PREPARATION

1- Sampling with Sample and Hold

Before it is possible to transmit analog information via a digital system the analog signal must first be transformed into a digital format. The *first step* in such a transformation typically involves a sampling process.

Natural sampling

Natural sampling of an analog waveform (message) is examined in the experiment entitled *sampling and TDM*. Natural sampling takes a slice of the waveform, and the top of the slice preserves the shape of the waveform.

Flat top sampling

A very common and easily implemented method of sampling of an analog signal uses the sample-and-hold operation. This produces *flat top* samples. Flat top sampling takes a slice of the waveform, but cuts off the top of the slice horizontally. The top of the slice does *not* preserve the shape of the waveform. Figure 1 below contrasts the two methods.

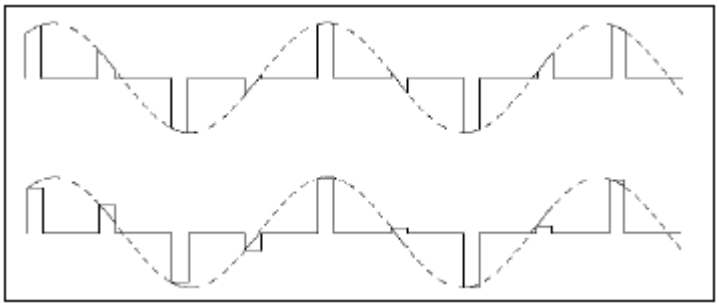

Figure 1: Natural sampling (above) and flat top (below)

Message reconstruction by low pass filtering

In the experiment entitled sampling and TDM a simple analysis showed that there was *no distortion* of the message when reconstruction was implemented by low pass filtering.

It will now be declared as an obvious fact:

if message reconstruction by low pass filtering of natural samples results in no distortion, then there must be distortion when flat top pulses are involved.

Analysis of the distortion for flat top pulses will not be attempted here. Instead some observations will be made, and you can draw your own conclusions.

Sample width

An important observation must be made. The pulse width determines the amount of energy in each pulse and so can determine the amplitude of the reconstructed message. But, in a linear and noise free system, the width of the samples plays no part in determining the amount of distortion of a reconstructed message.

Sample-and-hold sampling

The sample-and-hold operation is simple to implement and is a very commonly used method of sampling in communications systems. In its simplest form the sample is held until the next sample is taken. So it is of maximum width.

This is illustrated in Figure 2 below.

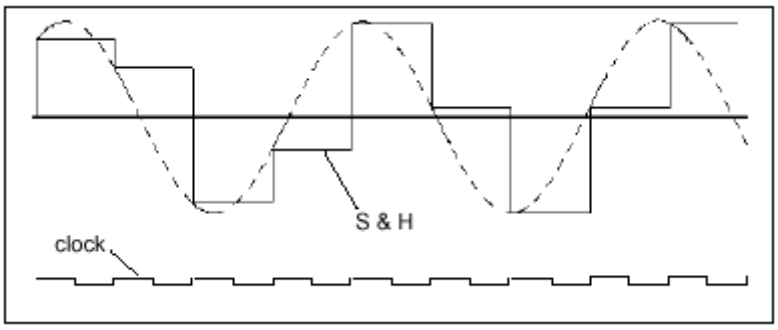

Figure 2: Sampling by sample-and-hold (for full sample width)

In the above example the sampling instant is coincident with the rising edge of the clock signal. In practice there may be a 'processing delay' before the stepped waveform is presented at the output. This is the case in the sub-system being examined in this experiment.

2- PCM Encoding

This is an introductory experiment to pulse code modulation - PCM. The experiment will acquaint you with the PCM ENCODER, which is one of the TIMS Advanced Modules. This module generates a PCM output signal from an analog input message.

In this experiment the module will be used in isolation; that is, it will not be part of a larger system. The formatting of a PCM signal will be examined in the time domain. After in this experiment, we will illustrate the recovery of the analog message from the digital signal.

PCM Encoder

The input to the PCM ENCODER module is an analog message. This must be constrained to a defined bandwidth and amplitude range. The maximum allowable message bandwidth will depend upon the sampling rate to be used. The Nyquist criterion must be observed. The amplitude range must be held within the ± 2.0 volts range of the TIMS ANALOG REFERENCE LEVEL. This is in keeping with the input amplitude limits set for all analog modules.

A step-by-step description of the operation of the module follows:

1. The module is driven by an external TTL clock.
2. The input analog message is *sampled* periodically. The *sample rate* is determined by the external clock.
3. The sampling is a *sample-and-hold* operation. It is internal to the module, and cannot be viewed by the user. What is held is the *amplitude* of the analog message *at the sampling instan*t.
4. Each sample amplitude is compared with a finite set of amplitude levels. These are distributed (uniformly, for *linear* sampling) within the range ± 2.0 volts (the TIMS ANALOG REFERENCE LEVEL). These are the system *quantizing* levels.
5. Each quantizing level is assigned a *numbe*r, starting from zero for the lowest (most negative) level, with the highest number being (L-1), where L is the available number of levels.
6. Each sample is *assigned* a digital (binary) code word representing the number associated with the quantizing level which is closest to the sample amplitude. The number of bits 'n' in the digital code word will depend upon the number of quantizing levels. In fact, $n = \log_2(L)$.
7. The code word is *assembled into a time frame* together with other bits as may be required (described below). In the TIMS PCM ENCODER (and many commercial systems) a single extra

bit is added, in the least significant bit position. This is alternately a *one* or a *zero*. These bits are used by subsequent decoders for frame synchronization.

