CCITT SGXV Document # 91
Specialists Group on Coding for Visual Telephony

SOURCE : THE NETHERLANDS

TITLE : EXAMPLE OF SIMULATIONS OF THE DUTCH HYBRID CODING CONFIGURATION AT 384 KBIT/S AND 64 KBIT/S

In this document details are presented concerning the simulations at 60 kbit/s and 300 kbit/s using a DCT based hybrid coding scheme with motion compensation.

The following source material and parameters is used:

1 COST sequence

luminance : 360 x 288
chrominance : 180 x 144
frame rate : 10 Hz (intermediate format subsampled 3:1)
bitrate : 300 kbit/s

2 Miss America

luminance : 360 x 288
chrominance : 180 x 144
frame rate : 10 Hz (intermediate format subsampled 3:1)
bitrate : 300 kbit/s

3. Checked Jacket

luminance : 360 x 288
chrominance : 180 x 144
frame rate : 10 Hz (intermediate format subsampled 3:1)
bitrate : 300 kbit/s

4. Graphics

luminance : 360 x 288
chrominance : 180 x 144
frame rate : 10 Hz (intermediate format subsampled 3:1)
bitrate : 300 kbit/s

5 Miss America

luminance	:	360 x 288
chrominance	:	180 x 144
frame rate	:	7.5 Hz (intermediate format subsampled 4:1)
bitrate	:	60 kbit/s

GENERAL OUTLINE OF THE CODING ALGORITHM.

The image is processed in blocks of 8 x 8 pixels. f(q,t) denotes a sub-block q at time t. A transformed sub-block is described by F(q,t) at time t. In order to obtain a minimum prediction error we have to minimize e(q,t);

$$e(q,t) = f(q,t) - \hat{f}(q,t)$$

$$\hat{f}(q,t) = f(r,t-1,D)$$

where :

$e(q,t)$	=	prediction error actual block
$f(q,t)$	=	actual block
$\hat{f}(q,t)$	=	displaced block
D	=	the displacement vector
r	=	sub-block to be estimated.
$f(t-1)$	=	previous frame

The frame memory contains the previous frame f(t-1). The prediction error e(q,t) is transformed (E(q,t)), quantized ($E^{*}(q,t)$) , variable length coded $M(E^{*}(q,t))$ and transmitted.

$$e(q,t) \rightarrow E(q,t) \rightarrow E^{*}(q,t) \rightarrow M(E^{*}(q,t))$$

A further reduction of the prediction error can be achieved by the use of motion compensation. The displacement estimation is based on a block match procedure.

The coding strategy is based on :

- a discrete cosine transform (blocksize 8 x 8)
- a non-uniform quantizer with a variable stepsize; which is adaptive to the buffer status.
- a variable length code which is a simple tree at the moment.

In this coding scheme only 30.000 bits per frame are used including all the overhead. The buffer status gives an indication for the adaptation of the variable bitrate. E.g. when the buffer is full, coarse quantizing is the result. Apriori knowlegde is used in order to equalize the buffer content. This will be explained in the paragraph buffer control.

The configuration of the coder and decoder are shown in figure 1 and figure 2.

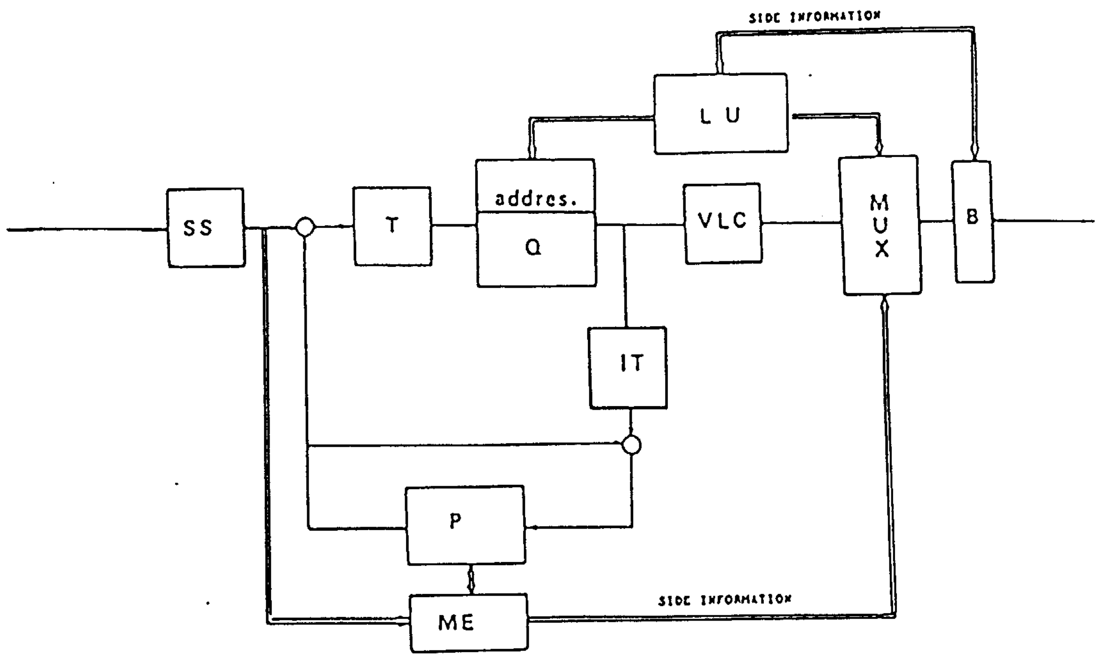


figure 1 : hybrid coding scheme

SS : Subsampling
T : Transform
Q : Quantiser
IT : Inverse Transform
MUX : Multiplexer

P : Predictor
ME : Motion Estimator
VLC : Variable length Coder
B : Buffer
LU : Logical unit

