signal of about 1 volt peak is adequate for this purpose. For synchronization to be achieved, this module must be manually tuned to within a few percent of the frequency to which synchronization is desired.

### B.2.2 Basic Specifications

- Frequency Range 300Hz to 10kHz
- Analog Output Level 4V pk-pk
- Distortion $< 0.1\%$ analog outputs only
- Digital Output TTL level

## B.3 Dual Analog Switch

Two identical analog switches are controlled by digital, TTL level signals. The outputs of the two switches are added internally and presented at the output of the module.

### B.3.1 Use

Each switch may be closed independently by a TTL HIGH at the respective control input. The switch outputs are combined internally and are presented at the common output socket. Open circuit voltage gain between each input and the module output is unity when the switch is closed.

### B.3.2 Basic Specifications

- Analog Input Bandwidth $>$ 300kHz
- Maximum CONTROL clock $>$ 100kHz

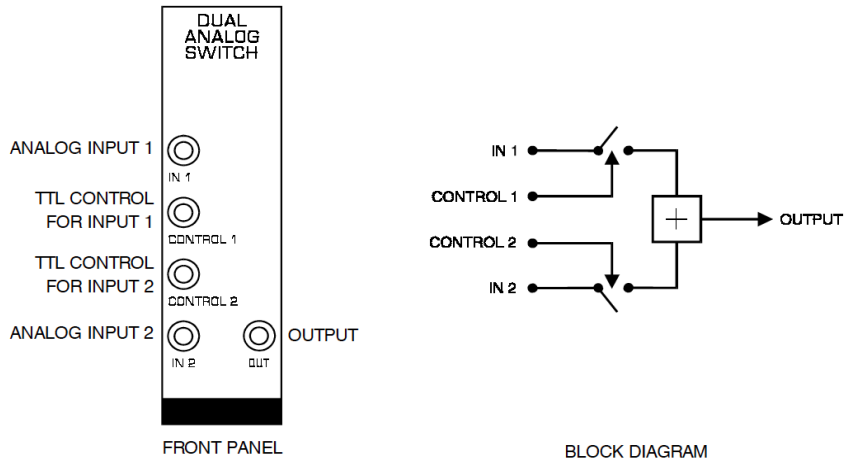



Figure B.3: Dual analog switch module

- CONTROL Input Levels TTL only
- Maximum Analog Input Level $\pm$ 8V

## B.4 Multiplier

Two analog input signals $X(t)$ and $Y(t)$ may be multiplied together. The resulting product is scaled by a factor of approximately 1/2 so that, with standard level inputs, later stages are not overloaded.

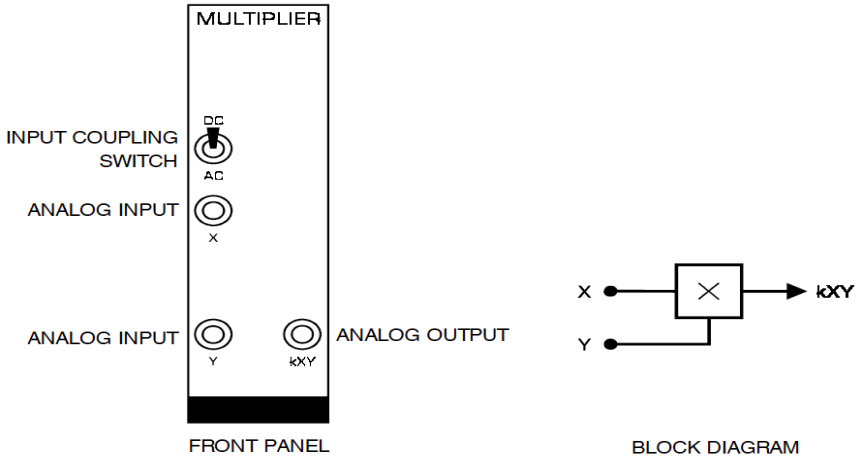



Figure B.4: Multiplier module

### B.4.1 Use

The input coupling switch may be used to remove input DC components by switching to AC coupling. The $k$ factor (a scaling parameter associated with four quadrant multipliers) is approximately one half and may be measured experimentally.

### B.4.2 Basic Specifications

- Bandwidth approx 1MHz
- Characteristic $kX(t)Y(t)$
- $k$ approx $1/2$

## B.5 Phase Shifter

It introduces a phase shift between its input and output. This phase shift is adjustable by the user. The frequency range of operation can be selected by on-board switch.

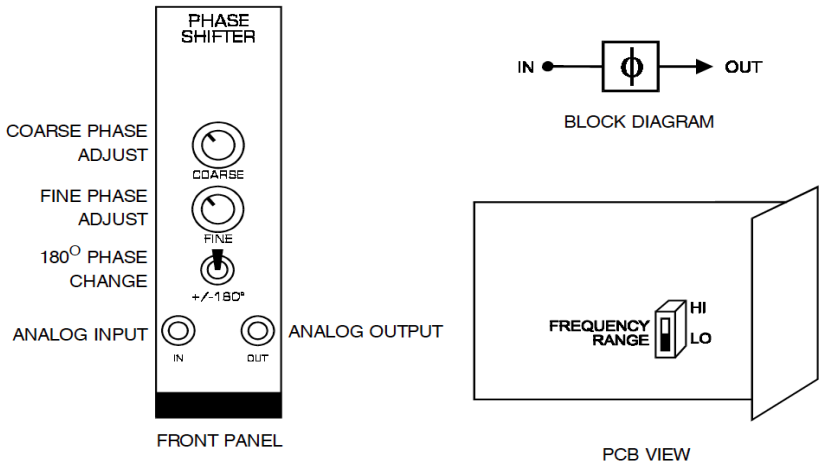



Figure B.5: Phase shifter module

### B.5.1 Use

This variable phase shifter is capable of varying the magnitude of the phase shift through 360 degrees in two steps. The 180 degree switch selects the

step or region of interest; the COARSE and FINE controls are used to then obtain the required phase shift, $\phi$.

If the input is $\cos(\mu t)$, then the output is $\cos(\mu t - \phi)$, where $\phi$ lies between 0 and 180 degrees. Although the Phase Shifter will operate from a few hertz up to 1MHz it has been optimized to operate in the neighborhood of two frequencies: around 100kHz in the HI range and around 2kHz in the LO range. An on-board switch is used to select the frequency range.

The open circuit gain through the phase shifter is essentially unity for all phases, but note that the amount of phase shift, $\phi$, is a function of frequency. This is not a wideband phase changer: thus all the frequency components of a complex signals spectra are not shifted by the same phase.

### B.5.2 Basic Specifications

- Bandwidth $<$ 1MHz
- Frequency Range HI approx 100kHz, LO approx 2kHz (For 0 to 360 degree range of phase shift. The phase shift range increases i.e. resolution decreases as the input frequency increases)
- Coarse approx 180 degrees shift
- Fine approx 20 degrees shift

## B.6 Sequence Generator

Using a common external clock signal, the sequence generator outputs two independent pseudorandom sequences $X$ and $Y$. A SYNC output is provided which is coincident with the start of the sequences. The sequences may be stopped and restarted at any time via front panel controls. Sequences $X$ and $Y$ are available as either standard TTL or analog level output.

### B.6.1 Use

An external clock signal must be provided. This may be sinusoidal or TTL: separate input sockets are used.

The sequences may be stopped at any time by either depressing the RESET button or applying a TTL HI signal to the RESET input. To

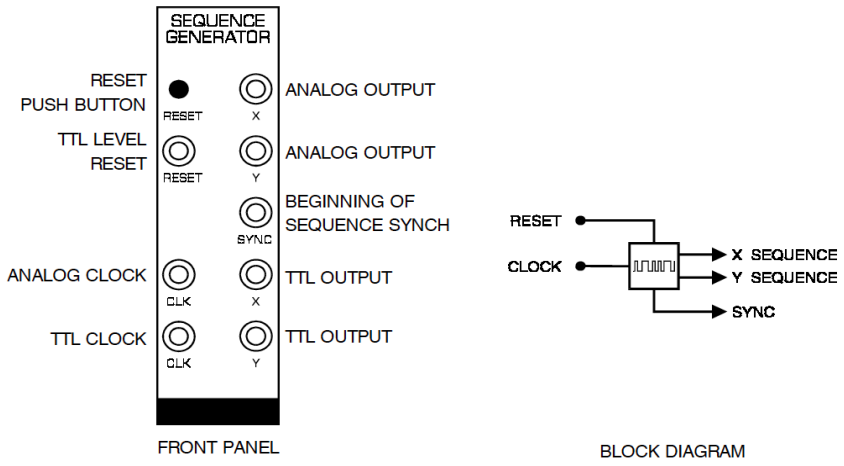



Figure B.6: Sequence generator module

restart the sequences from the beginning, release the RESET button or apply a TTL LO to the RESET input.

The length of the sequences may be selected by a PCB mounted dip switch. Four independent sequence pairs are available from lengths of $2^5$ to $2^{11}$. The sequences are selected as follows:

| DIP Switch Code | | $n$ | Sequence length $2^n$ |
|---|---|---|---|
| msb 0 | 0 | 5 | 32 |
| 0 | 1 | 8 | 256 |
| 1 | 0 | 8 | 256 |
| 1 | 1 | 11 | 2048 |

### B.6.2 Basic Specifications

- Input Clock Range TTL 1Hz to 1MHz
- Number of Sequences 4 pairs
- Sequence Lengths $2^5, 2^8, 2^8, 2^{11}$
- Sync indicates start of sequence

## B.7 Tuneable LPF

The cutoff frequency of this LPF can be varied using the TUNE control. Two frequency ranges, WIDE and NORMAL, can be selected by a front panel switch. The GAIN control allows signal amplitudes to be varied if required.

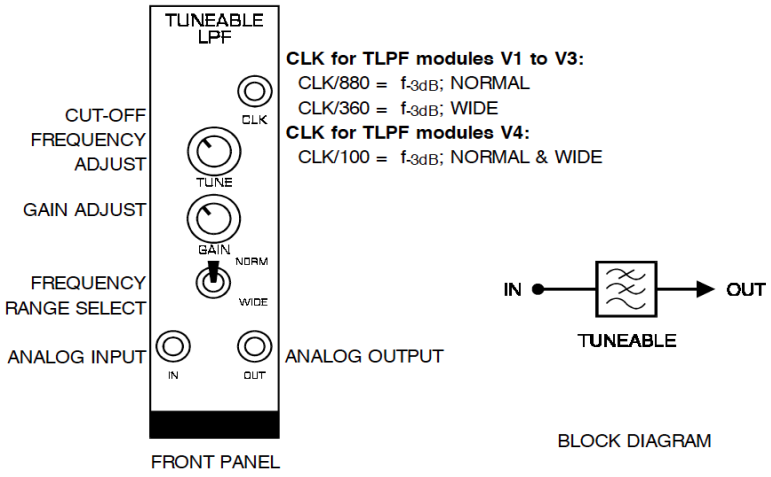



Figure B.7: Tuneable LPF module

### B.7.1 Use

This lowpass filter has an elliptic filter characteristic. The stopband attenuation is typically 50dB and passband ripple is approximately 0.5dB.

The GAIN control is used to vary the amplitude of the output signal. Two frequency ranges are provided. NORMAL range provides more precise control over the lower audio band, used for telecommunications message channels. The WIDE range expands the filters range to above 10kHz. The CLK output provides an indication of the filters cutoff frequency.

### B.7.2 Basic Specifications

- Filter Ranges 900 Hz < NORMAL < 5 kHz and 2.0 kHz < WIDE < 12 kHz
- Filter Order 7th order, Elliptic
- Stopband Attenuation > 50dB and Passband Ripple <0.5dB

## B.8 Twin Pulse Generator

A positive going edge applied at the CLOCK input causes a positive pulse to occur at the output terminals. There are two operating modes: TWIN and SINGLE. Only TWIN mode is limited to low frequency CLOCK inputs.

In TWIN mode, $Q_1$ outputs the leading pulse and $Q_2$ outputs the delayed pulse. The time between pulses $Q_1$ and $Q_2$ can be varied, as can the pulses widths.

In SINGLE mode, only $Q_1$ outputs a positive going pulse, while $Q_2$ outputs the inverse of $Q_1$. The pulse width can be varied.

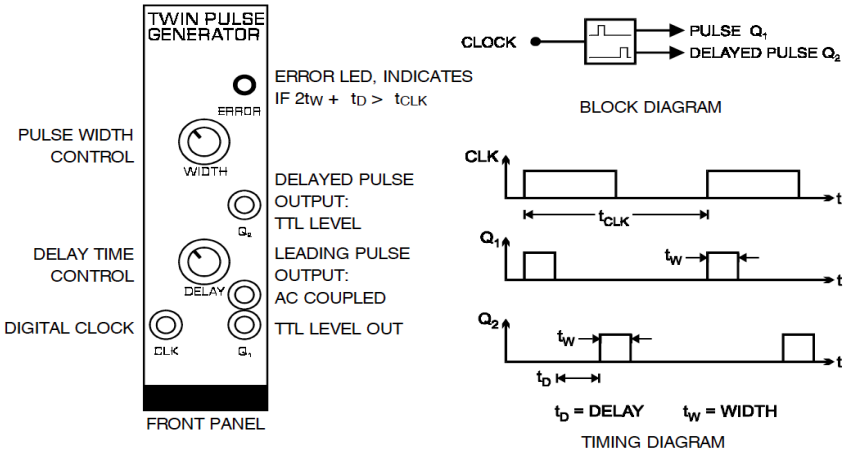



Figure B.8: Twin pulse generator module

### B.8.1 Use

A digital TTL level signal is applied to the CLK input. The GENERATOR then outputs one or two pulses, depending upon the operating mode selected. Use the PCB mounted MODE switch to select either SINGLE or TWIN operating mode.