8. The *frames* are transmitted serially. They are transmitted at the same rate as the samples are taken. The serial bit stream appears at the output of the module.

9. Also available from the module is a synchronizing signal FS ('frame synch'). This signals the *end* of each data frame.

PCM ENCODER module front panel features

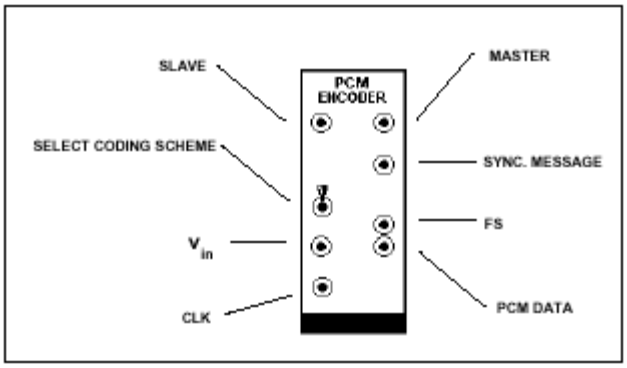

Figure 3: Front panel layout of the PCM ENCODER

The front panel layout of the module is shown in Figure 3. Technical details are described in the TIMS *Advanced Modules User Manual*. Note and understand the purpose of each of the input and output connections, and the three-position toggle switch. Counting from the top, these are:

- ***SLAVE***: not used during this experiment. Do *not* connect anything to this input.

- ***MASTER***: not used during this experiment. Do *not* connect anything to this output.

- ***SYNC. MESSAGE:*** periodic, 'synchronized' message. Either sinusoidal or sinusoidal-like ('sinuous') its frequency being a sub-multiple of the MASTER CLOCK (being any one of four frequencies selected by an on-board switch SW2). A message synchronized to the system clock is convenient for obtaining stable oscilloscope displays. Having a recognizable shape (but being more complex than a simple sine wave) gives a qualitative idea of distortion during the decoding process. See Table A-1 in the Appendix to this experiment for more details.

- ***SELECT CODING SCHEME:*** a three-position toggle switch which selects the 4-bit or 7-bit encoding scheme of the analog samples; or (together with an on-board jumper connection) the companding scheme.

- ***FS***: frame synchronization, a signal which indicates the end of each data frame.

- ***Vin:*** the analog signal to be encoded.

- ***PCM DATA***: the output data stream, the examination of which forms the major part of this experiment.

- ***CLK***: this is a TTL (red) input, and serves as the MASTER CLOCK for the module. Clock rate must be 10 kHz or less. For this experiment you will use the 8.333 kHz TTL signal from the MASTER SIGNALS module.

TIMS PCM time frame

Each binary word is located in a *time fram*e. The time frame contains eight *slots* of equal length, and is eight clock periods long. The slots, from first to last, are numbered 7 through 0. These slots contain the bits of a binary word. The least significant bit (LSB) is contained in slot 0.

The LSB consists of alternating *ones* and *zero*s. These are placed ('embedded') in the frame by the encoder itself and cannot be modified by the user. They are used by subsequent decoders to determine the location of each frame in the data stream, and its length.

The remaining seven slots are available for the bits of the binary code word. Thus the system is capable of a resolution of seven-bits maximum. This resolution, for purposes of experiment, can be reduced to four bits (by front panel switch). The 4-bit mode uses only five of the available eight slots - one for the embedded frame synchronization bits, and the remaining four for the binary code word (in slots 4, 3, 2, and 1).

3- PCM DECODING

The signal that you will generate using the PCM ENCODER module will be decoded in this experiment by *TIMS PCM Decoder Module*.

A clock synchronization signal will be stolen from the encoder.

In the PCM DECODER module there is circuitry which automatically identifies the location of each frame in the serial data stream. To do this it collects groups of eight data bits and looks for the repeating pattern of alternate ones and zeros placed there (embedded) by the PCM ENCODER in the LSB position.

It can be shown that such a pattern cannot occur elsewhere in the data stream provided that the original band limited analog signal is sampled at or below the Nyquist rate.

When the embedded pattern is found an 'end of frame' synchronization signal FS is generated and made available at the front panel.

The search for the frame is continuously updated. Why?

Under noisy conditions (not relevant for this particular experiment) the reliability of the process will depend upon the size of the group of frames to be examined. This can be set by the on-board switch SW3 of the PCM DECODER module. See *Table A-2* in the Appendix to this experiment for details.

Stolen

Frame synchronization can also be achieved, of course, by 'stealing' the synchronization signal, FS, from the PCM ENCODER module. Use of this signal would assume that the clock signal to the PCM DECODER is of the correct phase. This is assured in this experiment, but would need adjustment if the PCM signal is transmitted via a band limited channel (see Tutorial Question 0). Hence, the embedded frame synchronization information.