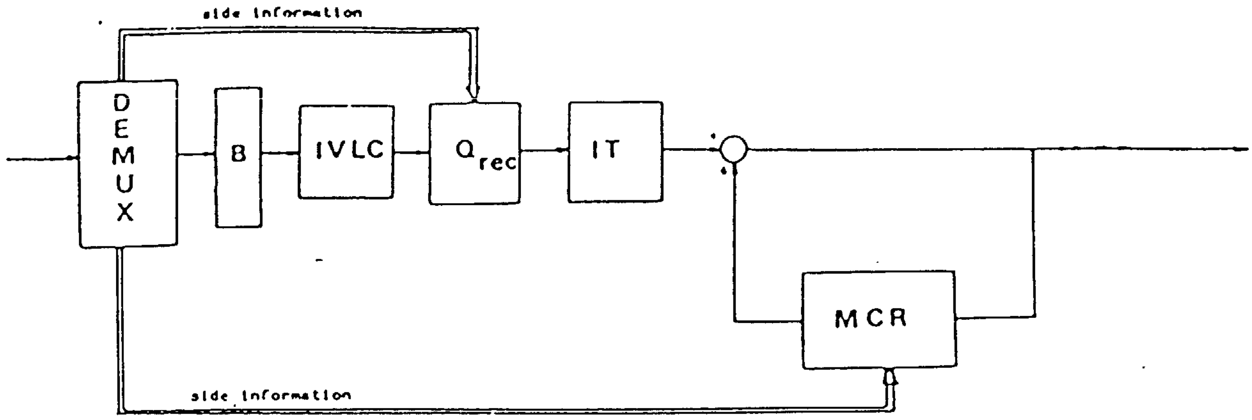


figure 2 : hybrid decoding scheme

IT : Inverse Transform
IVLC : Inverse Variable Length Coder
Qrec : Recontruction quantiser levels
MCR : Motion Compensated Reconstruction

The total number of 8 x 8 blocks for the intermediate format is 1620. Using a 40 percent criterium for the moving blocks, and a frame frequency of 10 Hz, only 44 bits/block are allowed for one block.

USED TRANSFORM

To code the prediction error e(q,t) the 2-dimensional orthogonal. discrete cosine transform is used with a blocksize of 8 x 8. The blocksize for the luminance and the chrominance is the same.

The 2-D DCT is defined as :

$$F(u,v) = \frac{2}{\sqrt{MN}} \sum_{j=1}^{M} \sum_{k=1}^{N} C(u)\ C(v)\ f(j,k)\ \left(\cos\left(\frac{\pi u(2j+1)}{2M}\right)\cos\left(\frac{\pi u(2k+1)}{2N}\right)\right.$$

with $u,v = 0,1,2 \ldots\ldots\ldots N-1$

$$f(j,k) = \frac{2}{\sqrt{MN}} \sum_{u=1}^{M} \sum_{v=1}^{N} C(u)\ C(v)\ F(u,v)\ \left(\cos\left(\frac{\pi u(2j+1)}{2M}\right)\cos\left(\frac{\pi u(2k+1)}{2N}\right)\right.$$

where j,k = spatial coordinates in the time domain
u,v = sequency coordinates in the transform domain

$C(u),C(v) = 1/\sqrt{2}$ for $u,v = 0$
~~0~~ 1 otherwise

N,M = block size

because of the separability :

$$T f = T_c\ f\ T_r^t = F$$

where: T_c = transformation matrix on the columns
T_r = transformation matrix on the rows
f = original sub-block

In the case of an ortho~~gonal~~normal transform:

$$T_r\ I\ T_c^t = I$$

with a = scaling factor
I = unity matrix

The dynamic range of the transform coefficients E(q,t) for a blocksize 8 x 8 pixels becomes, +/- 2048.

MOTION ESTIMATION

The decoder operates independantly of the method used to calculate the displacement vectors. For the configuration the displacement vector, is transmitted as side information. The aim of the displacement estimator is to find the displaced sub-block f(q,t-1,D) which minimizes the prediction error according to a optimization criterium. Common optimalization criterions are the sum of the square or absolute prediction errors. In the simulations the sum of the absolute prediction errors is used.
In case of a non-full search algorithm, the true (global) minimum is not always found. The global minimum will certainly be found by a full search (brute force) along all possible displacement vectors within the search window.
The histogram of the displacement vectors for the different sequences show a peaked function with the zero vector as its maximum (see for example figure 3.)

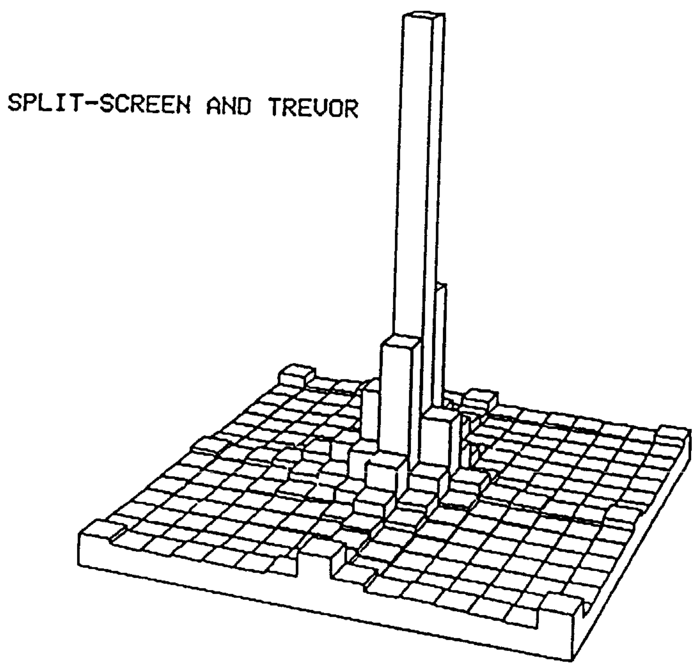


figure 3 Histogram displacement vectors

Variable length coding of these vectors is used to decrease the amount of side information.
The assignment is given in table 1.

displacement	code
-7	1101
-6	11111111
-5	11111101
-4	1111101
-3	111101
-2	11101
-1	101
0	0
-1	100
-2	11100
-3	111100
-4	1111100
-5	11111100
-6	11111110
-7	1100

Table 1

The global minimum is in the case of the sub-optimal procedures not always found. The aim of the block match procedures is to minimize the prediction error only and will not yield in a true vector. The number of calculations for the brute force is not acceptable; it introduces a heavy computational load . As a result of the sub-optimal search algorithms the number of calculations will decrease, the prediction error however increases which cost bits.

In the described hybrid coder a truncated brute force (TBF) is implemented with a self adjusting truncation point in order to decrease the number of calculations. All the positions within the search window still have to be checked but the speed for each block depends on a self adjusting threshold.

The initial threshold value is based on the frame difference (FD).

The procedure for the TBF is given as :

$$T_{part} = \sum_{i=1}^{8} \sum_{j=1}^{8} | C_{i,j}(t) - C_{i+\Delta x, j+\Delta y}(t-1) |$$

where:

T_{part} = partial sum for the comparison with the old threshold.

$C_{i,j}(t)$ = sub-block in actual frame .