**Twin mode**

It is used when two sequential pulses are needed. Two equal width positive pulses occur as a result of each CLK signal positive edge. Pulse $Q_1$ always occurs before pulse $Q_2$. The width of both pulses is controlled by the front panel WIDTH control. The DELAY control varies the spacing between the two pulses. Note that TWIN mode will only accept CLOCK input

signals of up to 50kHz, depending upon front panel settings. If WIDTH and DELAY have been incorrectly set, causing anomalous operation, the ERROR LED will be lit. To eliminate the error reduce DELAY and then WIDTH - by turning counter clockwise.

**Single mode**

It is used to obtain a train of equal width pulses from any TTL level signal. Equal width positive pulses occur at Q1 output as a result of each CLK signal positive edge. The width of the pulses is controlled by the front panel WIDTH control. Q2 simultaneously outputs the compliment of $Q_1$. The DELAY control is not used in this mode. Note that $Q_1$ includes both a TTL level and an AC coupled output pulse.

### B.8.2 Basic Specifications

**Twin mode**

- Clock Frequency Range $<$ 50kHz
- Pulse WIDTH $3\mu s < t_w < 25\mu s$
- Pulse DELAY $(Q_2 - Q_1)$ $10\mu s < t_d < 120\mu s$
- Error Indication $2t_w + t_d > t_{CLK}$

**Single mode**

- Clock Frequency Range $<$ 200kHz
- Pulse WIDTH $3\mu s < t_w < 25\mu s$

## B.9 Utilities

The Utilities Module houses 4 independent functional blocks:

1. A signal COMPARATOR with TTL output and CLIPPER with bipolar output, for squaring analog waveforms. The threshold of the COMPARATOR threshold level may be set as required by applying a DC voltage to the REF input. The gain of the CLIPPER gain may be set by adjusting DIP switches SW1 and SW2.

2. Precision halfwave RECTIFIER.

3. Simple diode and single pole, audio range, RC Lowpass Filter.

4. Single pole, audio range, RC Lowpass Filter.

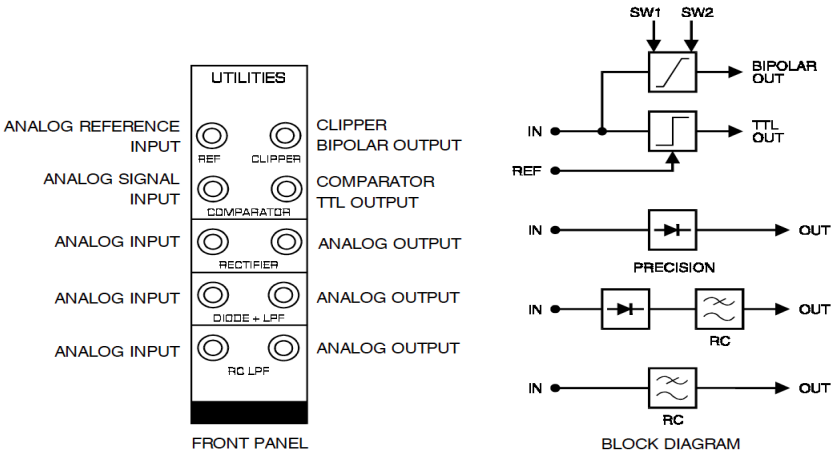



Figure B.9: Utilities module

### B.9.1 Use

**Comparator**

it will square any analog signal and provide a standard TTL level output. The switching threshold level is determined by the voltage level applied to the REF input. The REF input may be connected to GROUND, VARIABLE DC or any other signal source.

**Clipper**

it will amplify any analog TIMS level signal and then clip the amplitude of the amplified signal, to a fixed level of approximately $\pm$ 1.8V. The clipping action is performed by standard small signal diodes. The REF input is not used by the clipper.

### B.9.2 Basic Specifications

**Comparator**

- Operating Range > 500kHz
- TTL Output Risetime 100ns (typ)

**Clipper**

- Operating Range > 500kHz
- Output Level 1.8Vpk (typ)
- Adjustable Gains 3 steps; x0.8, x8 and x40 (approx)

**Rectifier**

- Bandwidth DC to 500kHz (approx)

**Diode and LPF**

- LPF -3dB 2.8kHz (approx)

**RC LPF**

- LPF -3dB 2.8kHz (approx)

## B.10 Voltage Controlled Oscillator

This module functions in two modes: either as a voltage controlled oscillator with analog input voltage or as an FSK generator with digital input. Both modes have two frequency ranges of operation which are selected by a range switch. The VCO frequency and input sensitivity can be controlled from the front panel.

### B.10.1 Use

**VCO mode**

The VCO output frequency is controlled by an analog input voltage. The input voltage, $V_{\text{in}}$, is scaled - amplified - by the front panel GAIN control. A DC voltage can be added to $V_{\text{in}}$ internally, thus setting the start or CENTER

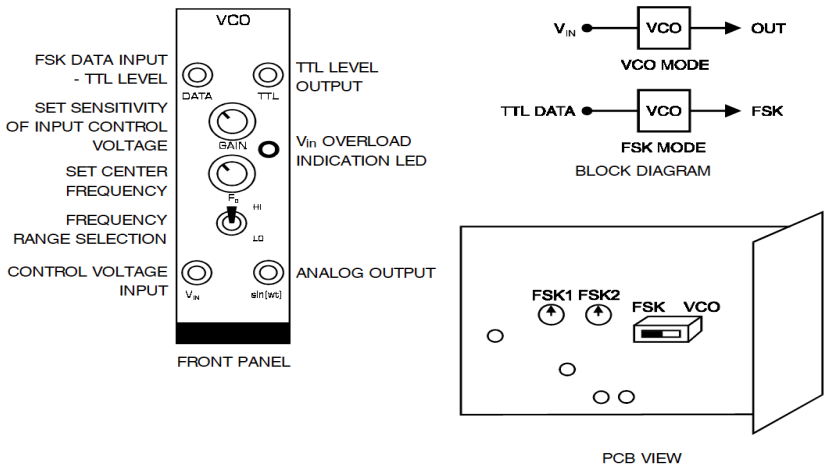



Figure B.10: Voltage controlled oscillator module

FREQUENCY, $f_0$. The CENTER FREQUENCY is defined as the VCO output frequency, when no voltage is applied. The $V_{\text{in}}$ input is internally tied to ground if no signal is applied. The $V_{\text{in}}$ OVERLOAD LED is lit when the sum of these voltages - scaled $V_{\text{in}}$ plus CENTER FREQUENCY DC offset - exceed the oscillators internal operating limits. Decrease the GAIN - turn counter clockwise - and/or shift the CENTER FREQUENCY, $f_0$ , to extinguish the LED. The frequency range switch selects between the HI or carrier band and the LO or audio band. Both sinewave and digital outputs are available.

**FSK mode**

A PCB mounted slide switch selects between FSK and VCO modes of operation. The two output frequencies, FSK1 and FSK2, (MARK and SPACE), are set by varying the PCB mounted, finger adjustable trimmers. As in VCO mode, the frequency range switch selects between the HI or carrier band and the LO or audio band. The digital data input accepts only TTL level signals. Both sinewave and digital outputs are available. GAIN and CENTER FREQ, $f_0$, controls and the $V_{\text{in}}$ connector are not used in the FSK mode.

### B.10.2 Basic Specifications

**VCO mode**

- Frequency Ranges 1.5kHz < LO < 17kHz; sinewave and TTL ( < 300Hz with external input voltage, $V_{\text{in}}$ ) 70kHz < HI < 130kHz; sinewave and TTL
- Input Voltage -3V < $V_{\text{in}}$ < 3V
- Overload limit indication LED $V_{\text{VCO}} > \pm$ 3V; $V_{\text{VCO}}$ is the internal voltage finally applied to the VCO circuitry.
- GAIN $G.V_{\text{in}}$ : $1 < G < 2$
- Center Frequency Voltage Range - 3V < $V_{\text{fc}}$ < 3V; $V_{\text{fc}}$ is a DC voltage added internally to $G.V_{\text{in}}$

**FSK mode**

- Frequency Ranges low: 1.5kHz <FSK1< 9kHz , 500Hz <FSK2< 4kHz High: 80kHz < FSK1< 200kHz , 20kHz <FSK2< 120kHz
- Data Input TTL level message

# Appendix C

# ADVANCED MODULES USER INSTRUCTIONS

## C.1 Baseband Channel Filters

Four switch selectable, baseband channels are provided, comprising three different filters and one straight-through connection. Each of the three filters has a stop-band frequency of near 4kHz.

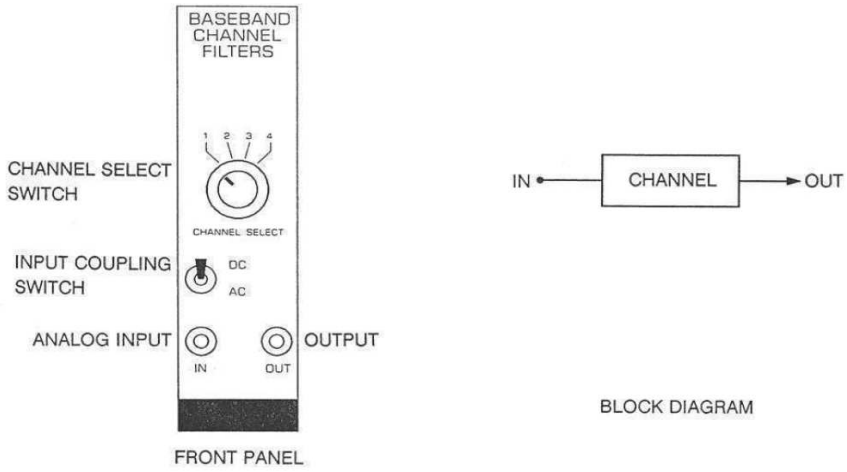



Figure C.1: Baseband Channel Filters module

### C.1.1 Use

Only one channel may be selected and used at a time. Note that each of the four channels may be AC or DC coupled by front panel toggle switch.

Actual measurement of amplitude and phase responses can be carried out using the AUDIO OSCILLATOR and TRUE RMS METER modules or an oscilloscope.

### C.1.2 Basic Specifications

- Input coupling AC or DC, channels 1 to 4
- Channel responses
  - Channel 1 straight-through
  - Channel 2 Butterworth, 7th order
  - Channel 3 Bessel, 7th order
  - Channel 4 OpFil Linear Phase, 7th order (a proprietary filter design having a sharp roll-off characteristic with a linear phase response in the passband)
- Stop-band attenuation approx 40dB, 4kHz
- Passband ripple 0.5dB

## C.2 Decision Maker

Digital signals may become corrupted by noise and interference in the communications channel. After demodulation or receiver filtering, a corrupted digital signal would need to be squared and converted to a clean digital waveform with an associated in phase bit clock, so that further digital processing, decoding or message recovery can be performed. The tasks of squaring the corrupted digital signal and aligning the bit clock can be carried out by this module.

The DECISION MAKER module accepts up to two TTL, unipolar or bipolar level, baseband digital signals and a synchronised bit clock. The input signals are sampled at a point determined by the user and are output

as clean digital signals, along with an in-phase and synchronised bit clock. Using an oscilloscope, the decision point is displayed as a bright marker on the input digital waveform.

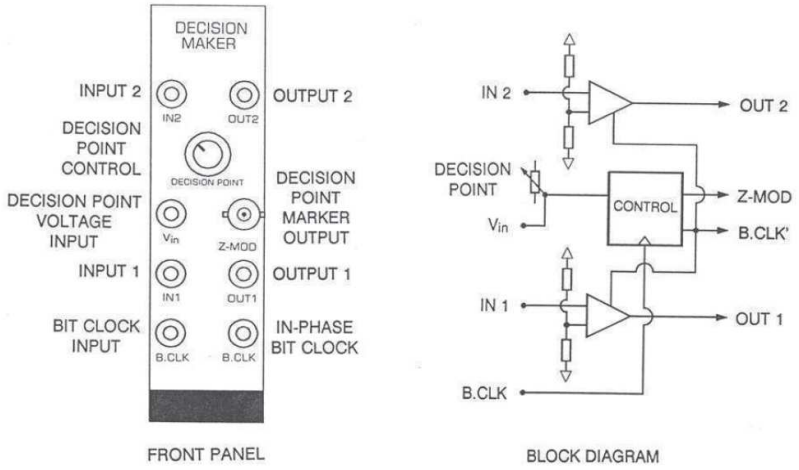



Figure C.2: Decision maker module

### C.2.1 Use

**Inputs IN1 and IN2**

IN1 and IN2 will each accept an incoming digital signal. If only one digital signal is available, then either input may be used: leave the unused input unconnected. When a digital signal is connected to each input, then both signals must have the same waveform format. Ensure the amplitudes of the input digital signals are within TIMS standard limits of ±2V bipolar, +2V, 0V unipolar and +5V, 0V TTL.