Companding

You should prepare by reading something about the principles of *companding*. You will already be aware that the PCM ENCODER module can incorporate compression into its encoding scheme. The PCM DECODER module can introduce the complementary expansion. The *existence* of these characteristics will be confirmed, but their *effectiveness* in intelligibility enhancement (when speech is the message) is not examined.

PCM decoding

The PCM DECODER module is driven by an external clock. This clock signal is synchronized to that of the transmitter. For this experiment a 'stolen' clock will be used. The source of frame timing information has been discussed above.

Upon reception, the PCM DECODER:

1. Extracts a frame synchronization signal FS from the data itself (from the embedded alternate ones and zeros in the LSB position), or uses an FS signal stolen from the transmitter (see above).
2. Extracts the binary number, which is the coded (and quantized) amplitude of the sample from which it was derived, from the frame.
3. Identifies the quantization level which this number represents.
4. Generates a voltage proportional to this amplitude level.
5. Presents this voltage to the output V_{out}. The voltage appears at V_{out} for the duration of the frame under examination.

6. Message reconstruction can be achieved, albeit with some distortion, by low pass filtering. A built-in reconstruction filter is provided in the module.

Encoding

At the encoder the sample-and-hold operation (before encoding) is executed periodically. It produces a rectangular pulse form. Each pulse in the waveform is of *exactly* the same amplitude as the message *at the sampling instant*.

But *it is not possible* to recover a *distortionless* message from these samples. They are *flat top*, rather than *natural* samples.

Call this the sampling distortion.

At the encoder the amplitude of this waveform was then *quantized.* It is still a rectangular pulsed waveform, but the amplitude of each pulse will, in general, be in error by a small amount. Call this waveform s(t).

Decoding

The voltage at V_{out} of the decoder is *identical with* s(t) above. The decoder itself has introduced no distortion of the received signal.

But s(t) is already an inexact version of the sample-and-hold operation at the encoder. This will give rise to *quantization distortion* as well as the *sampling distortion* already mentioned.

You should read about these phenomena in a Text book.

TIMS PCM DECODER module

A TIMS PCM DECODER module will be used for decoding.

The front panel of this module is shown in Figure 4. Technical details are described in the TIMS *Advanced Modules User Manual*.

Note and understand the purpose of the input and output connections, and the toggle switches. Counting from the top, these are:

- ***SLAVE***: not used during this experiment. Do *not* connect anything to this input.
- ***MASTER***: not used during this experiment. Do *not* connect anything to this output.
- ***SELECT CODING SCHEME****:* a three position toggle which selects the coding scheme used by the signal to be decoded

- ***FS SELECT:*** a two-position toggle switch which selects the method of obtaining the frame synchronization signal (FS) either external at (EXT.FS), or derived internally from the embedded information in the received PCM itself (EMBED FS).

- ***EXT. FS:*** connect an external frame sync. signal here if this method of frame synchronization is to be used.

- ***EMBED FS:*** if the frame synch. signal is derived internally from the embedded information, it is available for inspection at this output.

- ***PCM DATA:*** the PCM signal to be decoded is connected here.

- ***VOUT:*** the decoded PCM signal.

- ***CLK:*** this is a TTL (red) input, and serves as the MASTER CLOCK for the module. Clock rate must be 10 kHz or less. For this experiment you will use the 8.333 kHz TTL signal from the MASTER SIGNALS module.

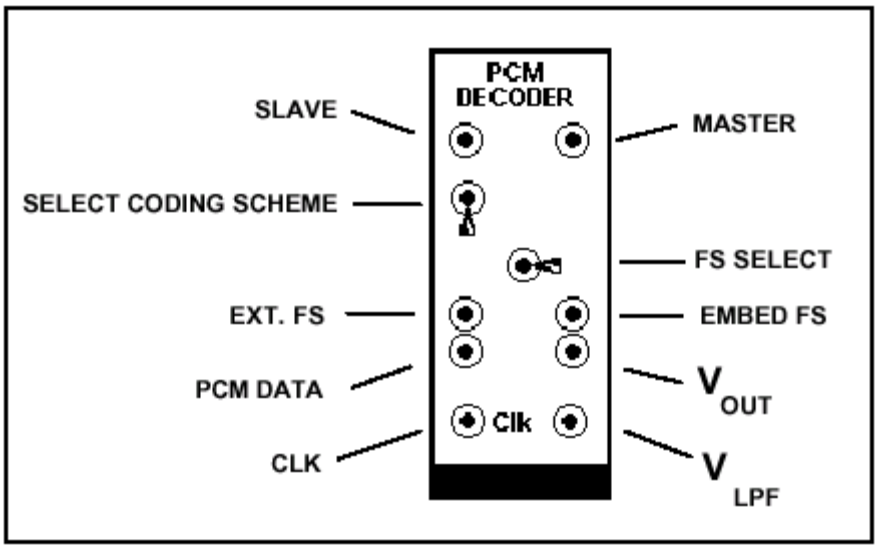

Figure 4: Front panel layout of the PCM DECODER

PRELAB

Sampling with Sample-and-Hold

Sample-and-hold circuits are used in linear systems. They are typically used in analog-to-digital converters to eliminate variations in input signal that can corrupt the conversion process.