$C_{i+\Delta x, j+\Delta y}(t-1)$ = displaced sub-block within search window at position $\Delta x, \Delta y$

D = displacement vector (Δx , Δy) belonging to global minimum

when

$$D = \mathrm{Min}_{\Delta x, \Delta y \in W} \{T\}$$

with :

W = search window in previous frame

$\Delta x, \Delta y$ = position within the search area

Two different situations are to be distinguished:

$T_{part} >= T_{\Delta x, \Delta y}$ stop calculation ,shift to next position

$T_{part} < T_{\Delta x, \Delta y}$ Threshold value $T_{\Delta x, \Delta y} = T_{part}$

In the last case the sum of N x N absolute values is calulated.

SUMMARY OF TBF

The procedure to follow is:

step 1 ---> T = FD i.e. $|f(q,t)-f(q,t-1)|$

step 2
- start to calculate T_{part} for position A on the inner circle and check if $T_{part} > T$
- if this true stop the calculation of the partial sum an proceed to next position on inner circle. Store the displacement and $T = T_{part}$

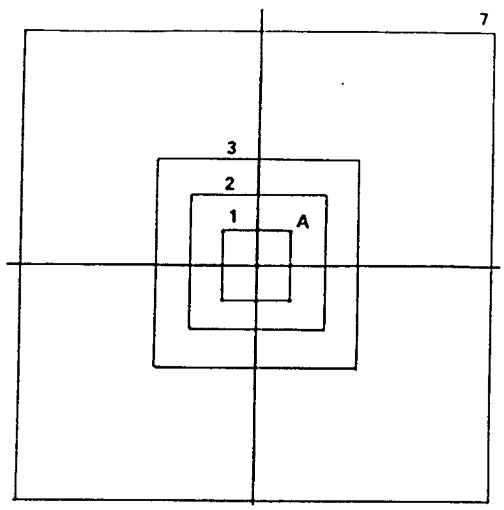


figure 4 search pattern Truncated Brute Force

The blocksize is 8 x 8 , the maximum displacement range is limited to +/- 7, the displacement resolution is 1 pel. Due to the histogram of the displacement vectors the given search pattern in figure 4 will yield in a quicker procedure to obtain the the global minimum. In the case of a normal scanning i.e. if the motion vectors are uniformly distributed, the number of calulations is:

blocksize	8 x 8
search window	22 x 22
maximum displacement	+/- 7
number of positions	225
absolute value criterium	64
number of operations per block	14400

The gain obtained by using the truncation depends on the distribution of the displacement vectors.(approximately up to a factor of 10) Experiments have shown that the displacement vectors for videoconferencing scenes are concentrated arround the zero vector. Using this knowlegde the self-adjusting threshold will converge quicker into the direction of the minimum distortion. (see figure 4.)

THE PREDICTION

The predictor has two main tasks;

- Decision to use the displaced sub-block (transmit teh displacement vector) or to transmit only the zero-vector.
- mode selection intra/inter frame coding.

To be sure that in anycase the motion compensation will give a higher gain (less bits) the following relation is used:

$$\frac{FD}{DFD}\,(FD - DFD) > T_{gain}$$

with FD = frame diffrence
DFD = displaced frame difference
T_{gain} = threshold

If the expression is valid (i.e. true) the motion vector is used otherwise the zero vector. The decision to use the zero vector or the motion vector is made before the transformation and quantisation. This method gives the answer to the question what to do with the overhead using a blocksize of 8 x 8.

A scene change is detected with a threshold on the the frame difference. If the scene change is detected, the entire frame is "intra" frame coded with twice the number of bits, i.e. an extra frame drop. In all other cases interframe coding is used.

QUANTIZER

The transformed block E(q,t) is fed into a non-uniform quantizer with the following characteristic:

$$q_n = T + (n-1)\,g + \sum_{m=3}^{n-1} m \cdot d$$

with q_n = decision level

g = stepsize

T = threshold (1.5 x g)

d = adaptive to block difference (in the simulation d = g)

The representation levels become ;

$$q_{rep} = \frac{q_n + q_{n-1}}{2}$$

The stepsize is controlled by the logical unit which monitors the buffer status.
The implemented quantizer characteristic is show in figure 5.

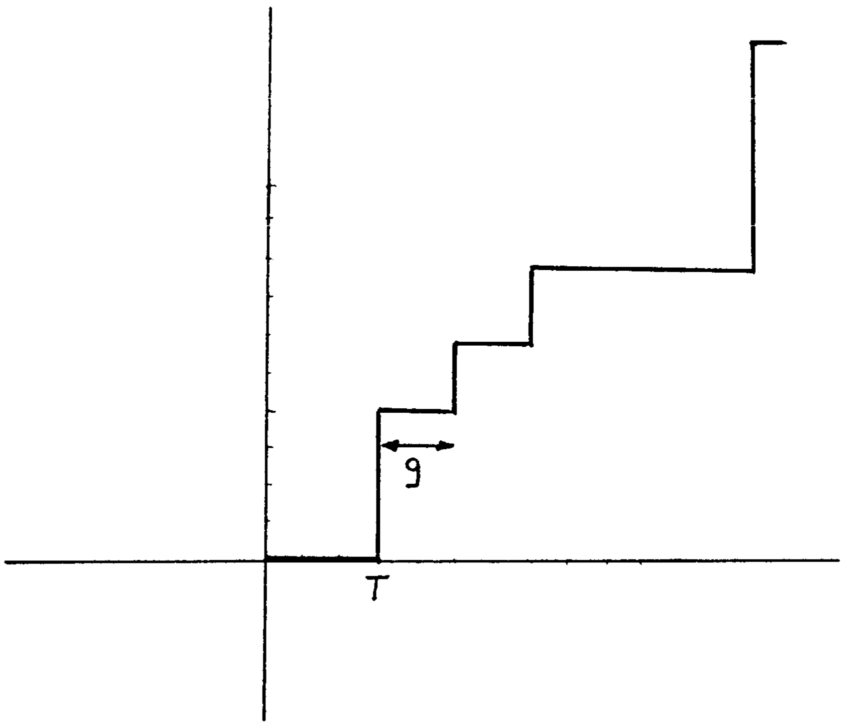


figure 5 quantizer characteristic

The quantizer can be characterized by:

- the threshold
- the linear part, in order to cope with small coefficients
- non-linear part to cope with the possible dynamic range of the coefficients within a block.

The quantized coefficients are Huffman coded.

The quantizer is adapted after two rows of the luminance due to the chrominance block size which is also 8 x 8 i.e. a physical blocksize of 16 x 16 due to the sub-sampling.