**Waveform format selection**

The correct waveform format must be selected. The user has a choice of eight bipolar and unipolar waveforms (Line-Codes) as well as standard TTL waveforms. Set switch SW1 to the required waveform format position. SW1 is a PCB mounted, ten position rotary switch at the rear of the module.

**Bit clock and outputs OUT1 and OUT2**

This module primarily operates with bit clocks of around 2kHz. The input bit clock, B.CLK, must be synchronized to the input digital signal(s) and so should be either regenerated from an input digital signal or may be

stolen from the transmitter. The output bit clock, B.CLK, is synchronized and aligned with the output bit stream(s) in the following manner: each new bit occurs on the falling B.CLK edges. The position of the output bit clocks negative edge is determined by the DECISION POINT control.

**Decision point control**

The DECISION POINT is the point at which the incoming digital data is sampled. At the sampling time, a decision is made as to whether the sample is "HI" or "LO" and the result is output to the corresponding output, OUT1 or OUT2. If a digital signal is present at each input, then both are sampled simultaneously: the results are also output simultaneously.

The user has direct control over the position of the DECISION POINT across the bit width. The threshold voltages for the decision are set by fixed resistors. The threshold voltages are listed in the TABLE below, where the default setting $V_+ \approx 1V, V_- \approx -1V, V_0 \approx 0V$. With an input bit clock of 2kHz, the DECISION POINT can be moved continuously across more than 90% of the bit width. The DECISION POINT can be moved by either front panel control (INTernal control) or by external DC voltage applied to input Vin, (EXTernal control). Sliding switch SW2, located on the PCB, selects INTernal or EXTernal control mode.

| Waveform | Thresholds | Output level | Bit width |
|---|---|---|---|
| NRZ-TTL | $V_+$ | $0, +5V$ | full |
| NRZ-L | $V_0$ | $\pm 2V$ | full |
| NRZ-M | $V_0$ | $\pm 2V$ | full |
| UNI-RZ | $V_+$ | $0, +2V$ | half |
| BIP-RZ | $V_+, V_-$ | $0, \pm 2V$ | half |
| RZ-AMI | $V_+, V_-$ | $0, \pm 2V$ | half |
| BiO-L | $V_0$ | $\pm 2V$ | half |
| DICOCE | $V_+, V_-$ | $0, \pm 2V$ | full |
| DUOBINARY | $V_+, V_-$ | $0, \pm 2V$ | full |

### C.2.2 Basic Specifications

- Digital waveform inputs: two, IN1 and IN2
- Digital waveform outputs: two, OUT1 and OUT2

- Input/Output levels: for TTL : +5V, 0V, Unipolar : +2V, 0V and Bipolar : ±2V
- Waveform format selection: by 10 position rotary switch, SW1
- Waveform formats supported: NRZ-TTL, NRZ-L, NRZ-M, UNI-RZ, BIPOLAR-RZ, RZ-AMI, BIPHASE-L, DICODE, DUOBINARY
- Bit Clock input, B.CLK: TTL level, nominally 2kHz; operational 250Hz to 3.5kHz, performance not specified
- Bit Clock output, B.CLK: synchronized to the OUTput waveform; negative Bit Clock edge aligned with each new output bit
- Decision point span > 90% of bit width, with 2kHz B.CLK
- DECISION POINT control selection: INTernal or External, by switch SW2
- DECISION POINT control: continuous, by front panel knob (INT), or, 0 to 5V DC EXTERNAL input signal (EXT) at Vin

## C.3 Error Counting Utilities

Two independent functional blocks are provided, which in conjunction with other TIMS modules, can be used to carry-out Bit Error Rate measurements. The two blocks are an Exclusive-OR gate for comparing two digital data streams and a precise monostable for gating a pulse counter.

### C.3.1 Use

**Exclusive-OR logic gate**

The X-OR logic gate accepts standard TTL input signals. It operates in two modes: normal and pulse output. In normal mode, no clock signal should be connected to the logic gates CLK input. The output will be an uninterrupted result of the X-OR gate. In pulse mode, a clock signal must be connected to the logic gates CLK input. The logic gates result will only be passed to the output during the clocks HI period. Therefore, if the logic gates result is HI (logic 1) the output will appear as one pulse or as a

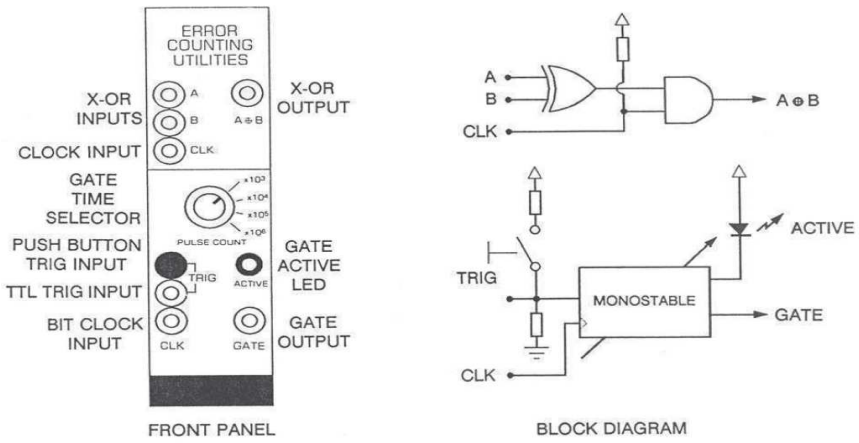



Figure C.3: Error counting utilities module

sequence of pulses if the result is HI for more than one clock cycle. Typically the clock is an in-phase and synchronised bit clock associated with the data streams being compared by the logic gate.

**Monostable**

A digital clock signal must always be connected to the CLK input. Typically this would be the bit clock associated with the digital data of the experiment being carried out.

The output GATE signal is activated, or TRIGgered, by either depressing the front panel push button switch or applying a digital level signal to the TRIG input. The output LED, labelled ACTIVE, is lit continuously while the GATE is activate and only flashes during the last 10% of the GATE period. The LED is not lit when the GATE is not active. While the output GATE is active, the Monostable may be reTRIGgered at any time, by depressing the TRIG push button or applying a signal to the TRIG input. When reTRIGgering occurs, the GATE output immediately clears (becomes inactive) and is then reactivated for the new monostable period.

The output GATE time is determined by a preselected count of input clock pulses. The number of clock pulses counted is selected initially by the PULSE COUNT front panel rotary switch. Under normal mode four GATE times are available: $10^3, 10^4, 10^5 and 10^6$ clock pulses. There are another twelve EXTENDED and sixteen EXPANDED counting modes.

### C.3.2 Basic Specifications

**Exclusive-OR gate**

- Inputs A and B: TTL level
- Output: continuous X-OR result or gated with HI time of the input CLK
- CLOCK input TTL level, $f_{\max} > 40\text{kHz}$

**Monostable**

- GATE active level: DIP switch selectable, active HI or active LO
- GATE time:
  - normal mode: $10^3, 10^4, 10^5, 10^6$
  - extended mode: normal mode x2, x4 or x8
  - expanded mode: same as normal or extended modes BUT divides the PULSE COUNT selected by twelve
- GATE output LED: continuously lit while GATE is active, flashing only during last 10% of active time
- CLOCK input: TTL level, $f_{\max} > 20\text{kHz}$
- TRIG input: depress push button, or input signal
- TRIG signal level: TTL level, DIP switch selectable active level, active HI or active LO
- TRIG signal min width $> 0.2\mu\text{s}$

## C.4 Line Code Encoder

A TTL level data stream is simultaneously encoded into eight Line-Codes (PCM Waveforms) and one Precoded Duobinary Code. The incoming data stream must be clocked by the ENCODERS bit clock output.

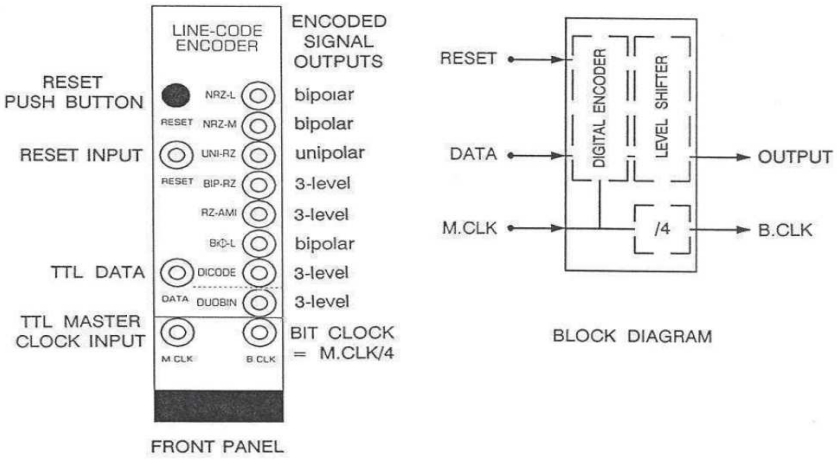



Figure C.4: Line code encoder module

### C.4.1 Use

**Master and bit clocks**

A TTL level clock should always be connected to the M.CLK (MASTER CLOCK) input. Note that the frequency of the output B.CLK signal will be one quarter of the applied M.CLK signal. A convenient M.CLK source is the 8.3kHz TTL available from the MASTER SIGNALS module. The input DATA stream should always be generated by or clocked with this modules B.CLK (BIT CLOCK) signal. Alignment between the incoming data and the B.CLK must be such that each new bit transition of the TTL data stream occurs on positive going B.CLK edges. The resulting encoded bit appears at the ENCODERS outputs on the following negative B.CLK edge. If the PSEUDORANDOM SEQUENCE GENERATOR module is used to provide the DATA, then clock the SEQUENCE GENERATOR using the ENCODER modules B.CLK output directly.

**Resetting**

Press the RESET push button once the M.CLK has been connected. If during the course of the experiment the M.CLK is interrupted, then repeat the reset procedure, by depressing the RESET push button. Resetting of the LINE-CODE ENCODER module is necessary as some Line-Codes must commence from a known initial state, for subsequent output signals to be

correctly encoded and later decoded.

**Signal levels**

The Line-Code waveforms have standard TIMS amplitude of 2Vp-p. Voltage levels used are for Unipolar (0V, +2V), Bipolar (-2V, +2V) and 3-level (-2V, 0V, +2V).

### C.4.2 Basic Specifications

**Inputs**

DATA: TTL level, digital signal M.CLK: TTL level, digital signal; $f_{\max} >$ 400kHz

**Outputs**

B.CLK: TTL level, digital signal LINE-CODE outputs: ±2Vp-p, ±10%

### C.4.3 Definitions of encoded waveform formats

- NRZ-L Non-return to zero - level; (bipolar) 1 : high level; 0 : low level.
- NRZ-M Non-return to zero - mark; (bipolar)1 : transition at beginning of interval; 0 : no transition.
- UNI-RZ Unipolar return to zero; (unipolar) 1 : pulse in the first half of the bit width; 0 : no pulse.
- BIP-RZ Bipolar return to zero; (3-level) 1 : positive pulse in the first half of the bit width; 0 : negative pulse in the first half of the bit width.
- RZ-AMI Return to zero - alternate mark invert; (3-level)1 : pulse in the first half of the bit width, alternating polarity pulse to pulse; 0 : no pulse.
- BiO-L Biphase - level ( Manchester ); (bipolar) 1 : transition from high to low in the middle of the bit interval; 0 : transition from low to high in the middle of the bit interval.

- DICODE-NRZ Dicode - nonreturn to zero; (3-level) 1 to 0, or 0 to 1 transition : change in pulse polarity; 1 to 1, or 0 to 0 transition : no pulse.
- PRECODED DUOBINARY (PARTIAL RESPONSE SIGNALING); (3-level)

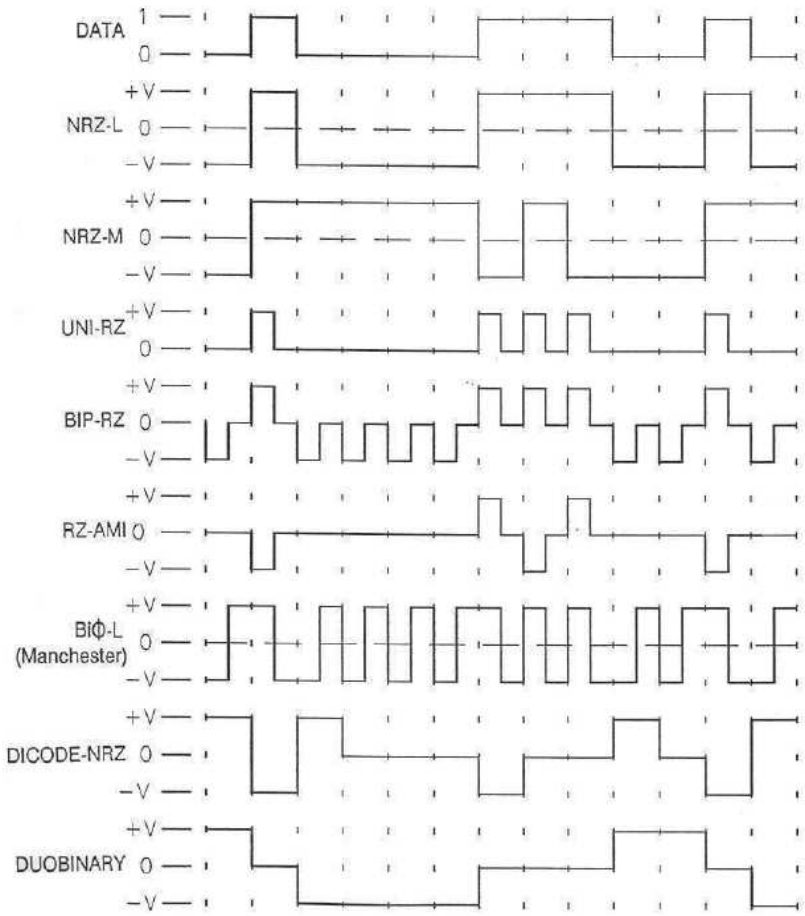



Figure C.5: Encoded waveforms

## C.5 Line Code Decoder

Each of the encoded signals generated by the LINE-CODE ENCODER module can be decoded, producing a TTL level data stream. A synchronized bit clock with correct alignment must be provided to the DECODER.