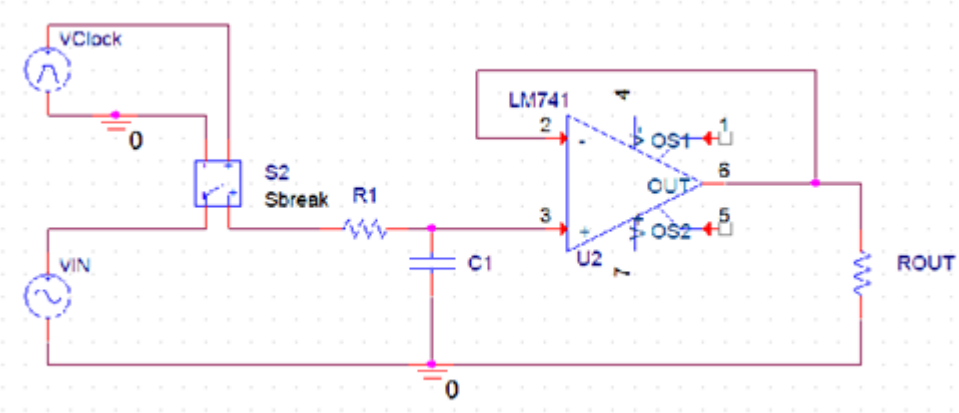

Figure 5: Example of a Sample-and-Hold Circuit

Design a sample-and-hold circuit using PSPICE. Show your analog signal and sampled outputs. Submit your schematics and waveforms.

EXPERIMENT

1- Sampling with Sample and Hold

There is a stand alone SAMPLE-AND-HOLD sub-system in the INTEGRATE & DUMP module. This will be used in the present experiment.

T1 *acquire an INTEGRATE & DUMP module. This is a multi-purpose module. Within it is a sub-system which performs sample-and-hold operations. Before plugging it in, set the on-board switch SW1 to the S&H 1 position ('0'). Analog signals connected to the input socket labeled I&D 1 will now undergo a sample-and-hold (S&H 1) operation, the result appearing at the I&D 1 output socket. Ignore the duplicate S&H 2 option available at the I&D 2 sockets.*

T2 *patch up the module according to Figure 6 below.*

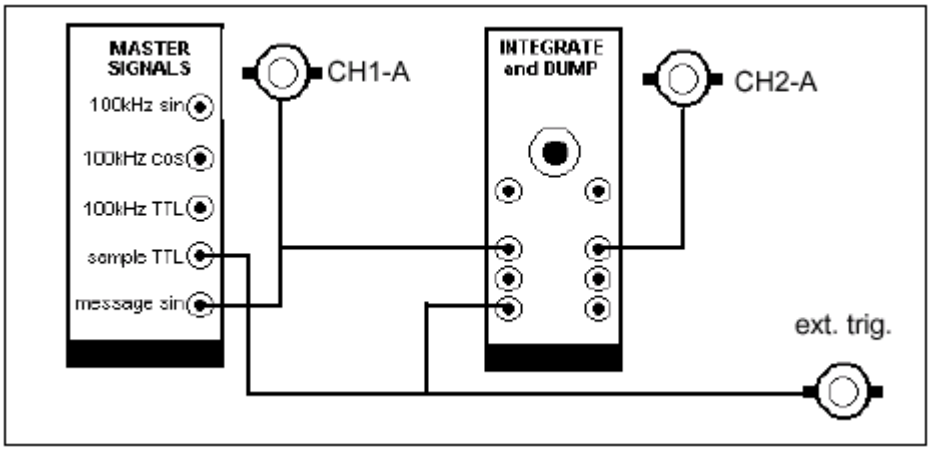

Figure 6: ***TIMS model***

For a stable view of both input and output it is convenient to use a message which is a submultiple of the sample clock frequency. Thus use the 2.03 kHz message (sine wave) from the MASTER SIGNALS module, together with the 8.333 kHz TTL clock.

T3 *select a sweep speed to show two or three periods of the message – say 0.1 ms/cm. Set equal gains of both channels - say 1 volt/cm. With the patching shown in Figure 6 you might expect to obtain oscilloscope displays similar to that of Figure 2. Try it.*

T4 *note the output from the socket labeled READY. Sketch it with respect to the clock and output signal, showing time relationships.*

There is a processing delay within the sample-and-hold sub-system. As a result, the two displays will be shifted relative in time. The ready signal occurs within the time during which the sample is available, and could be used to signal analog-to-digital (A/D) circuitry to start a conversion.

Message reconstruction

Now that you have seen a sample-and-hold operation, you are ready to reconstruct the message from it. This is a low pass filtering operation.

T5 *use a TUNEABLE LPF module to reconstruct the message. Decide on, then set, a 'suitable' bandwidth. Report your findings. Then read on:*

To what passband width did you set the filter?

Remember, you are looking for any possible distortion components introduced by the sample-and-hold operation, and then the reconstruction process.

Since the message is at 2.03 kHz a passband of 3 kHz would be wide enough?

Yes and no!

This would indeed be wide enough to pass the message, but it would not be wide enough to pass any harmonic distortion components.

But the filter passband could not be made wider than half the sampling frequency (else the Nyquist criterion would be violated), and that is not much more than the current message frequency. So something has to be changed.