The quantiser stepsize g for the DCT-coefficients, can be adapted after two lines of luminance blocks. This information has to be transmitted as side information. The possible differences compared to the previous quantizer choice are (-4,-2,-1,0,+1,+2,+4) .

The difference between the consequtive rows should not exceed 4 otherwise a visible band might appear after decoding. The number of bits used after each two rows is:

-4	4 bits	1 1 1 1
+4	4 bits	1 1 1 0
-2	4 bits	1 1 0 1
+2	4 bits	1 1 0 0
-1	3 bits	1 0 1
+1	3 bits	1 0 0
0	1 bit	0

The coder can make a choice out of 20 stepsizes i.e.

2,4,6,8,10,12,14,16,18,20,22,24,27,30,33,36,40,45,52,64.

ADDRESSING

After quantization a scan for columns which contains only zeros is made. Run length coding of consequtive zeros can be applied in order to decrease the redundancy even further. The horizontal and vertical pointers "hor" and "ver" indicate the smallest area with non-zeros :

```
            ver
             |
             V
    0  1 2 3 4 5 6 7 8

h   1  |x|x|_|_|_|_|_|_|
o   2  |0|0|_|_|_|_|_|_|
r --> 3  |0|x|_|_|_|_|_|_|
    4  |_|_|_|_|_|_|_|_|
    5  |_|_|_|_|_|_|_|_|
    6  |_|_|_|_|_|_|_|_|
    7  |_|_|_|_|_|_|_|_|
    8  |_|_|_|_|_|_|_|_|
```

	hor	ver
0	0	-
1	10	0
2	110	10
3	1110	110
4	11110	1110
5	111110	11110
6	1111110	111110
7	11111110	1111110
8	11111111	1111111

For the above given example the overhead is six bit. Due to the transformed displaced frame difference it is preferable to use variable length coding instead of uniform adressing.

BUFFER CONTROL

In this paragraph the control unit is described in order to explain the typical degradation of this coder. To have a stable regulation the following hypothesis is used:

$$\text{CONTROL} = \frac{FD_{\text{row of block}}}{FD_{\text{total}}} \times \frac{\text{allowed number of bits to spend for the frame}}{\text{number of bits spend for the row of blocks}}$$

CONTROL should have the value 1.
Two parameters are used :

- the partial frame difference,

- the stepsize which is adapted AFTER coding of each row.

The value of CONTROL is not used but (CONTROL - 1) denoted by R. In the case of a too course quantizer, CONTROL will decrease. But for CONTROL equals 1 we should adapt the quantizer and use a smaller stepsize. From control theory we know that these operations can cause instability. In order to have a quicker and more stable response the following procedure is used;

Use the sign of R and tendancy of R. Several possebilities can be distinguished then:

- decreasing positive R , do not change the stepsize

- decreasing negative R , decrease the stepsize.

- increasing negative R , do not change anything,.

- increasing positive R increase stepsize.

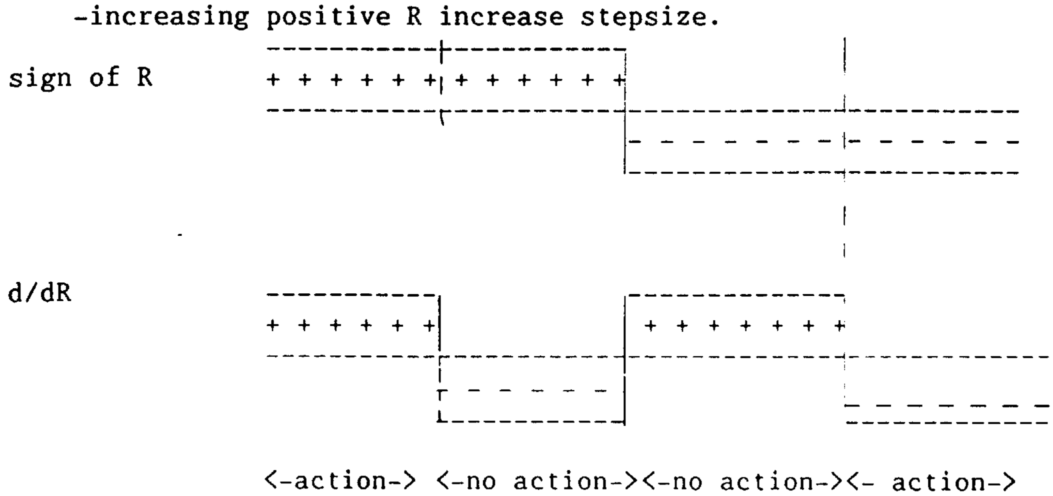


figure 6 character of R

SUMMARY

The described codes uses:

1 - blocksize 8 x 8
2 - motion estimation based on truncated full search
3 - maximum displacement +/- 7 pixels
4 - Variable length coding is applied on:
    displacement vectors
    bits indicating coded/non-coded block
5 - gain criterium for choice processing displaced block
    or orthogonal projection (FD)
6 - buffer control adapting to energy content of rows.

RESULTS

In the next part the results are given. ~~ratio~~.

The sequences are:

The bitrate ~~rate~~ 60 kbit/s; frame rate 7.5 Hz.

1 - Miss America	reduced intermediate	30 frames
2 - Miss America	intermediate	30 frames