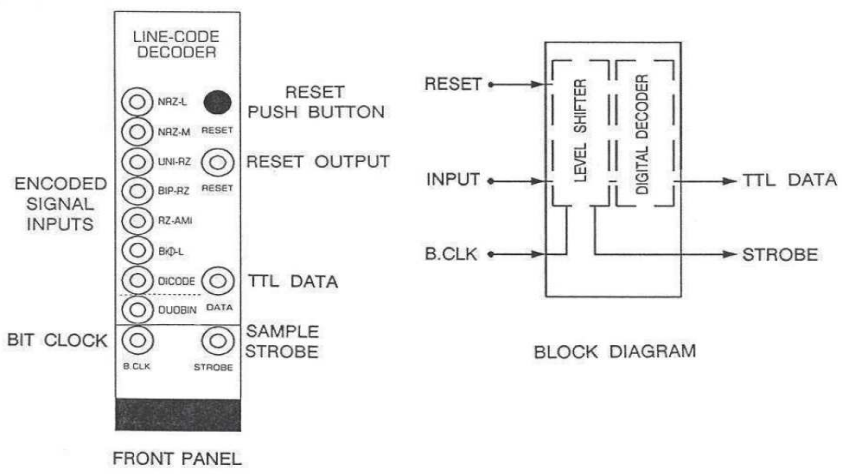



Figure C.6: Line code decoder module

### C.5.1 Use

The incoming encoded signal must be clean and distortion free. The task of cleaning and squaring a recovered signal must be carried-out beforehand, by other modules, such as the TIMS DECISION MAKER. Only one encoded signal may be applied to any DECODER input at any one time.

A TTL level clock must always be connected to the B.CLK, (BIT CLOCK) input. The B.CLK signal must be synchronised and aligned to the incoming encoded bit stream in the following manner: each new bit transition of the incoming encoded data stream occurs on falling B.CLK edges.

The STROBE output is derived from the incoming B.CLK. The positive going edge of the STROBE output is the exact moment the DECODER samples the incoming signal for the decoding process: the decoded TTL output data is then immediately available at the DATA output.

The DECODER module requires resetting after the B.CLK or input waveform has been applied or interrupted. Resetting of the LINE-CODE ENCODER module is necessary as some Line-Codes must be decoded from a known initial state, for subsequent output data to be correct.Two equivalent methods of resetting the ENCODER/DECODER pair are available, the first method is the automatic resetting of both modules, in which we patch the DECODERS RESET output to the ENCODERS RESET input,

then momentarily depress either the ENCODERS or DECODERS RESET push button. The second method is manual resetting of each module, in which we hold down the ENCODERS RESET push button, while momentarily depressing the DECODERS RESET push button, then release the ENCODERS RESET push button.

### C.5.2 Basic Specifications

**Inputs**

B.CLK: TTL level bit clock, synchronized to the input data; $f_{\max} >$ 100kHz Encoded signal inputs: definitions are given in the line code encoder description

**Outputs**

DATA: decoded TTL level data STROBE: TTL level signal

## C.6 Noise Generator

A broadband noise source, with a 12 step output amplitude attenuator.

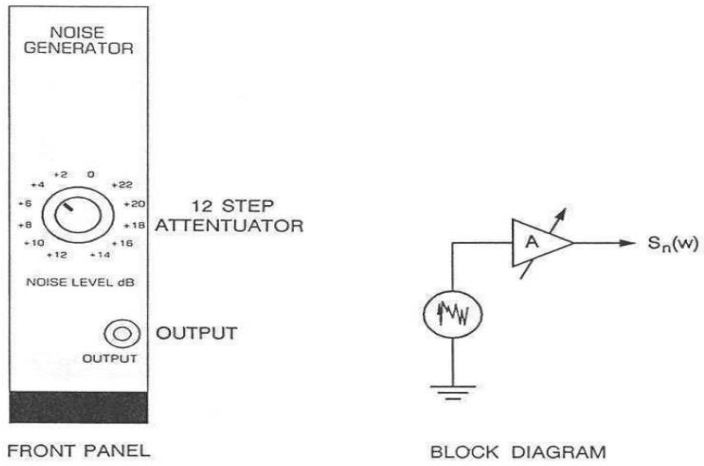



Figure C.7: Noise generator module

### C.6.1 Use

The module requires no input or control signals. The output noise level can be varied in discrete steps of 2dB, where the minimum noise level is at 0dB

and maximum noise level is at +22dB.

If required, the characteristics of the output noise signal can be altered by: filtering, using any of the TIMS filter modules; or attenuated or amplified, using the TIMS BUFFER AMPLIFIER or ADDER modules.

### C.6.2 Basic Specifications

- Bandwidth 1Hz to < 500kHz
- Maximum level approx 1Vrms at +22dB position
- Attenuator steps 12 steps, 0dB to +22dB ( 2dB per step )
- Attenuator accuracy < ±0.25dB to any two adjacent steps ( ±0.1dB typically ) < ±0.35dB between any two steps

## C.7 Wideband True RMS Volt Meter

A wideband, true RMS volt meter with large, LED digital display and a buffered DC output.

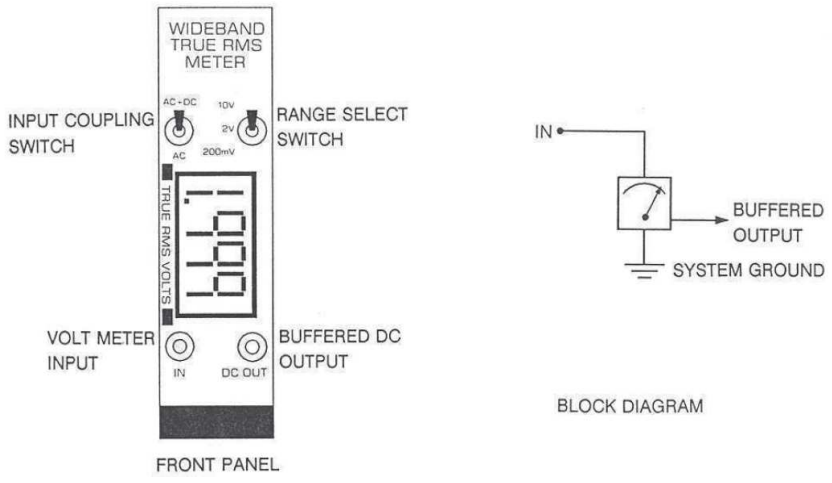



Figure C.8: Wideband true RMS volt metermodule

### C.7.1 Use

The input signal may include AC and DC components. If only the AC components of the signal are to be measured, then select the AC coupling.

Otherwise select AC+DC coupling. Before connecting any input signals, always select the 10V range first. If greater resolution is required, then select the lower ranges, 2V or 200mV.

The DC OUTPUT provides a standard TIMS level, buffered DC voltage, which is directly proportional to the digital displays reading.

### C.7.2 Basic Specifications

- Input ranges: three ranges 200mV, 2V and 10V
- Crest factor 8:1 (peak voltage to RMS voltage)
- Maximum allowable input: 15V peak, all ranges
- Input impedance: 100k ohm in parallel with less than 100pF
- Bandwidth: DC, 100Hz to 1.2MHz
- DC output: approximately 1mV DC per digit, giving 2V full scale

## C.8 100kHz Channel Filters

Three switch selectable, 100kHz channels are provided, comprising two different filters and one straight-through connection.

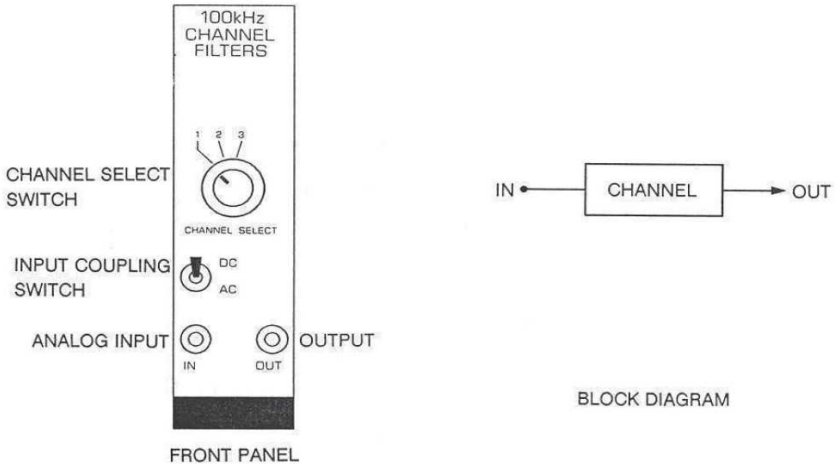



Figure C.9: 100kHz channel filters module

### C.8.1 Use

Only one channel may be selected and used at a time. Note that each of the three channels may be AC or DC coupled by front panel toggle switch. Each channel should be characterized by actual measurement of amplitude and phase responses using the VCO and TRUE RMS METER modules or an oscilloscope.

### C.8.2 Basic Specifications

- Input coupling: AC or DC, channels 1 to 3
- Channel responses
  - Channel 1 straight-through
  - Channel 2 bandpass filter
  - Channel 3 lowpass filter
- Stop-band attenuation: approx 40dB

## C.9 Bit Clock Regeneration

Four independent functional blocks are provided, which may be used independently or in combination with other TIMS modules, to recover the bit clock of any TIMS generated Line-Code.

### C.9.1 Use

**Divide by N**

The DIVIDE BY N is a general purpose digital divider. It accepts a standard TTL level signal at the input and outputs a standard TTL level signal. The PCB mounted DIP switch, SW2, is used to select the division factor, as illustrated in the table below.

| SW2-1 (A) | SW2-2 (B) | DIV. MODE |
|---|---|---|
| OFF | OFF | divide by 8 |
| OFF | ON | divide by 4 |
| ON | OFF | divide by 2 |
| ON | ON | divide by -1 (invert) |

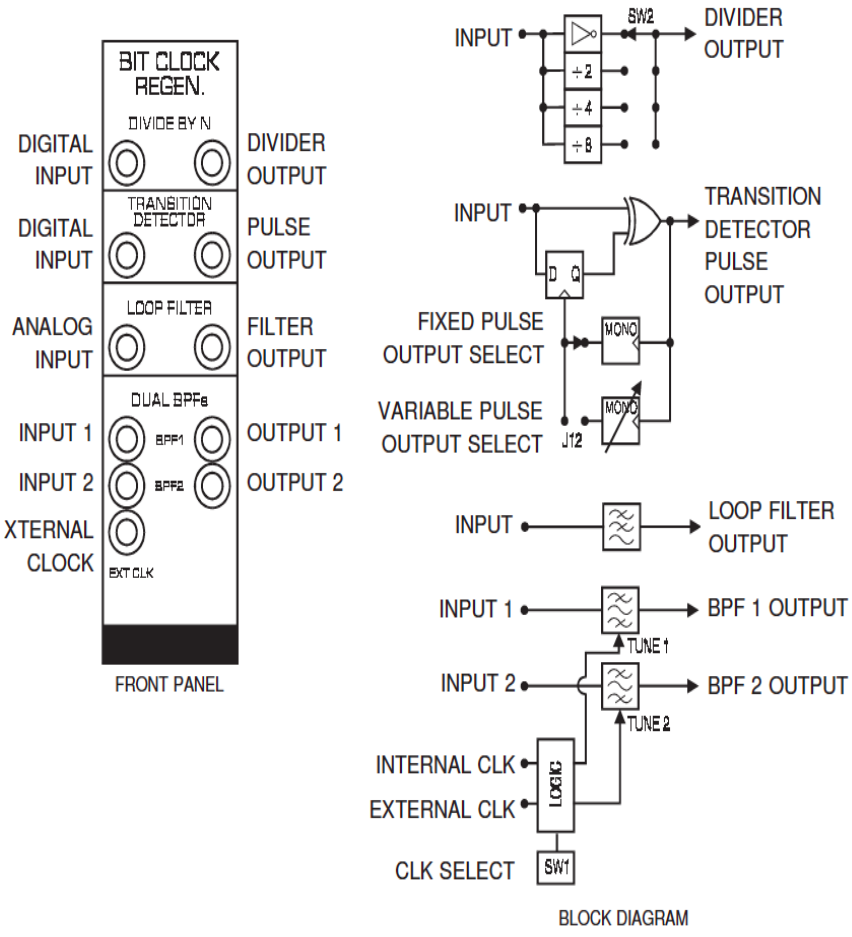


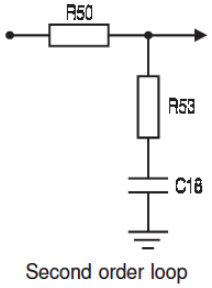

Figure C.10: Bit clock regeneration module

**Transition detector**

The TRANSITION DETECTOR will produce a TTL level output pulse for every transition in logic level of the input digital sequence. The input sequence must be TTL level. Operation of the TRANSITION DETECTOR is such that the input sequence is delayed using a clocked flip-flop. The exclusive-OR circuit then performs the equivalent of a multiplication operation. The width of the output pulse is dependent upon the width of the monostables pulse.