Is a synchronous message necessary? Not any longer, after having seen the stationary sample-and-hold waveform. So why not use an AUDIO OSCILLATOR, set to its lowest frequency (about 300 Hz) and the 3 kHz LPF within the HEADPHONE AMPLIFIER module. This would give plenty of room for any distortion components to appear at the output. However, unless they are of significant amplitude, they may not be visible on the oscilloscope.

T6 *do as suggested above. Use the oscilloscope to view both the input and output sine waves simultaneously. Synchronize the oscilloscope (externally) to the source of the message. As an engineering estimate, if the distortion is not obvious, then one could say the signal-to-distortion ratio is better than 30 dB (probably better than 40 dB).*

As well as one can judge the two waveforms are 'identical'? Could you estimate the amount of distortion introduced by the reconstruction process?

If there *was* visible distortion then one should check the 3 kHz LPF reconstruction filter - does it introduce its own distortion? Compare the message shape *before* sampling, *but via this filte*r, as well as *after* reconstruction.

Could you attempt to *measure* the mount of distortion?

The unwanted components will probably be hidden in the noise level; meaning the signal-to-distortion ratio is much better than 40 dB.

Two-tone test signal

Testing for distortion with a single sine wave is perhaps not demanding enough. Should you try a two-tone test signal?

With the 3 kHz LPF as the reconstruction filter, and an 8.333 kHz sample rate, there should be no sign of aliasing distortion. To demonstrate aliasing distortion:

T7 *replace the 8.333 kHz sampling signal from the MASTER SIGNALS module with the TTL output from a VCO. Monitor the VCO frequency with the FREQUENCY COUNTER. Starting with the VCO set to its highest frequency on the LO range (about 15 kHz), slowly reduce it, while watching the reconstructed message wave shape. As soon as distortion is evident note the VCO frequency. Knowing the reconstruction filter amplitude characteristic, how does this agree with the Nyquist criterion?*

2- PCM Encoding

The only module required for this part of experiment is a TIMS PCM ENCODER.

It is not necessary, for this experiment, to become involved with *how* the PCM ENCODER module achieves its purpose. What will be discovered is *what* it does under various conditions of operation.

Before plugging the module in:

T8 *select the TIMS companding A_4-law with the on-board COMP jumper (in preparation for a later part of the experiment).*

T9 *locate the on-board switch SW2. Put the LEFT HAND toggle DOWN and the RIGHT HAND toggle UP. This sets the frequency of a message from the module at SYNC. MESSAGE. This message is synchronized to a sub-multiple of the MASTER CLOCK frequency.*

Patching up

To determine some of the properties of the analog to digital conversion process it is best to start with a DC message. This ensures completely stable oscilloscope displays, and enables easy identification of the quantizing levels.

Selecting the 4-bit encoding scheme reduces the number of levels (2^4) to be examined.

T10 *insert the module into the TIMS frame. Switch the front panel toggle switch to 4-BIT LINEAR (i.e. no companding).*

T11 *patch the 8.333 kHz TTL SAMPLE CLOCK from the MASTER SIGNALS module to the CLK input of the PCM ENCODER module.*

T12 *connect the V_{in} input socket to ground of the variable DC module.*

T13 *connect the frame synchronization signal FS to the oscilloscope ext. synch. input.*

T14 *on CH1-A display the frame synchronization signal FS. Adjust the sweep speed to show three frame markers. These mark the* ***end*** *of each frame.*

T15 *on CH2-A display the CLK signal.*

T16 *record the number of clock periods per frame.*

Currently the analog input signal is zero volts (V_{in} is grounded). Before checking with the oscilloscope, consider what the PCM output signal might look like. Make a sketch of this signal, fully annotated. Then:

T17 *on CH2-B display the PCM DATA from the PCM DATA output socket.*

Except for the alternating pattern of '1' and '0' in the frame marker slot, you might have expected nothing else in the frame (all zeros), because the input analog signal is at zero volts. But you do not now the coding scheme.

There *is* an analog *input* signal to the encoder. It is of zero volts. This will have been coded into a 4-bit binary *output* number, which will appear in *each* frame. It need not be '0000'. The *same* number appears in *each* frame because the analog input is *constan*t.

Your display should be similar to that of Figure 7 below, except that this shows five frames (too many frames on the oscilloscope display makes bit identification more difficult).

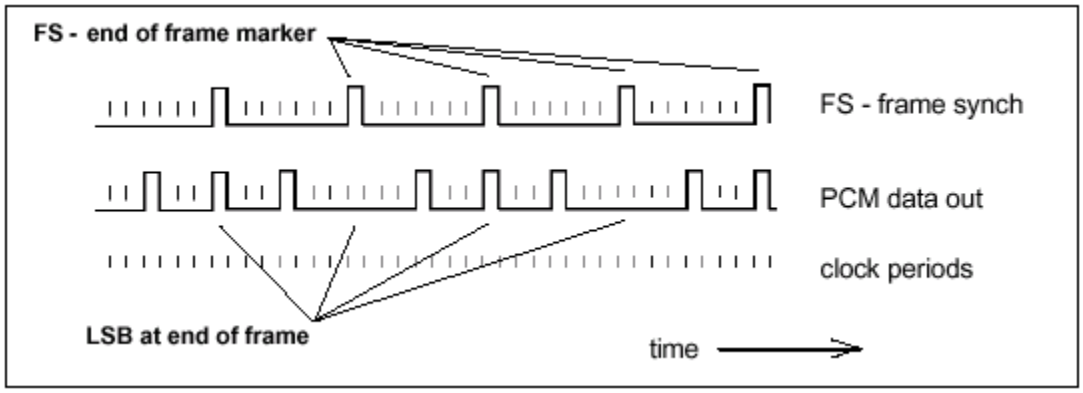

Figure 7: 5 frames of 4-bit PCM output for zero amplitude input

Knowing:

1. The number of slots per frame is 8
2. The location of the least significant bit is coincident with the end of the frame
3. The binary word length is four bits
4. The first three slots are 'empty' (in fact filled with zeros, but these remain unchanged under all conditions of the 4-bit coding scheme)

then:

T18 *identify the binary word in slots 4, 3, 2, and 1.*

Quantizing levels for 4-bit linear encoding

You will now proceed to determine the quantizing/encoding scheme for the 4-bit linear case.

T19 *remove the ground connection, and connect the output of the VARIABLE DC module to* V_{in}*. Sweep the DC voltage slowly backwards and forwards over its complete range, and note how the data pattern changes in discrete jumps.*

T20 *if you have a WIDEBAND TRUE RMS METER module use this to monitor the DC amplitude at* V_{in} *- otherwise use the oscilloscope (CH1-B). Adjust* V_{in} *to its maximum negative value. Record the DC voltage and the pattern of the 4-bit binary number.*

T21 *slowly increase the amplitude of the DC input signal until there is a sudden change to the PCM output signal format. Record the format of the new digital word, and the input amplitude at which the change occurred.*

T22 *continue this process over the full range of the DC supply.*

T23 *draw a diagram showing the quantizing levels and their associated binary numbers.*

4-bit data format

From measurements made so far you should be able to answer the questions:

- what is the sampling rate?
- what is the frame width?
- what is the width of a data bit?
- what is the width of a data word?
- how many quantizing levels are there?
- are the quantizing levels uniformly (linearly) spaced?

7-bit linear encoding

T24 *change to 7-bit linear encoding by use of the front panel toggle switch.*

It would take a long time to repeat all of the above Tasks for the 7-bit encoding scheme. Instead:

T25 *make sufficient measurements so that you can answer all of the above questions in the section titled* ***4-bit data format*** *above. Making one or two assumptions (such as?) you should be able to deduce the coding scheme used.*

Companding

This module is to be used in conjunction with the PCM DECODER in next part. As a pair they have a *companding* option. There is compression in the encoder, and expansion in the decoder. In the encoder this means the quantizing levels are closer together for small input amplitudes - that is, in effect, that the input amplitude peaks are compressed during encoding. At the decoder the 'reverse action' is introduced to restore an approximate linear input/output characteristic.

It can be shown that this sort of characteristic offers certain advantages, especially when the message has a high peak-to-average amplitude characteristic, as does speech, and where the signal-to-noise ratio is not high.

This improvement will not be checked in this part. But the existence of the non-linear quantization in the encoder will be confirmed. In the next part, entitled *PCM decoding*, it will be possible to check the input/output linearity of the modules as a compatible pair.

T26 *change to 4-bit companding by use of the front panel toggle switch.*

T27 *the TIMS A_4 companding law has already been selected (first Task). Make the necessary measurements to determine the nature of the law.*

Periodic messages

Although the experiment is substantially complete, you may have wondered why a periodic message was not chosen at any time. Try it.

T28 *take a periodic message from the SYNC. MESSAGE socket. This was set as the second Task.*

T29 *adjust the oscilloscope to display the message. Record its frequency and shape. Check if these are compatible with the Nyquist criterion; adjust the amplitude if necessary with one of the BUFFER AMPLIFIERS.*

T30 *now look at the PCM DATA output. Synchronize the oscilloscope (as previously) to the frame (FS) signal. Display two or three frames on CH1-A, and the PCM DATA output on CH2-A.*

You will see that the data signal reveals very little. It consists of many overlaid digital words, all different.

One would need more sophisticated equipment than is assumed here (a digital analyzer, a storage oscilloscope, ability to capture a single frame, and so on) to deduce the coding and quantizing scheme from such an input signal.

3- PCM Decoding

Transmitter (encoder)

A suitable source of PCM signal will be generated using a PCM ENCODER module. This module was examined in pervious part. You should set it up before patching up the demodulator.