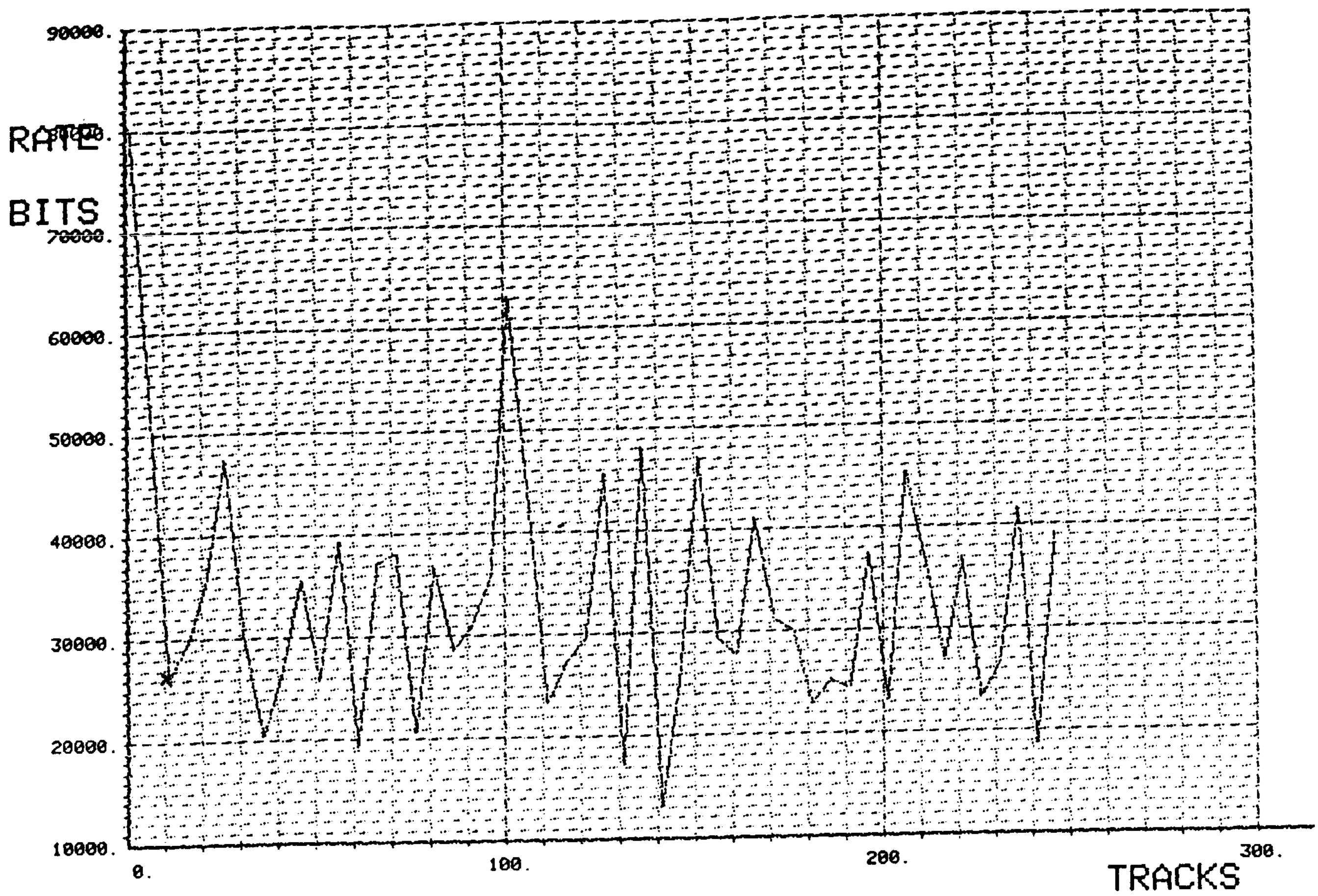
The ~~frame~~ bit rate 300 kbit/s; frame rate 10 Hz.

3 - COST (split- Trevor)	intermediate	48 frames
4 - Miss America	intermediate	30 frames
5 - Checked Jacket	intermediate	20 frames
6 - Graphics	intermediate	2 frames repeated 10 times

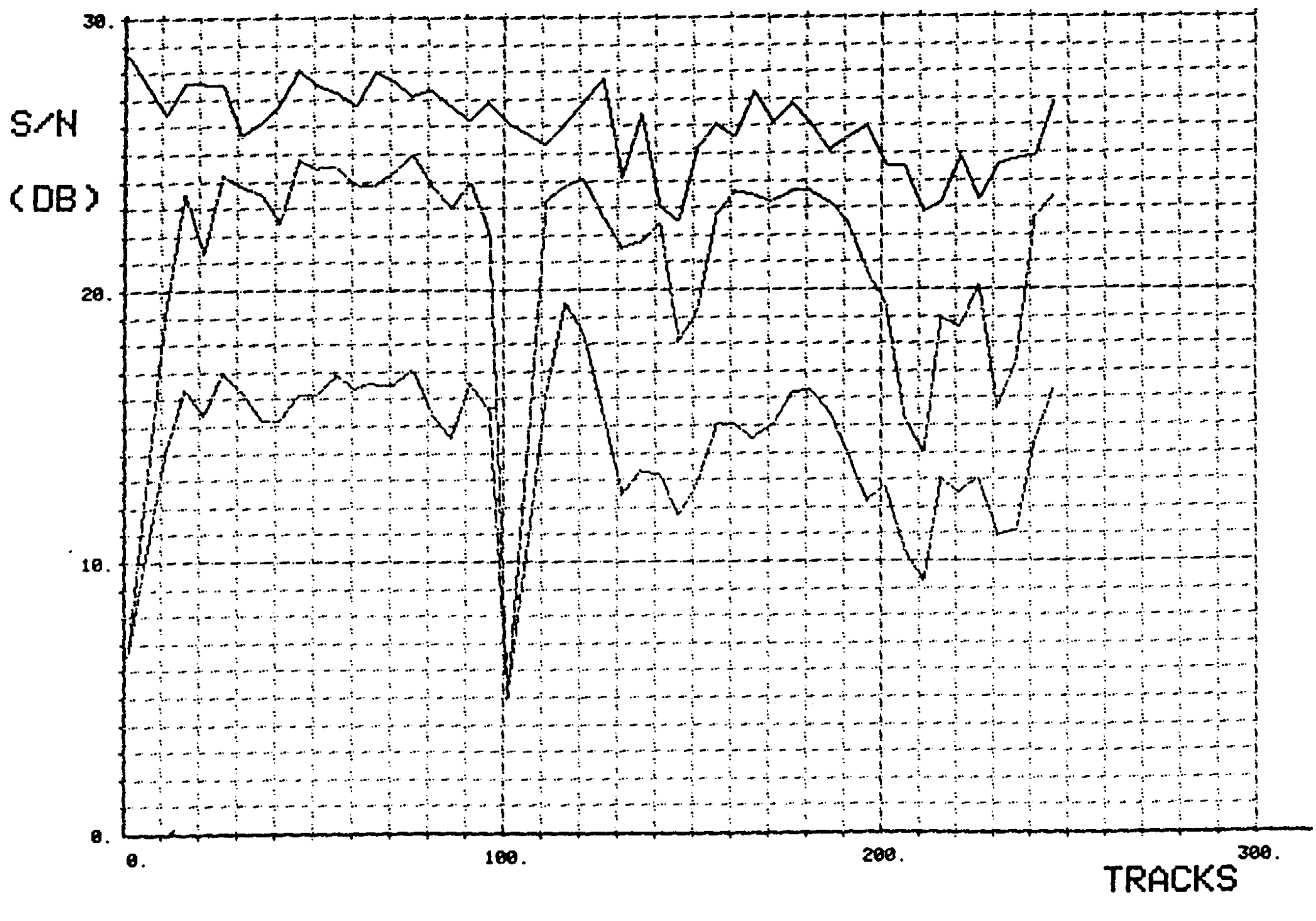
The graphs of the coding results are given ;