**Loop filter**

The LOOP FILTER is intended for use in Phase Lock Loop, PLL, applications such as demonstrating PLL bit-sync derivation. It is a conventional, passive, Type 1, second-order loop structure, as illustrated below. The factory selected component values are also given.




| Label | Value |
|---|---|
| R50 | 9k1R |
| R53 | 1k9R |
| C18 | 100nF |

Figure C.11: Loop filter

**Dual BPFs**

Two independent, tuneable, high-Q bandpass filters are provided. Each filter accepts and outputs standard TIMS level signals. Both filters have the same fixed Q of 22. The centre frequency of each filter is controlled by a digital clock signal. The frequency of the digital clock signal is 50 times the centre frequency of the BPF. The source of the digital clock signal may be either the internal (on-board) crystal oscillator or an external oscillator.

The PCB mounted DIP switch, SW1, is used to select each filters clock source. The internal crystal derived clock, INT.CLK, is optimized for use with the LINE-CODE ENCODER modules standard 2.083kHz bit clock. The external clock, EXT.CLK, may be used to tune the centre frequency of either or both of the filters between 1kHz and 5kHz. The external TTL

level clock source is applied via the front panel EXT CLK input. The table below lists all possible combinations of clock source for both filters.

| SW1-1 | SW1-2 (B) | BPF1 source | BPF2 source |
|---|---|---|---|
| OFF | OFF | External | External |
| OFF | ON | External | Internal |
| ON | OFF | Internal | External |
| ON | ON | Internal | Internal |

### C.9.2 Basic Specifications

**Divide by N**

- Input and Output TTL level, digital signals
- Clock input $<$ 1MHz
- Divisors -1, 2, 4 and 8, switch selectable

**Transition detector**

- Input and Output TTL level, digital signals

**Loop filter**

- Input and Output: standard TIMS level, analog signals
- Type: conventional, passive, Type 1, second-order loop structure

**Dual bandpass filters**

- Input and Output standard TIMS level, analog signals
- Number: two identical bandpass filters
- Type: fourth order Chebyshev with 3dB passband ripple
- Q: approx. 22, fixed
- Ratio of Tuning Clock to Filters Centre Frequency: 50
- Internal Clock Frequency: 104kHz, crystal derived, giving 2.083kHz filter centre frequency
- External Clock Frequency Range: 50kHz to 250kHz, TTL level

## C.10 M-Level Encoder

A continuous sequence of TTL-level data bits is grouped into sets of L bits, (where L = 2, 3 or 4). Each set of L bits is encoded to form a pair of M-level baseband signals, q and i. This q and i signal pair can be represented as $2^L$ unique points (or symbols) in a signal-state-space diagram, or constellation. Six different encoding formats are available, selected via front panel switches, for generating 4-QAM, 8-QAM, 16-QAM, 4-PSK, 8-PSK and 16-PSK signals. A demonstration mode for viewing constellation displays is also provided.

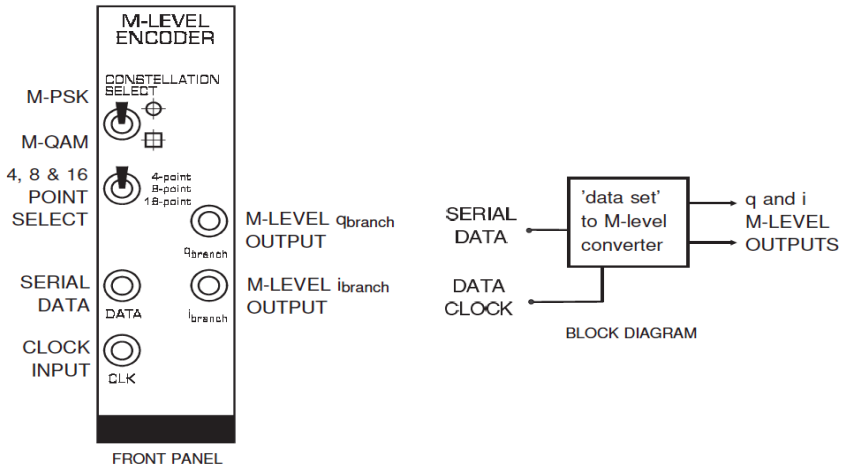



Figure C.12: M-level encoder module

### C.10.1 Use

**Operating modes**

Two operating modes are provided, NORMal and DEMO. The PCB mounted jumper, J3 is used to set the operating mode. NORMal mode provides full functional operation of the module. Both DATA and CLK input signals are required for normal operation. DEMO mode has limited functional application. It is used only for self test and illustration purposes, to allow the quick setting-up of a constellation display on an oscilloscope. Only a clock signal at the CLK input is required: the DATA input is unused.

**Input signals**

Two TTL level input signals are required for normal operation: DATA and CLK. The DATA input signal must be synchronised and in-phase with the CLK signal.

**Constellation select**

Two front panel CONSTELLATION SELECT switches are used to choose the encoding format required. The upper, 2 position switch selects between either a circular (phase) or square (amplitude) array. The lower, 3 position switch selects the number of points in the constellation: 4, 8 or 16.

**Output signals**

Two multi-level analog signals are output, labeled q-branch and i-branch. The number of discrete M-levels and the voltage difference between each level is determined by the front panel CONSTELLATION SELECT switch settings. For each of the six available settings, the peak-to-peak amplitude of the ibranch and qbranch signals will always be ±2.5V.

### C.10.2 Basic Specifications

- DATA Input: serial, TTL-level
- CLK Input: up to 10kHz, TTL-level
- OPERATING MODES: PCB jumper selectable
  - NORM converts sets of input DATA into pairs of multi-level signals
  - DEMO for testing and displaying constellations only
- CONSTELLATION SELECT: front panel switch selectable, offering either circular or rectangular, 4, 8 and 16 point constellations
- i-branch and q-branch Outputs 2, 3, 4 or 8 level, depending upon constellation selected, ±2.5Vpk-pk

## C.11 M-Level Decoder

A pair of baseband, multi-level encoded signals, q and i, originally generated by the M-LEVEL ENCODER module are sampled, decoded into unique

groups of bits length L and output as a continuous serial data stream. The output data is synchronised and in-phase with the bit clock. The input signals, q and i, are sampled at a point determined by the user. Using an oscilloscope, the decision point is displayed as a bright marker on the input waveforms. The sampled and held q and i signals are also output Q and I. Seven different decoding formats are available. The six standard operating mode formats, 4-QAM, 8-QAM, 16-QAM, 4-PSK, 8-PSK and 16-PSK are selected via front panel switches. The seventh decoding format, BPSK, is enabled via a special operating mode of the M-LEVEL DECODER module.

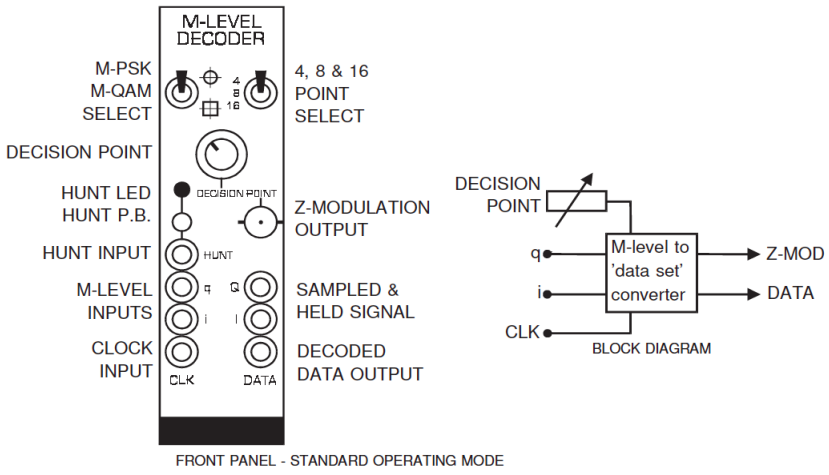



Figure C.13: M-level decoder module

### C.11.1 Use

**Input signals**

Three input signals are required for standard operation: multi-level encoded signals q, i and the data bit clock, CLK.

The peak-to-peak amplitude of the q and i signals must be approximately ±2.5V for optimum decoding performance. Hence, when setting-up experiments always ensure that the amplitudes of the signals being presented to the q and i inputs are correctly adjusted using the gain or amplitude controls of the preceding modules.

The clock input, CLK, accepts a TTL-level signal. It must be synchro-

nised with the incoming q and i M-level signals, though its frequency must be the bit clock rate of the output data. This data bit clock may be regenerated locally, or for maintaining simplicity of the experiment, may be stolen from the M-LEVEL ENCODER modules clock input source.

In order to optimize performance of the user variable decision point, a PCB mounted RANGE jumper must be set to correctly match the input clock frequency. Set the RANGE jumper to LO for clock frequencies up to 4kHz. For clock frequencies above 4kHz, set the RANGE jumper to HI.

**Data output**

A TTL-level data stream of decoded data is output continuously at the DATA output. The data stream is in-phase and synchronised with the bit clock signal at the CLK input. The signals at the Q and I outputs are the actual sampled and held representations of the q and i input signals presented to the internal decoders analog-to-digital converter. Any accumulated DC-offset in either the q or i branch may be viewed at the Q and I outputs, and nulled by adjusting the respective PCB mounted trimmer, RV2 or RV1. Note that the Q and I signals are offset by approximately +2.5V with respect to the q and i input signals.

**Decision point control**

The decision point is the point at which the incoming signals q and i are sampled within each q and i signals symbol. At the sampling instant, the internal decoder makes a decision as to the state, or level, of the sample. Both inputs, q and i, are sampled simultaneously. The thresholds, or decision boundaries, which the internal decoder follows vary, depending upon the constellation selected. The user has control over the sampling instant via the front panel DECISION POINT control knob and the HUNT push button (and HUNT input). The sampling instant is displayed on an oscilloscope as a bright marker, via the Z-MODULATION output, when the q and i input signals are viewed. The sampling instant is moved across each q and i symbol using both the DECISION POINT control knob and the HUNT push button.

### C.11.2 Basic Specifications

- i and q Inputs: 2, 3, 4 or 8 level, depending upon constellation selected, ±2.5Vpk-pk
- CLK Input: up to 10kHz, TTL-level, synchronized with input symbols
- DATA Output: continuous stream of decoded data bits, TTL-level
- I and Q Outputs: sampled and held representation of the input signals, with offset
- OPERATING MODES: selected by method of power-up
  - STANDARD for decoding six front panel selectable constellations
  - BPSK for decoding BPSK signals only
- CONSTELLATION SELECT: front panel switch selectable, offering either circular or rectangular, 4, 8 and 16 point constellations
- Decision boundaries: preset and fixed for each constellation; refer to diagrams in User Manual
- DECISION POINT control: continuous regions, with region selected by HUNT function
- HUNT control: steps DECISION POINT across adjacent regions of the symbol to be sampled
- HUNT Input: TTL-level positive going edge
- HUNT LED: has three functions
  - Slow, regular flashing indicates BPSK operation mode
  - Turns-on to confirm HUNT function has been enabled
  - Indicates invalid data at DATA output
- i and q Input offset control PCB trimmer adjustable, ±0.25V,

## C.12 Quadrature Utilities

Three independent functions are provided: two independent multipliers and an independent adder.

Each MULTIPLIER allows two analog signals $X(t)$ and $Y(t)$ to be multiplier together. The resulting product is scaled by a factor of approximately 1/2.

The ADDER allows two input signals $A(t)$ and $B(t)$ to be added together, in adjustable proportions $G$ and $g$.

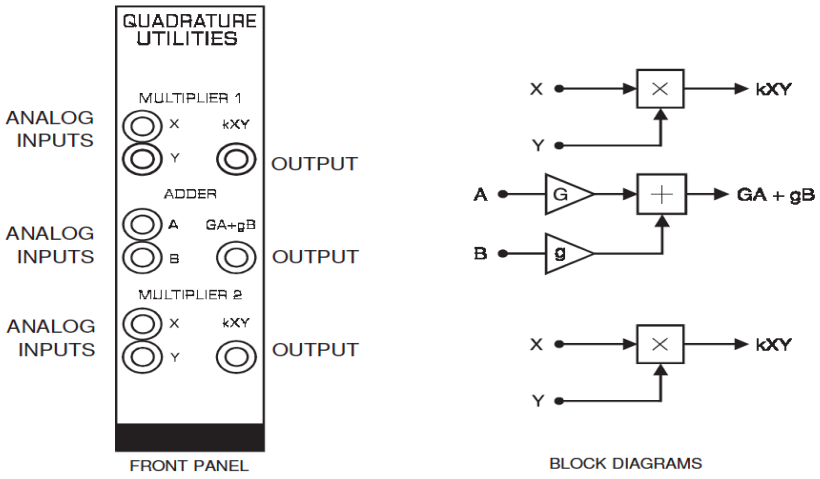



Figure C.14: Quadrature utilities module

### C.12.1 Use

**Multipliers**

Each multiplier has two inputs. The inputs and outputs are DC coupled. The $k$ factor (a scaling parameter associated with four quadrant multipliers) is approximately one half. It is defined with respect to the OUTPUT of the multiplier and may be measured experimentally.