T31 *before plugging in PCM ENCODER module, set the toggles of the on-board SYNC MESSAGE switch SW2. Set the left hand toggle DOWN, and the right hand toggle UP. This selects a 130 Hz sinusoidal message, which will be used later. Now insert the module into the TIMS system.*

T32 *use the 8.333 kHz TTL signal from the MASTER SIGNALS module for the CLK.*

T33 *select, with the front panel toggle switch, the 4-bit LINEAR coding scheme.*

T34 *synchronize the oscilloscope 'externally' to the frame synchronization signal at FS. Set the sweep speed to 0.5 ms/cm (say). This should show a few frames on the screen.*

T35 *connect CH1-A of the SCOPE SELECTOR to the PCM OUTPUT of the PCM ENCODER.*

T36 *we would like to recognize the PCM DATA out signal. So choose a 'large' negative DC for the message (from the VARIABLE DC module). From previous work we know the corresponding code word is '0000', so only the embedded alternating '0' and '1' bits (for remote FS) in the LSB position should be seen. Confirm this. They should be 1920ms apart. Confirm this both by measurement and calculation!*

T37 *vary the DC output and show the appearance of new patterns on CH1-A. When finished, return the DC to its maximum negative value (control fully anti-clockwise).*

The PCM signal is now ready for transmission. In a later experiment the PCM signal will be sent via a noisy, band limited channel. For the present it will be connected directly to a TIMS PCM DECODER module.

Receiver (decoder)

T38 *use the front panel toggle switch to select the 4-bit LINEAR decoding scheme (to match that of the transmitter)*

T39 *'steal' an 8.333 kHz TTL clock signal from the transmitter and connect it to the CLK input.*

T40 *in the first instance 'steal' the frame synchronization signal FS from the transmitter by connecting it to the frame synchronization input FS of the receiver. At the same time ensure that the FS SELECT toggle switch on the receiver is set to EXT. FS.*

T41 *ensure both channels of the oscilloscope are set to accept DC; set their gains to 1 volt/cm. With their inputs grounded set their traces in the center of their respective halves of the screen. Remove the grounds.*

T42 *connect CH2-A to the sample-and-hold output of the PCM DECODER.*

DC message

You are now ready to check the overall transmission from transmitter input to decoder output. The message is a DC signal.

T43 *connect the PCM DATA output signal from the transmitter to the PCM DATA input of the receiver.*

T44 *slowly vary the DC output from the VARIABLE DC module back and forth over its complete range. Observe the behavior of the two traces. The input to the encoder moves continuously. The output from the decoder moves in discrete steps. These are the 16 amplitude quantizing steps of the PCM ENCODER.*

You are observing the source of quantizing noise. The output can take up only one of 16 predetermined values.

T45 *draw up a table relating input to output voltages.*

You can now see the number of quantizing levels at the transmitter, and their values.

T46 *compare the quantizing levels just measured with those determined in the experiment entitled* ***PCM encoding****.*

T47 *reset the coding scheme on both modules to 7-bit. Sweep the input DC signal over the complete range as before. Notice the 'granularity' in the output is almost un-noticeable compared with the 4-bit case. There are now* 2^7 *rather than* 2^4 *steps over the range.*

Periodic message

It was not possible, when examining the PCM ENCODER in the experiment entitled *PCM encoding*, to see the sample-and-hold waveform within the *encoder*. But you have just been looking at it (assuming perfect decoding) at the output of the *decoder*.

With a periodic message its appearance may be more familiar to you.

T48 *turn back to 4-bit mode. Change to a periodic message by connecting the SYNC MESSAGE of the PCM ENCODER,* ***via*** *a BUFFER AMPLIFIER, to its input* V_{in}*. An amplitude of 2 Vpp is suitable. Slow down the oscilloscope sweep speed to 1 ms/cm. Observe and record the signal at CH2-A.*

When you agree that what you see is what you expected to see, prepare to make a change and predict the outcome.

Currently the encoding scheme is generating a 4-bit digital word for each sample.

What would be the change to the waveform, now displaying on CH2-A, if, at the encoder, the coding scheme was changed from 4-bit to 7-bit?

Sketch your answer to this question - show the waveform *before* and then *after* the change.

T49 *change the coding scheme from 4-bit to 7-bit. That is, change the front panel toggle switch of* ***both*** *the PCM ENCODER* ***and*** *the PCM DECODER from 4-bit to 7-bit. Observe, record, and explain the change to the waveform on CH2-A.*

When satisfied, proceed.

Message reconstruction

You can see, qualitatively, that the output is related to the input. The message could probably be recovered from this waveform. But it would be difficult to predict with what accuracy.

Low pass filtering of the waveform at the output of the decoder will reconstruct the message, although theory shows that it will not be perfect. It will improve with the number of quantizing levels.

What amplitude characteristic is required for the reconstruction filter?

If any distortion components are present they would most likely include harmonics of the message. If these are to be measurable (visible on the oscilloscope, in the present case), then they must not be removed by the filter and so give a false indication of performance.

So we could look for harmonics in the output of the filter. But we do not have conveniently available a spectrum analyzer.

An alternative is to use a two-tone test message. Changes to its shape (especially its envelope) are an indication of distortion, and are more easily observed (with an oscilloscope) than when a pure sine wave is used. It will be difficult to make one of these for this experiment, because our messages have been restricted to rather low frequencies, which are outside the range of most TIMS modules.

But there is provided in the PCM ENCODER a message with a shape slightly more complex than a sine wave. It can be selected with the switch SW2 on the encoder circuit board. Set the left hand toggle UP, and the right hand toggle DOWN. See the Appendix to this experiment for more details.

A message reconstruction LPF is installed in the PCM DECODER module (version 2 and above). If you do not have such a module then bypass the next two Tasks.