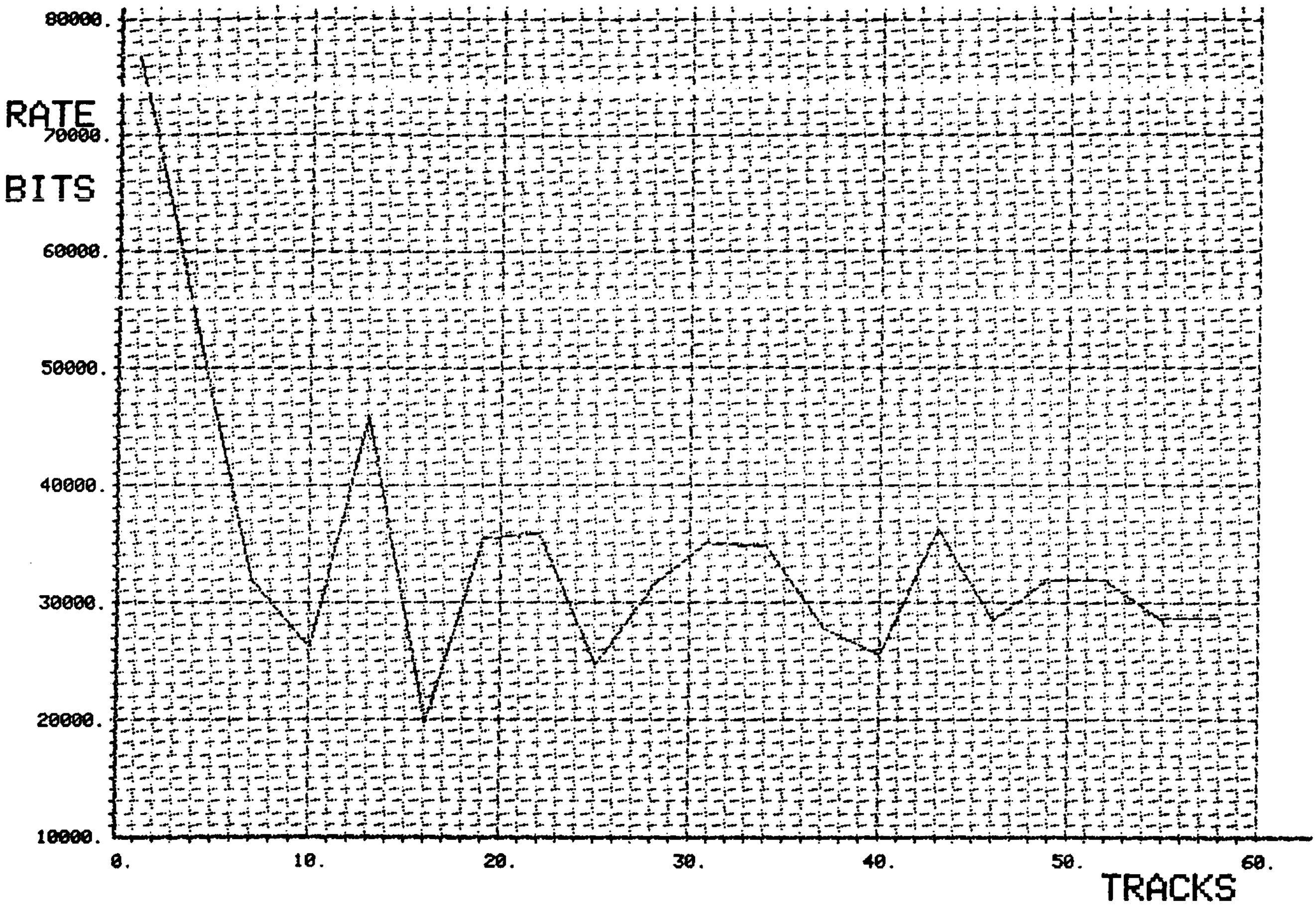
- 1 the bitrate

- 2 the signal to noise ratio.