**Adder**

The adder input gains $G$ and $g$ can be adjusted via board-mounted trimmers RV1 and RV3, respectively. Note that these two trimmers have knobs to allow for finger adjustment.

### C.12.2 Basic Specifications

**Multipliers**

- Inputs and Outputs DC coupled
- Bandwidth approx. 1 MHz
- Characteristic $kX(t)Y(t)$ $k$ is approx. 1/2

**Adder**

- Gain range $0 < G, g < 1.5$
- Bandwidth approx. 500 kHz

## C.13 Digital Utilities

Provides six independent digital dividers, a digital inverter and a logical HI output.

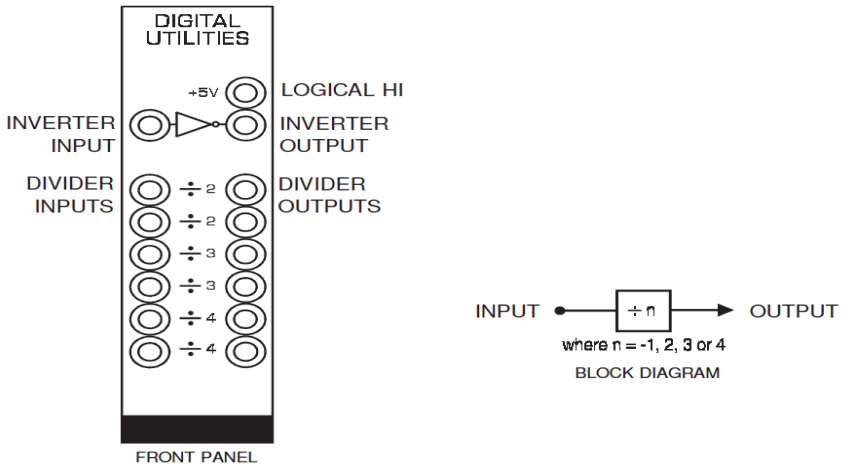



Figure C.15: Digital utilities module

### C.13.1 Use

**Digital inverter and logical HI**

The digital INVERTER only accepts standard TTL-level digital signals. The LOGICAL HI outputs approximately +5 V and is intended only for connection to digital inputs.

**Dividers**

The six, independent digital dividers may be used in any combination to achieve the division ratio required.

### C.13.2 Basic Specifications

- Inputs and Outputs TTL level digital signals
- Input frequency range 0 to 300 kHz

## C.14 Four-Path Time-Invariant Channel Module

It emulates a time-invariant fading channel. This arises in point-to-point communication, such as wireline communications or fixed wireless communications, where the channel response is modeled by a linear time-invariant (LTI) system. It is also a special case in mobile communications where for some period of time (sufficiently long with respect to the symbol interval) the location of mobile terminal as well as positions of obstacles and scatterers is fixed.

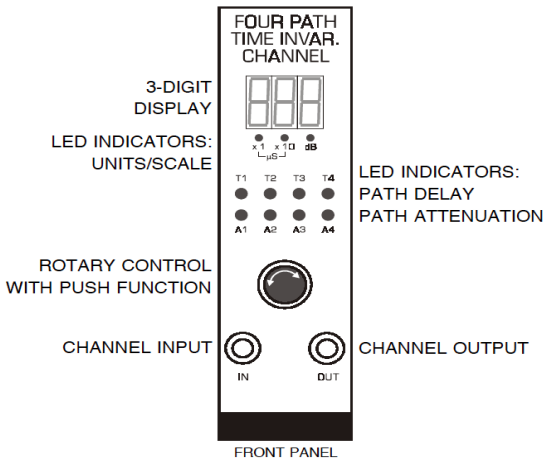



Figure C.16: Four path time-invariant channel module

### C.14.1 Use

**Input**

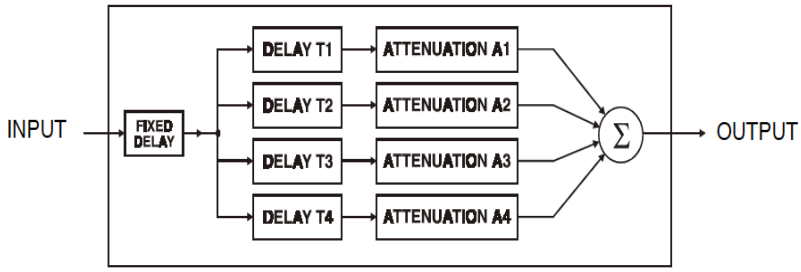



Figure C.17: Four path time-invariant channel block diagram

The channel input will accept analog or digital signals with an amplitude up to ±3 V, including DC component, but due to internal digital signal processing, the input signal should not contain spectral components above 500 kHz, at a level greater than -50 dB.

**Output**

The channel output provides the sum of 4 individually adjustable (delay and attenuation) paths. Depending on the attenuation and delay setup, the channel output amplitude can reach up to ±12 V, therefore be careful to avoid overloading TIMS modules connected to the output.

**Rotary control with push function**

Default setup when the module is first plugged-in: **1st path**: delay: 0 ms, attenuation 0 dB, **2nd path**: delay: 0 ms, attenuation $\infty$ dB (the path is turned off), **3rd path**: delay: 0 ms, attenuation $\infty$ dB (the path is turned off), **4th path**: delay: 0 ms, attenuation $\infty$ dB (the path is turned off)

Pressing the rotary control button cycles between the delay and attenuation settings. The selected setting is indicated by the LED indicator, denoted as: T1, T2, T3, T4 for DELAY or A1, A2, A3, A4 for ATTENUATION. Turning the rotary control knob varies the delay or attenuation is increased.

T1 to T4 indicates the delay for the corresponding path. The delay can be set from 0 $\mu$s to 999 $\mu$s with 1 $\mu$s steps, and from 1 ms to 4.09 ms with 10 $\mu$s steps. The range is switched automatically and is indicated by 2 LEDs. The x1 LED indicates 1 $\mu$s step. The x10 LED indicates 10 $\mu$s steps.

A1 to A4 indicates the attenuation for the corresponding path in dB.

The set attenuation is displayed when the dB LED is lit. The attenuation step is 1 dBV. Maximum achievable attenuation is 16 dBV. A value of 17 on the display indicates that the corresponding path is turned off.

### C.14.2 Basic Specifications

- Input: Analog or digital up a maximum of ±3 V, spectrally limited to -50 dBV with respect to the maximum ±3 V, with amplitude spectrum for frequencies greater than 500 kHz.
- Input level: ±3 V MAX including DC component, with the spectral limitation defined above
- Input filter: passband to 250kHz; $f_{-3\text{dB}}$ is 315kHz
- Output level: up to ±12 V including DC component, with the spectral limitation defined above. The actual output level depends on the delay and attenuation settings for each path.
- Output filter: passband to 250kHz; $f_{-3\text{dB}}$ is 315kHz
- Delay:
  - Range 1: user adjustable from 0 $\mu$s to 999 $\mu$s with 1 $\mu$s step
  - Range 2: user adjustable from 1000 $\mu$s to 4090 $\mu$s with 10 $\mu$s step
  - Range selection: ranges are switched automatically
  - Delay for each path set independently
  - Accuracy: $< 1$ $\mu$s through the whole adjustable range
  - Fixed delay of the module is approximately 15.5 $\mu$s
- adjustable from 0 dBV to 16 dBV with 1 dB steps, attenuation for each path is set independently

## C.15 Speech Module

The speech module allows speech and audio signals to be recorded and replayed. Three independent channels are provided: CHANNEL 1, CHAN-

NEL 2 and LIVE. The module includes an in-built microphone. An external input is also provided for recording externally generated signals. The recorded channels signals are band limited to 300Hz and 3.4kHz. The LIVE channel has user selectable LPF and HPF.

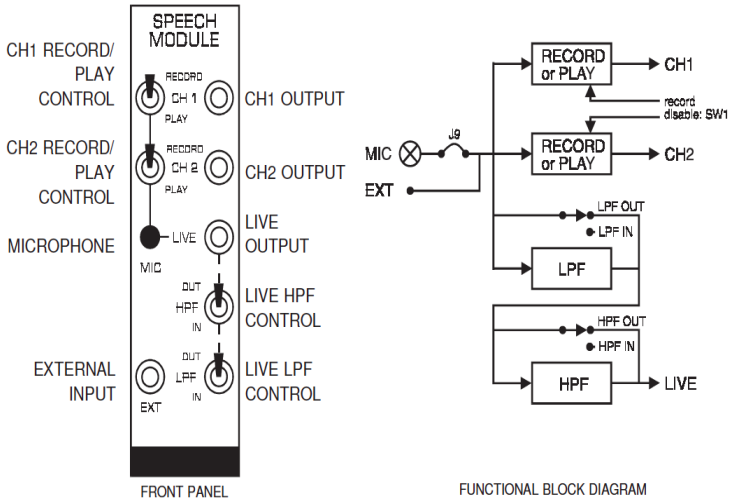



Figure C.18: Speech module

### C.15.1 Use

**Channel 1 and Channel 2**

Channels 1 and 2 will each record up to 32 seconds of speech and sounds from the common microphone input. To record speech or other sounds on either channel, set the front panel switch to RECORD and speak clearly into the microphone. The length of your message may be from a few seconds up to 32 seconds. As soon as you have finished your message, set the switch to the PLAY position. The recorded content will automatically repeat upon switching to PLAY. Note that the length of the recorded message will only be the length of time the switch was in the RECORD position.

The recorded message is stored in non-volatile analog storage arrays and is band limited from 300Hz to 3.4kHz. Each channel has an independent Automatic Gain Control, AGC, that allows for a wide dynamic range of recorded sounds from very quiet to loud voice. To disable the front panel

RECORD switch of either or both channels, SW1 can be set to disable the recording.

**Live Channel**

A third non-recordable channel is also provided where the sound at the microphone is continuously output as an electrical signal. The LIVE channel provides four filtering options with the two front panel selectable filters: a 3.6kHz LPF and a 300Hz HPF.

**Inputs**

Two input sources are provided: the microphone input and the external input. The microphone is a sensitive, electret-type microphone which is fixed in the front panel. This one microphone is common to all three channels. There is also a standard TIMS yellow input socket that allows electrical signals from other signal sources to be recorded and replayed. A pcb mounted jumper, J9, controls the input signal selection: either MIC+EXT, or EXT only. For MIC only operation, leave the EXT input is not connected.

### C.15.2 Basic Specifications

**Channel 1 and Channel 2**

- Bandwidth 300 Hz to 3400 Hz, fixed
- Record length 0 to 32 sec, each channel
- Sampling rate 8 kHz

**Live Channel**

Filters four user selectable settings

- none
- 300 Hz HPF
- 3600 Hz LPF
- 300 Hz HPF and 3600 Hz LPF

**Inputs**

- Microphone in-built electret-type
- External standard TIMS-level, 2V-pk

## C.16 Integrate and Dump

Two independent functional blocks are provided. The first block is a variable digital delay for TTL level clock signals, and may be used for aligning the phase of a bit clock to a data stream.

The second block includes dual channel sampling, integrate and dump and holding functions which can be switched in three combinations: Sample & Hold, Integrate & Dump and Integrate & Hold. A forth, switch selectable function is only available on channel 1, Pulse Width Modulation.

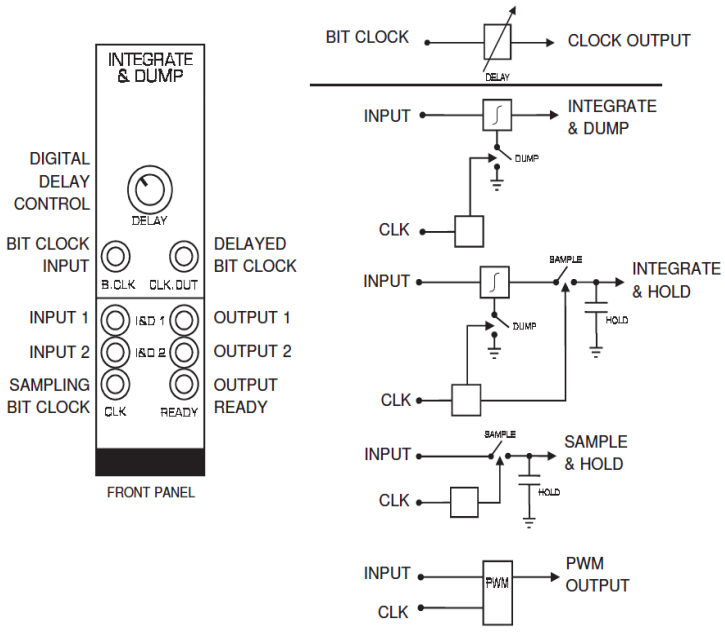



Figure C.19: Integrate and dump

### C.16.1 Use

#### Digital Delay

The variable digital delay accepts a standard TTL level signal at the B.CLK input and also outputs a standard TTL level signal at the CLK.OUT output. Adjusting the DELAY control knob provides a digital phase delay function by varying the time between the positive edge of the signal at the B.CLK input, with respect to the positive edge of the output signal

at CLK.OUT as shown in Fig. C.20. Note that the duty cycle of the input signal is not maintained during the digital delay function. The output signal at CLK.OUT is a fixed pulse of about 10$\mu$s width.