T50 *change to the complex message from the PCM ENCODER as described above.*

T51 *connect PCM DECODER output to tunable LPF. Set tunable LPF to low cutoff frequency and observe the reconstructed message. Make comparisons between the 4-bit linear and the 7-bit linear coding schemes. Try different message amplitudes into the PCM ENCODER. Can you observe any distortion? Record your observations.*

If you think the LPF itself might have introduced some distortion, you could check by connecting the complex message to its input direct, and observing the output.

Companding

It is now time to verify the companding algorithm installed in the encoder.

T52 *use the front panel toggle switches (on both modules) to select 4-bit companding. Use both 'low' and 'high' level messages into the PCM ENCODER. Check the quantizing characteristic. Record your observations and comment upon them.*

Frame synchronization

In all of the above work the frame synchronization signal FS has been stolen from the encoder (as has been the clock signal). This was not necessary.

The PCM ENCODER has circuitry for doing this automatically. It looks for the alternating '0' and '1' pattern embedded as the LSB of each frame. It is enabled by use of the FS SELECT front panel toggle switch. Currently this is set to EXT FS.

T53 *change the FS SELECT switch on the front panel of the PCM DECODER module from EXT FS to EMBED. Notice that frame synchronization is re-established after a 'short time'. Could you put an upper limit on this time?*

TUTORIAL QUESTIONS

Q1 *assuming a sine wave is accompanied by a small third harmonic component, how large would this have to be before its presence could be detected using only an oscilloscope? This question would not please the purists, because it raises more questions than it asks. But attempt an answer. You could even set up the signal using TIMS and* ***demonstrate*** *your reply.*

Q2 *define the 'slot bandwidth' of a low pass filter. Redefine the Nyquist criterion in terms of practical filter characteristics.*

Q3 *sample-and-hold (flat-top sampling) can be shown to introduce distortion of the message if it is reconstructed by using a low pass filter alone. From your general reading, or otherwise, is it possible to eliminate this distortion by further message processing?* ***hint****: key words are aperture effect, sinx/x correction.*

Q4 *from your knowledge of the PCM ENCODER module, obtained during preparation for the experiment, calculate the sampling rate of the analog input signal. Show that it is the same for both the 4-bit and the 7-bit coding schemes. What can you say about the bandwidth of an input analog signal to be encoded?*

Q5 *define what is meant by the data 'frame' in this experiment. Draw a diagram showing the composition of a frame for:*

a) The 4-bit coding scheme

b) The 7-bit coding

scheme

Q6 *it is possible to transmit each frame at a much slower rate than it was produced (and, of course, recover the original message as well). Explain how this might be done. When might this be an advantage?*

Q7 *explain why a DC message gives a stable oscilloscope display of the PCM DATA output. Why is the display 'unstable' when a sine wave (for example) is the message?*

Q8 *two PCM signals can be combined to produce a time division multiplexed (PCM TDM) signal. With the measurements so far performed this does not seem (and indeed, is not) possible with two PCM ENCODER modules! Why is this so? Suggest what changes could be made to the module to implement PCM TDM.*

Q9 *in the present experiment a 'stolen' clock signal was used. Why would transmission of the PCM signal via a band limited channel necessitate phase adjustment of this stolen clock signal to the PCM DECODER?*

Q10 *sketch the waveforms at the output* V_{out} *from the decoder, for the 4-bit and the 7-bit linear encoding scheme (and a 'large amplitude' sinusoidal, synchronous message at the encoder). A 'sketch' might show these as being the same, but a more accurate drawing would show more clearly the difference. Explain.*

Q11 *two sources of distortion of the reconstructed message have been identified; they were called sampling distortion and quantizing distortion.*

a) Assuming a sample-and-hold type sampler, what can be done about minimizing sampling distortion?

b) What can be done about minimizing quantizing distortion?

Q12 *quantizing distortion decreases with the number of quantizing levels available. There is usually a price to be paid for such an option. What would this be? Was that apparent in the present experiment? Explain.*

APPENDIX

For a MASTER CLOCK of 8.333 kHz, Table A-1 below gives the frequencies of the synchronized message at the SYNC. MESSAGE output for the setting of the on-board switch SW2. For other clock frequencies the message frequency can be calculated by using the 'divide by' entry in the Table.

These messages are periodic, but not necessarily sinusoidal in shape. The term 'sinuous' means sine-like.

LH toggle	*RH toggle*	*divide clock by*	*freq with 8.333kHz clock*	*approx. ampl. and waveform*
UP	UP	32	260.4 Hz	0.2 V_{pp} sine
DOWN	UP	64	130.2 Hz	2.0 V_{pp} sine
UP	DOWN	128	65.1 Hz	4.0 V_{pp} sinuous
DOWN	DOWN	256	32.6 Hz	4.0 V_{pp} sinuous

Table A-1

The PCM DECODER module has built in circuitry for locating the position of each frame in the serial data stream. The circuitry looks for the embedded and alternating '0' and '1' in the LSB position of each frame. The search is made by examining a section of data whose length is a multiple of eight bits. The length of this section can be changed by the on-board switch SW3. Under noisy conditions it is advantageous to use longer lengths.

The switch settings are listed in Table A-2 below.

left toggle	right toggle	groups of eight bits
UP	UP	4
UP	DOWN	8
DOWN	UP	16
DOWN	DOWN	32

Table A-2: Synchronization search length options