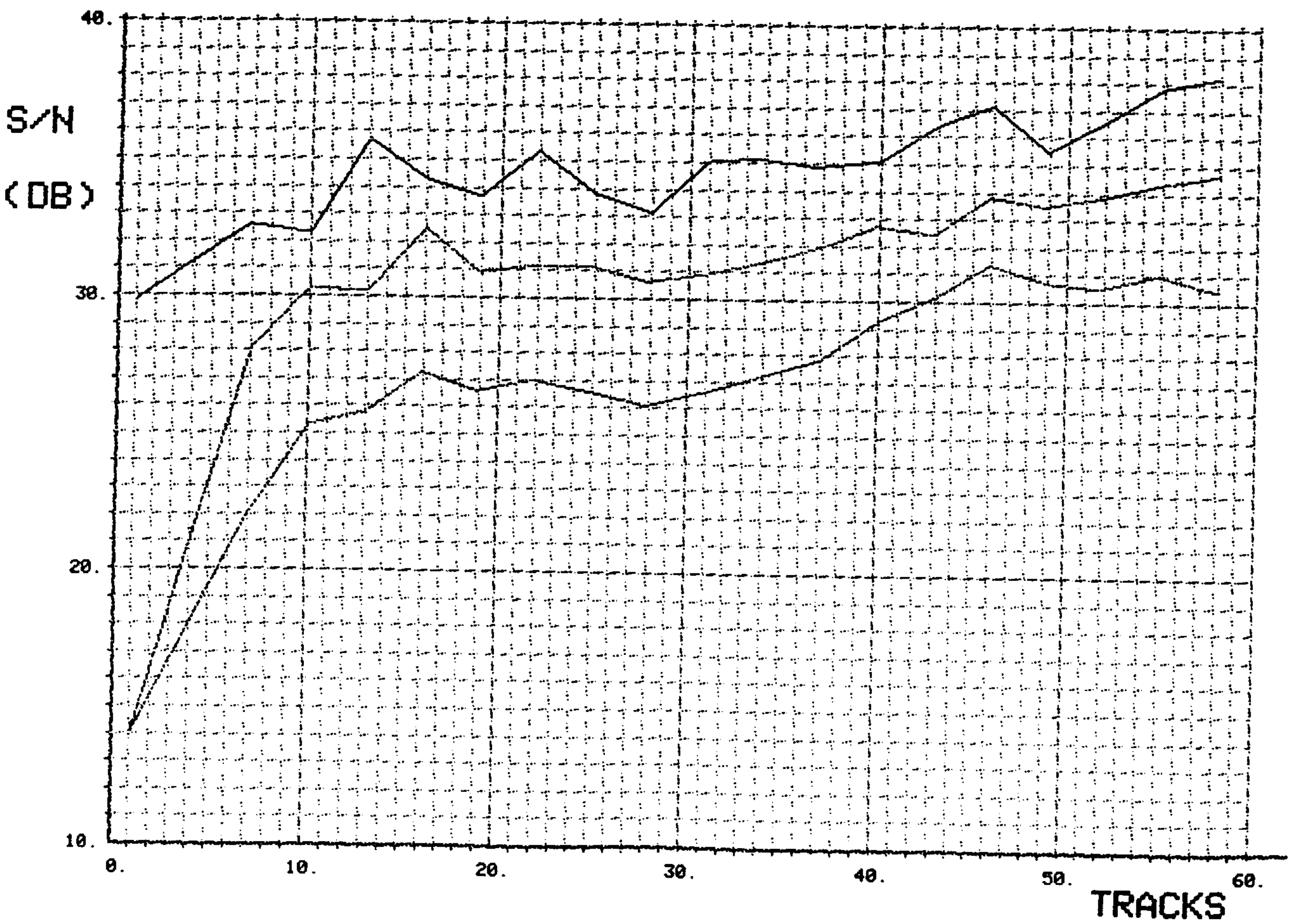


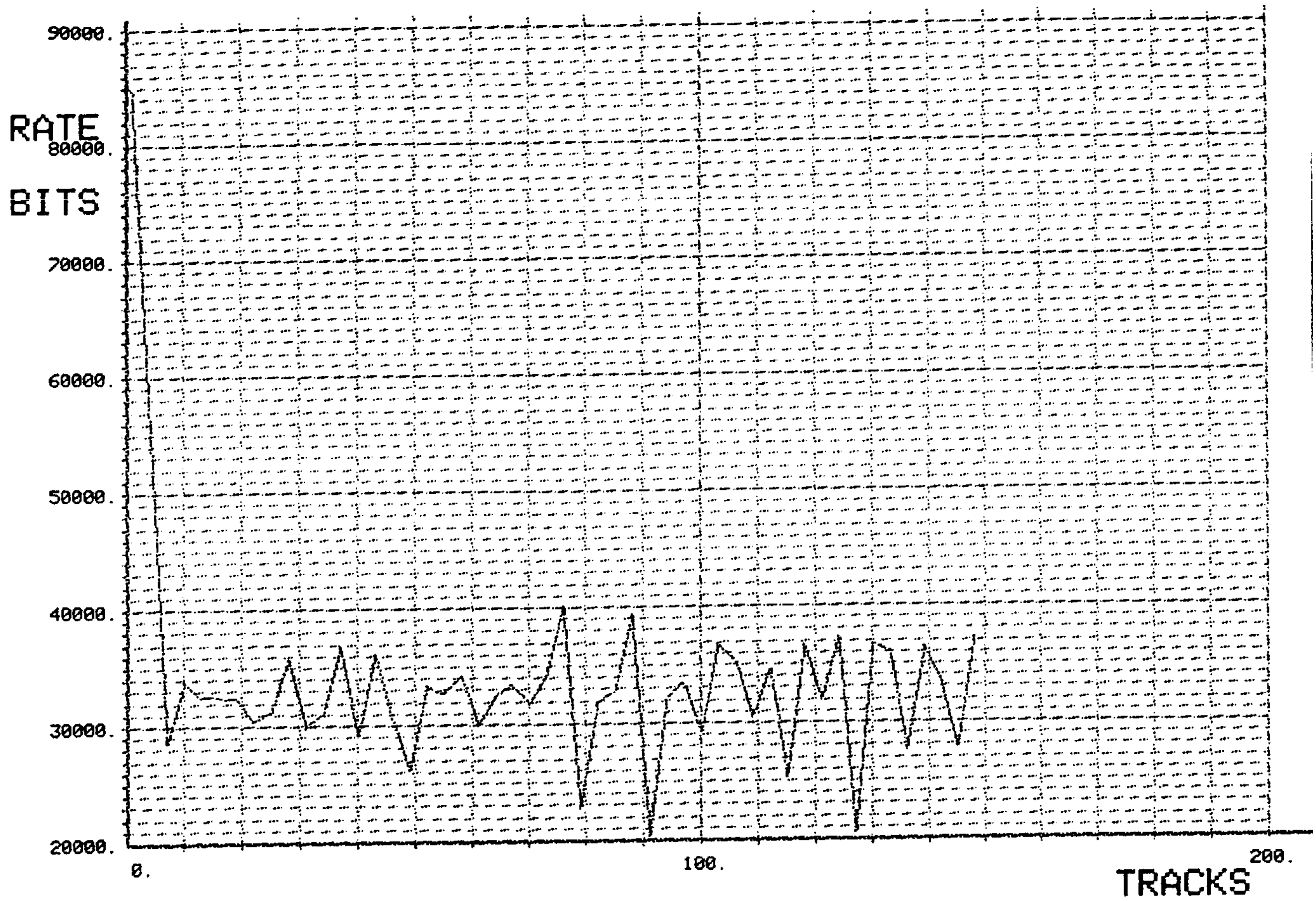
SPLIT-300, IF, 10 HZ



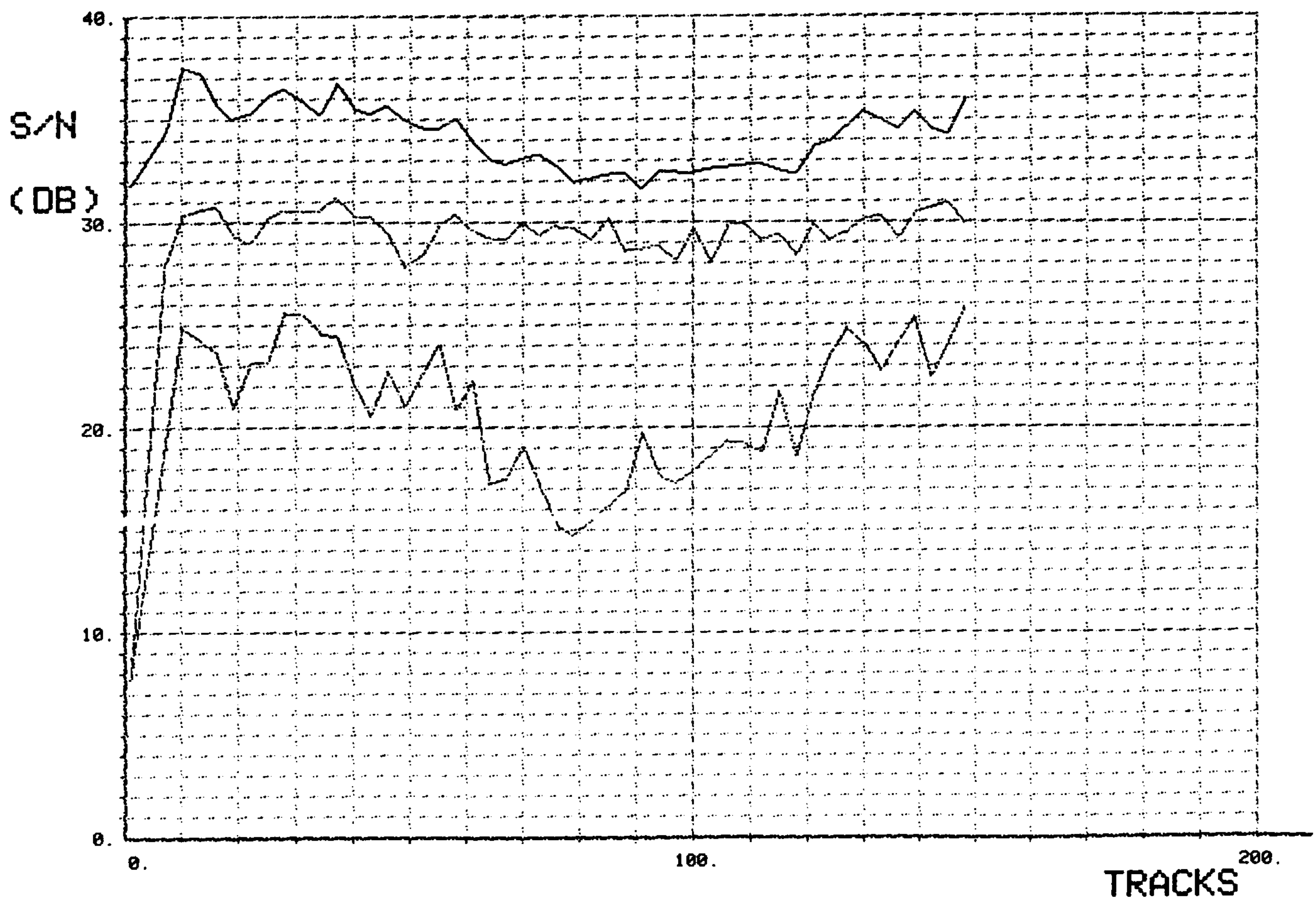