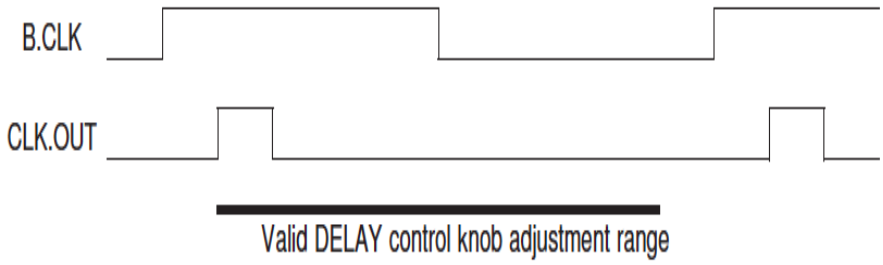



Figure C.20: Digital delay using Integrate and dump module

The DELAY control knob will vary the digital delay time from, approximately, 10 $\mu$s to 1500 $\mu$s, over four user selectable ranges. The adjustment range is selected via the PCB mount switch, SW3 as given in the table below.

| SW3-2 (A) | SW3-1 (B) | DELAY ranges |
|---|---|---|
| OFF | OFF | $10 - 100$ $\mu$s |
| OFF | ON | $60 - 500$ $\mu$s |
| ON | OFF | $100 - 1000$ $\mu$s |
| ON | ON | $150 - 1500$ $\mu$s |

**Sampling and Integrating Functions**

The sampling and integrating block provides two identical channels which operate simultaneously with a common sampling clock. Each channel, I&D1 and I&D2, takes a standard TIMS level analog input. The output signals are analog level.

The two channels require a bit clock for operation which is provided via the CLK input. A standard TTL level signal is required. The READY output pulse is only used when sample & hold or integrate & hold functions are selected. The positive edge of the READY pulse occurs immediately after the signal at the I&D1 or I&D2 outputs has been updated and has settled.

Each channel of the sampling and integrating block includes three circuit functions: a sampler, an integrator and a hold circuit. The user can select the configuration of these circuit functions via two PCB mount, rotary switches: SW1 for channel I&D1, and SW2 for channel I&D2. The available configurations, the corresponding PCB labels and functional descriptions are given in the table below and shown in Fig. C.21. An important note is, that the integrator both integrates and inverts the input signal.

| Label | Function | Description |
|---|---|---|
| S&H1, S&H2 | Sample & Hold | The input signal is sampled, held and output after the occurrence of each positive clk edge. |
| I&H1, I&H2 | Integrate & Hold | The input signal is integrated over the period of the clk signal. At the occurrence of each positive clk edge, the integrator value is transferred to a hold circuit, updating the value at the output. The integrator is then dumped and a new integration period commences. |
| I&D1, I&D2 | Integrate & Dump | The input signal is integrated over the period of the clk signal. During the occurrence of each ready pulse, the integrator is dumped and a new integration period is commenced. The integrator output is available at the output terminal. |

The following table summarizes the components and values associated with the integrator time constant of each channel:

| Channel | Integrator 's resistor | Integrator 's capacitor | Comments |
|---|---|---|---|
| I&D1 | 330 Kohm-R7 | 470 pF- C4 | Fixed RC |
| I&D2 | 330 Kohm-R26 | 470 pF- C34, 470 pF- C44 | When Jumper J1 is open, only C34 is selected and when it is closed it adds C44 to C34. |

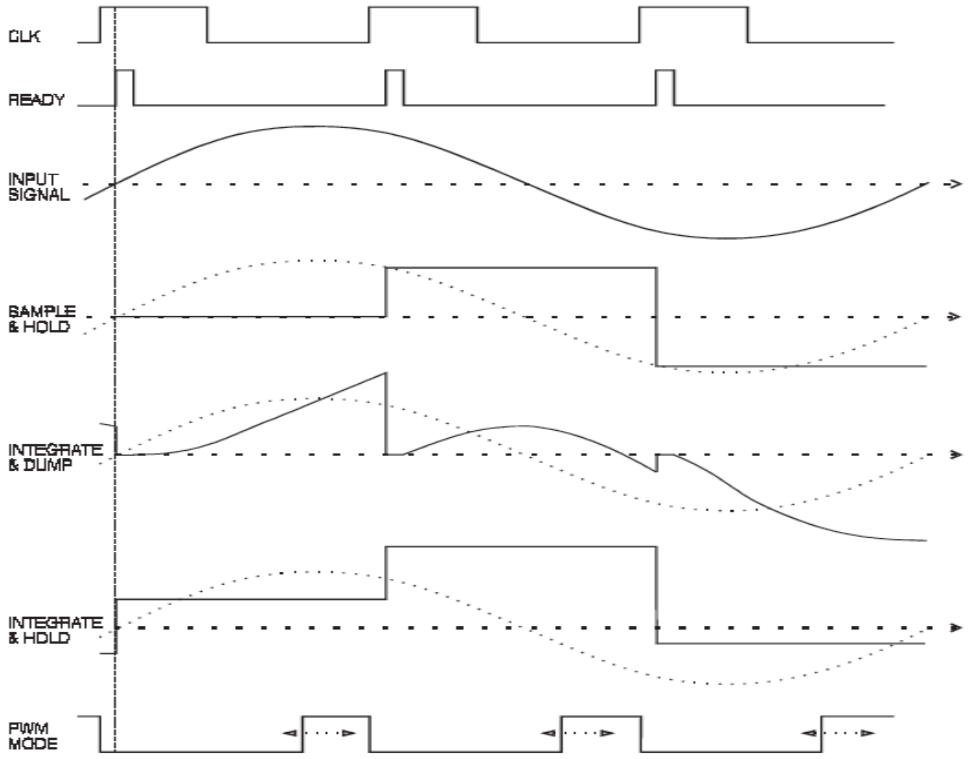



Figure C.21: Integrate and dump function waveforms

### C.16.2 Basic Specifications

#### Digital Delay

- Input & Output TTL level, digital signals

- Clock input $< 15$ kHz
- Variable delay range $10 - 1500\mu$s, in 4 switch selectable ranges

**Integrate & Dump**

- Operating modes: integrate & dump; integrate & hold; sample & hold; PWM.
- 2 channels, simultaneously operating with a common bit clock, with the exception of PWM mode, which is only available on channel 1, I&D1.
- Analog inputs and outputs standard TIMS level
- Clock input: from 500 Hz to 15kHz, standard TTL level
- Integrator: integration commences on the negative edge of the READY signal. When hold is selected, the integrator output is sampled on the positive edge of the clock signal. Dumping commences on the positive edge of the READY pulse. The output of the integrator is inverting.
- Sampler: the sampling of the input signal commences on the positive edge of the clock signal and is completed on the positive edge of the READY pulse.
- Ready TTL level pulse, $< 10$ $\mu$s width. Occurs after the hold circuit output has settled.

## C.17 PCM Encoder

An audio frequency analog-to-digital converter which outputs the digitised data in serial TTL-level PCM format. Both linear and non-linear (logarithmic) digitising schemes are provided. Frame synchronisation is implemented by both separate output synchronisation signal and also an embedded code within the serial data stream.

A variable frequency sinuous-type message is provided, which is always synchronised to the input bit clock.

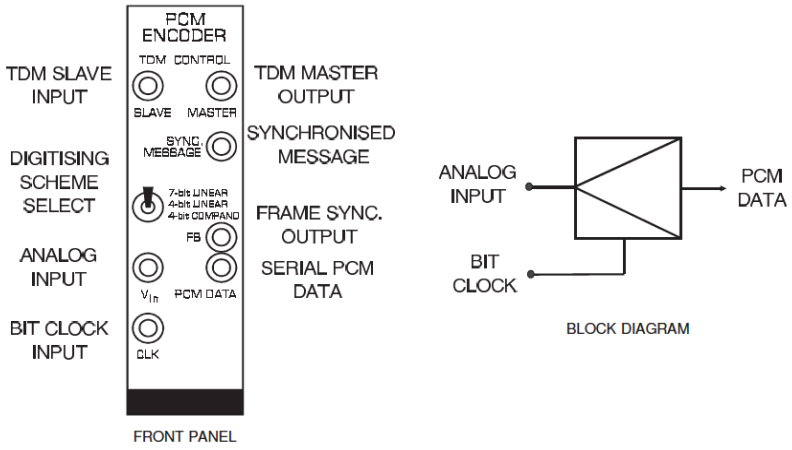



Figure C.22: PCM encoder

### C.17.1 Use

**Input Signals**

Two input signals are required for correct operation: the analog signal to be digitised, $V_{\text{in}}$, and the sampling bit clock, CLK. $V_{\text{in}}$ will accept TIMS-level, bipolar signals ranging from DC up to several kHz. Note that the $V_{\text{in}}$ input is not band limited, so that aliasing may be observed if desired.

The bit clock, CLK, must be a TTL-level signal, such as the TIMS MASTER SIGNALS 8.33 kHz SAMPLING CLOCK output. Note that careful consideration must be given regarding the sampling theorem, when selecting the relative frequencies of both $V_{\text{in}}$ and CLK.

**PCM Data**

The TTL-level digitised data is output serially. TIMS PCM code words are in standard offset binary format, with the first 7 bits allocated for data/coding and the least significant bit allocated for the frame synchronisation code. Three digitising schemes are provided for comparison purposes. Selection is made via front panel switch:

- 7-bit linear,
- 4-bit linear, and
- 4-bit companded, either TIMS A4-Law or TIMS $\mu$4-Law

Note that selection between TIMS A4-Law or TIMS $\mu$4-Law is made via jumper selector on the PCM ENCODER module PCB.

The following timing diagram describes PCM ENCODER operation.

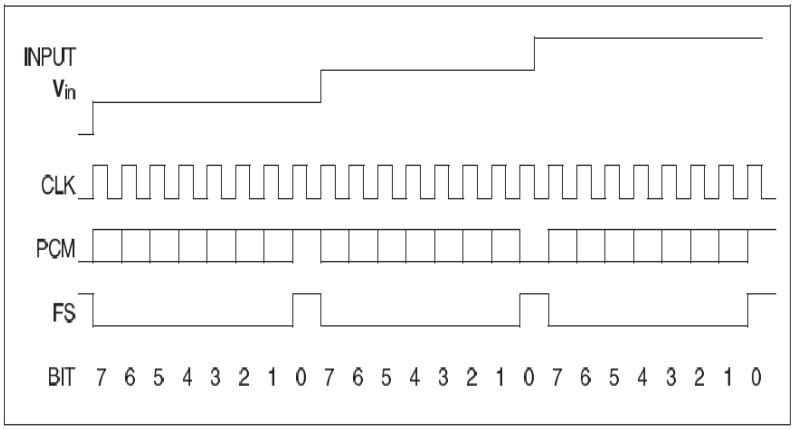



Figure C.23: PCM encoder operation

- INPUT $V_{\text{in}}$ is the input voltage applied at input $V_{\text{in}}$. The waveform is shown as presented to the analog-to-digital converter by the PCM ENCODER module internal sample-and-hold circuit.
- CLK is the applied bit clock at input CLK.
- PCM is the serial data signal at the PCM DATA output. Note that LSB of each frame , bit 0, is shown as carrying the embedded "0 - 1 - 0 - 1" frame synchronisation sequence.
- FS is the frame synchronisation signal as provided at the FS output.
- TIMS PCM CODE WORD RANGES
  - 7-bit LINEAR Frame 0000000X = -2.5V to 1111111X = +2.5V
  - 4-bit LINEAR Frame 0000000X = -2.5V to 0001111X = +2.5V
  - 4-bit COMPANDED Frame 0000000X = -2.5V to 0001111X = +2.5V

**Frame Synchronisation** Two methods are used to indicate frame synchronisation: a separate TTL-level output signal, FS, and an embedded

code within the digitised serial data. The frame synchronisation signal, FS, is normally low and only goes high for one bit period, at the time of the least significant bit of the PCM code word, bit 0.

The frame synchronisation signal is also embedded within the digitised code word, as the least significant bit, bit 0. The code selected is a repeating "0 - 1 - 0 - 1 ...." sequence. This is a unique sequence which corresponds to the Nyquist frequency of the sampled signal and so is otherwise considered a "disallowed" state.

### C.17.2 Basic Specifications

- Input $V_{\text{in}}$ +/-2 Vpk, DC coupled
- Bit Clock Input <10 kHz, TTL-level
- Output Signal serial, TTL-level data stream in offset binary format
- Output Format 8 bits data, including frame synchronisation bit as LSB
- Digitising Formats 7-bits linear; 4-bits linear, and 4-bits companded
- Companding Formats TIMS 4-bit A4-Law, TIMS 4-bit $\mu$4-Law (PCB selectable)
- Sinuous Message Output bipolar, standard TIMS-level and always synchronised to bit clock

## C.18 PCM Decoder

An audio frequency digital-to-analog converter which accepts digital data in serial format, as generated by the PCM ENCODER module. Frame synchronisation may be achieved either from an external synchronisation signal or may be extracted from the embedded frame synchronisation code generated by the PCM ENCODER module. The bit clock provided must be synchronised and in-phase with the incoming digital data.

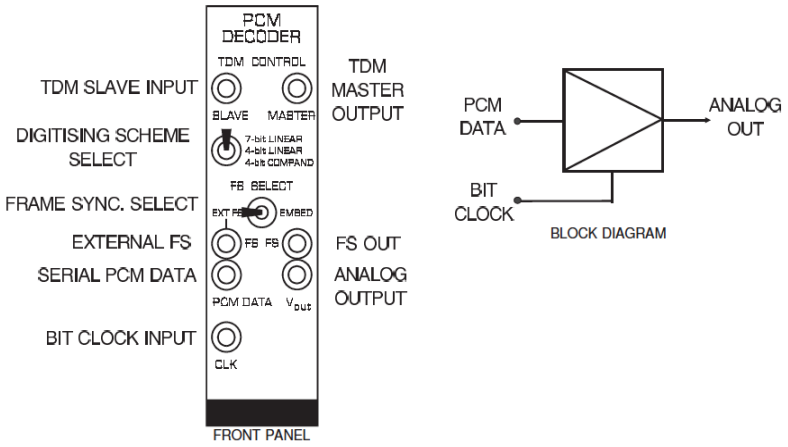



Figure C.24: PCM decoder

### C.18.1 Use

**Input Signals**

Two TTL-level digital signals are required for correct operation: PCM DATA, the serial digital data to be converted to an analog signal and, CLK, a synchronised and in-phase bit clock. Both these signals must be "clean", squared digital signals. Note that the TIMS DECISION MAKER module may be required to "clean-up" digital signals that have undergone any kind of distortion.

**PCM Data**

The format of the serial data expected at the PCM DATA input is exactly as generated by the TIMS PCM ENCODER module: TIMS PCM code words in standard offset binary, with the first 7 bits allocated for data/coding and the least significant bit allocated for the frame synchronisation code. The three digitising schemes provided by the TIMS PCM ENCODER module can be decoded. Selection is made via front panel switch:

- 7-bit linear,
- 4-bit linear, and
- 4-bit companded, either TIMS A4-Law or TIMS $\mu$4-Law

Note that selection between TIMS A4-Law or TIMS $\mu$4-Law is made via jumper selector on the PCM ENCODER module PCB.

### C.18.2 Basic Specifications

- Input PCM DATA serial, TTL level data stream in offset binary format
- Input Format 8 bits, including frame synchronisation bit as LSB
- Digitising Formats 7-bits linear, 4-bits linear, and 4-bits companded
- Companded Formats TIMS 4-bit A4-Law, and TIMS 4-bit $\mu$4-Law (PCB selectable)
- Bit Clock Input <10 kHz, TTL level; positive edges of CLK, PCM DATA coincident
- Output Signal +/-2 Vpk, DC coupled

## C.19 Block Code Encoder

Specifically formatted 8 bit frames of data are input and 8 bit codeword frames are output. Check bits generated by the selected linear code are inserted into predetermined bit positions within the frame. Note that this encoder will maintain a constant frame length of 8 bits by replacing up to 3 redundant data bits with check bits, depending upon the selected linear code. All three digital input signals must always be provided. Code selection is made via a front panel switch.

### C.19.1 Use

**Input Signals**

All three TTL level input signals must be provided for correct operation:

- A TTL level bit CLOCK, synchronised and in-phase with the serial, PCM format, data.
- A TTL level DATA stream, pre-formatted in frames of 8 bits. Correctly pre-formatted data is provided by the PCM ENCODER module, with 4-bit digitising selected.

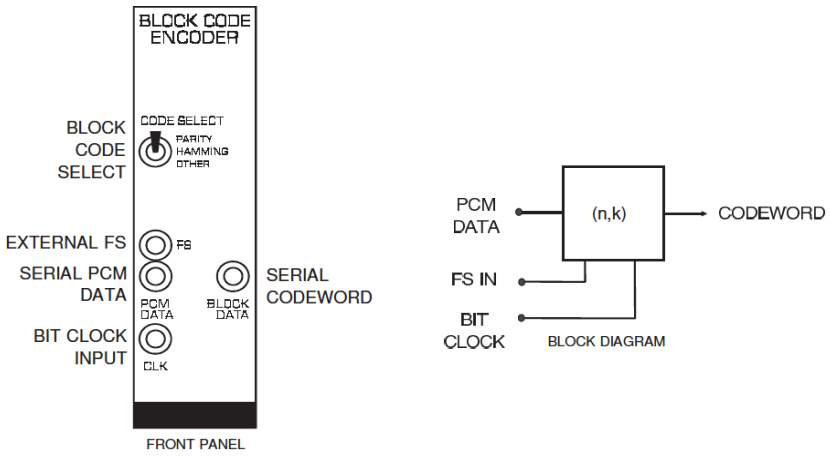



Figure C.25: Block code encoder

- A TTL level FRAME SYNCHRONISATION signal, as provided by the PCM ENCODER module.

**Code Selection**

Three codes are provided for encoding the data. Selection is made via a front panel toggle switch. Refer to the following table for available codes

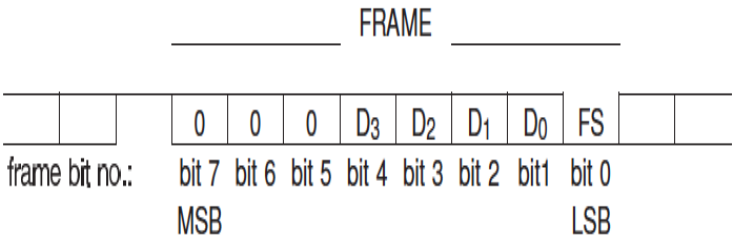
| CODE 1 | CODE 2 | CODE 3 |
|---|---|---|
| Even Parity | Hamming (7,4) | Odd Parity |
| single bit error detect | single bit error correct | single bit error detect |

**PCM and Codeword Bit Formats**

- Input Frame: The required format at the PCM DATA input is either TIMS PCM ENCODER 4-bit scheme, which is shown in Fig. C.26
- Output Frame: The BLOCK CODE ENCODER module outputs codeword bits in the following frame format, which is shown in Fig. C.27

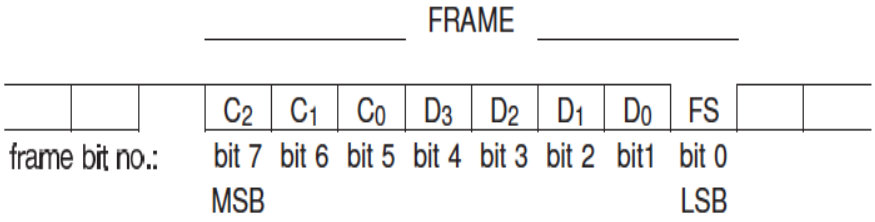
**Frame Synchronisation**

The BLOCK CODE ENCODER module uses the frame synchronisation signals generated by preceding modules, such as the PCM ENCODER




Frame length: 8 bits
Bit 0 (least significant bit): frame synchronisation bit, FS
Bits 1 to 4: message bits, $D_X$; bit 4 is the most significant message data bit
Bits 5 to 7: zero, (redundant data bits)

Figure C.26: Block code input frame




Frame length: 8 bits
Bit 0 (least significant bit): frame synchronisation bit, FS
Bits 1 to 4: message bits, $D_X$; bit 4 is the most significant message data bit
Bits 5 to 7: check bits, $C_x$, used for encoding
- Parity Bit is bit 5; $C_1$ & $C_2$ are set to zero
- Hamming and Cyclic check bits are bits 5, 6 and 7.

Figure C.27: Block code output frame

module. Note that the BLOCK CODE ENCODER module does not generate any separate or independent frame synchronisation signals and does not alter the embedded frame synchronisation bit, bit 0.

### C.19.2 Basic Specifications

- PCM Data Input serial, TTL-level
- PCM Data Input Format 8 bit frame, with 3 most significant bits zero, 4 message bits (bit 4 is most significant data bit) and bit 0 (LSB) is the embedded frame synchronisation bit
- Bit Clock Input typically 2 kHz, ($<$ 8 kHz maximum), TTL-level
- Output Block Data serial, TTL-level
- Output Block Data Format 8 bit frame, with 7 bit codeword plus, LSB as embedded frame synchronisation bit; 1, 2 or 3 most significant bits allocated as check bits, depending upon the selected code
- Frame Synchronisation Input FS synchronisation signal is taken from the preceding module, typically the PCM ENCODER module
- Linear Block Codes Parity - even; Hamming - single error correction; Parity - odd.

## C.20 Block Code Decoder

Frames of digital data which have been encoded using the BLOCK CODE ENCODER module are decoded with error detection and/or correction, depending upon the selected code. Error detection and error correction indication and output signals are provided, as appropriate to the selected code. Frame synchronisation may be achieved either from an external synchronisation signal or may be extracted from the embedded frame synchronisation code within the data received stream. The bit clock provided must be synchronised and in-phase with the incoming digital data. Code selection is made via a front panel switch.

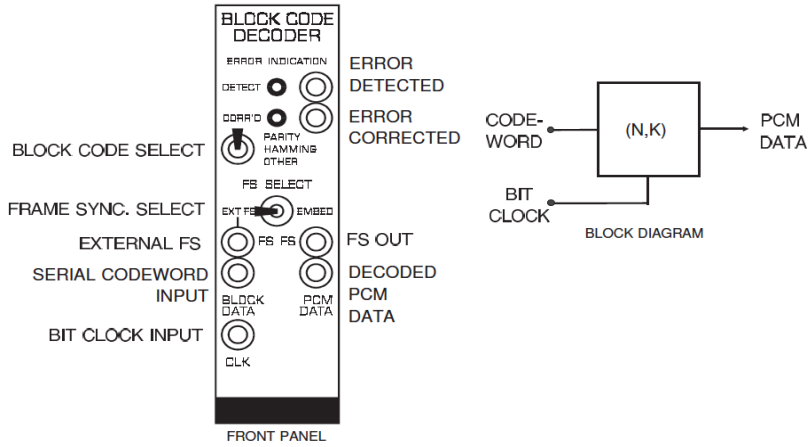



Figure C.28: Block code decoder

### C.20.1 Use

**Input Signals**

All three TTL level input signals must be provided for correct operation:

- A TTL level bit CLOCK, synchronised and in-phase with the serial, PCM format, data.
- A TTL level DATA stream, pre-formatted in frames of 8 bits. Correctly pre-formatted data is provided by the PCM ENCODER module, with 4-bit digitising selected.
- A TTL level FRAME SYNCHRONISATION signal, as provided by the PCM ENCODER module.

**Block Data** The format of the serial data expected at the BLOCK DATA input is exactly as generated by the TIMS BLOCK CODE ENCODER module: 8 bit frame length, with 7 bit codeword and a frame synchronisation bit at bit 0 (LSB).

**Pcm Data Output** The format of the serial data expected at the PCM DATA output is the TIMS standard 4-bit digitised scheme: 8 bit frame length, with 3 most significant bits zero, 4 message bits (bit 4 is the most significant data bit) and bit 0 (LSB) is the embedded frame synchronisation bit.

**Code Selection**

Three codes are provided for encoding the data. Selection is made via a front panel toggle switch. Refer to the following table for available codes

| CODE 1 | CODE 2 | CODE 3 |
|---|---|---|
| Even Parity | Hamming (7,4) | Odd Parity |
| single bit error detect | single bit error correct | single bit error detect |

**ERROR INDICATION**

The BLOCK CODE DECODER module will provide a visual indication of occurrences of error detection and/or error correction. As well, TTL-level signal outputs are provided to allow electronic counting of detection/correction events. The signal at each ERROR INDICATION output is a bit-wide pulse which will be output once per each frame in error. Only one of the two ERROR INDICATION outputs is active for each Block Code selected:

- The ERROR DETECT LED and output is only active for codes that can detect and not correct errors.
- The ERROR CORRECTED LED and output is only active for codes that can detect and correct errors: for these codes, the ERROR DETECT output is not active.

**Frame Synchronisation**

The BLOCK CODE ENCODER module uses the frame synchronisation signals generated by preceding modules, such as the PCM ENCODER module. Note that the BLOCK CODE ENCODER module does not generate any separate or independent frame synchronisation signals and does not alter the embedded frame synchronisation bit, bit 0.

### C.20.2 Basic Specifications

- Block Data Input serial, TTL level
- Block Data Input Format fixed 8 bit frame length, with 7 bit codeword plus, LSB as embedded frame synchronisation bit; 1, 2 or 3 most significant bits allocated as check bits, depending upon selected code

- Bit Clock Input typically 2 kHz, (< 8 kHz maximum), TTL-level
- Output PCM Data serial TTL level
- Output PCM Data Format fixed 8 bit frame, with 3 most significant bits zero, 4 message bits (bit 4 is most significant data bit) and bit 0 (LSB) is embedded frame synchronisation bit
- Frame Synchronisation LINE and EMBEDDED modes
- Linear Block Codes Parity - even; Hamming - single error correction; Parity - odd.
- Error Indication LED and TTL-level pulse output of error detection and error correction events