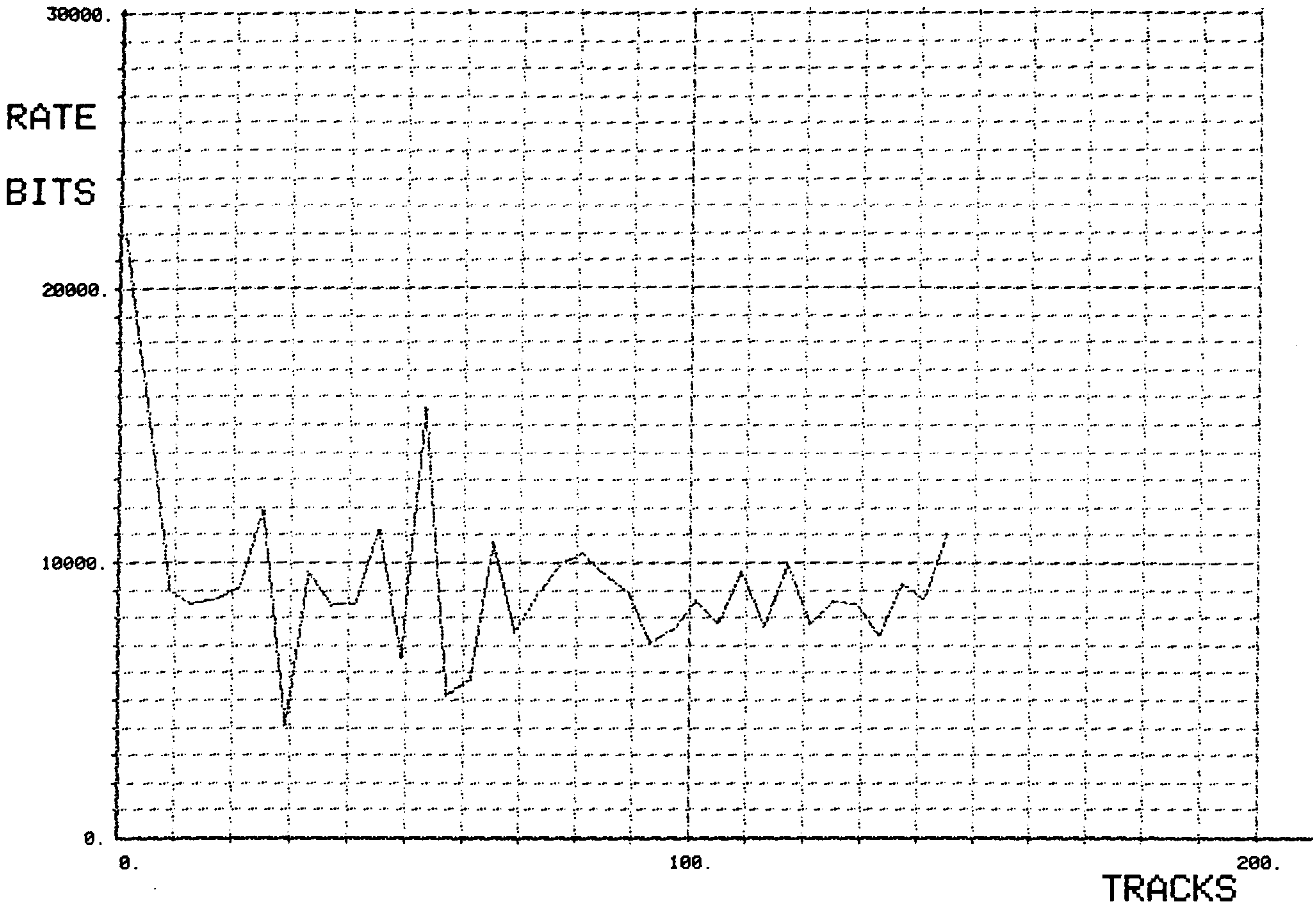
JACK-300, IF, 10 HZ





MISS-300, IF, 10 HZ





MISS-64, IF, 7.5 HZ