C.C.I.T.T.
STUDY GROUP XV WP XV/1
Specialist Group on Picture Coding
Orlando, Florida December 6-9, 1988

#442

English Version

Source: AT&T Bell Laboratories

Title: Progress on Inverse DCT Accuracy Specifications

This document contains copies of correspondence between parties interested in chip manufacture.

FACSI

DATE: 3/30, 1988 NO. PAGES INCL

FROM:

NAME:	Barry G. Haskell
TELEPHONE #:	(201) 949-5459
FACSIMILE #:	(201) 949-3697

TO:

J. Borel	Thomson - France
M. Liou	Bellcore - NJ
M. H. Moehrmann	Siemans - FRG
F. Molo	Telettra-I
W. A. L. Heijnemans	Philips-N
John Vigilante	Plessy, Mass.
Sadao Takahashi	Toshiba R&D
M. D. Carr	BTRL
Dr. Welzenbach	ANT
Mr. R[illegible]	

AT&T
Bell Laboratories

Crawfords Corner Road
Holmdel, New Jersey 07733
Phone (201) 949-3000

FACSIMILE

DATE: 10/12/88 NO. PAGES INCLUDING COVER SHEET: 4

FROM:

NAME: Barry G. Haskell LOCATION: Holmdel, NJ (USA)

TELEPHONE #: (201) 949-5459 ROOM #: 4C-538

FACSIMILE #: (201) 949-3697

TO:

NAME	*LOC.*	*PHONE#*	*FACSIMILE #*
Rolf Singer	FRG	+49 7191 13 3524	+49 7191 13 5387
Vernon Shaw	MH	+1 201 582 4505	+1 201 582 5654
~~Vincent Considine~~	~~UK~~	~~+44~~ 372-50581	+44 372 53410
Jean-Claude Carlach	France	+33 9902 4765	+33 9902 4098
Jean-Claull	France	+33 1 30 60 7000	+33 1 30 64 0693
Yasuhiro Kosugi	Japan	+81 44 717 1111	+81 44 751 7655
Alain Artieri	France	+33 1 4581 7991	+33 1 4580 4036
Sylvdin Kritter	France	+33 7649 3600	+33 7649 3646

To: Respondents to CCITT Request for IDCT

Dear Sirs and Mesdames:

Thank you for your responses so far to the CCITT request for IDCT chip information. Based on further computer simulations and hardware studies, we think we are very close to a performance specification for ISDN video codecs (see attached). Could you examine the proposed method and let us know your thoughts?

Please send responses to:

Barry Haskell
AT&T Bell Labs - 4C538
Holmdel, NJ 07733 USA
telephone: +1 201 949 9459
telefax : +1 201 949 3697

with a copy to:

Mr. Sakae Okubo
NTT Human Interface Labs
telefax: +81 468 59 2829

CCITT Specialists Group on Visual Telephony

Proposed Specification for Inverse DCT Chips

1. Generate random integer pixel data values in the range -L to +H according to the attached random number generator (C version). Arrange into 8x8 blocks. Data sets of 10000 blocks each should be generated for (L=256, H=255), (L=H=5) and L=H=300).
2. For each 8x8 block, perform a separable, orthonormal, matrix multiply, Forward Discrete Cosine Transform (FDCT) using at least 64-bit floating point accuracy.
3. For each block, round the 64 resulting transformed coefficients to the nearest integer values. Then clip them to the range -2048 to +2047. This is the 12-bit input data to the inverse transform.
4. For each 8x8 block of 12-bit data produced by step 3, perform a separable, orthonormal, matrix multiply, Inverse Discrete Cosine (IDCT) using at least 64-bit floating point accuracy. Round the resulting pixels to the nearest integer, and clip to the range -256 to +255. These blocks of 8x8 pixels are the "reference" IDCT output data.
5. For each 8x8 block of 12-bit data produced by step 3, use the proposed IDCT chip or an exact-bit simulation thereof to perform an Inverse Discrete Cosine Transform. Clip the output to the range -256 to +255. These blocks of 8x8 pixels are the "test" IDCT output data.
6. For each of the 64 IDCT output pixels, and for each of the 10,000 block data sets generated above, measure the peak, mean and mean square error between the "reference" and "test" data.
7. For any pixel, the peak error should not exceed 1 in magnitude.
   For any pixel, the mean square error should not exceed 0.06.
   Overall, the mean square error should not exceed 0.02.
   For any pixel, the mean error should not exceed 0.015 in magnitude.
   Overall, the mean error should not exceed 0.0015 in magnitude.
8. All-zeros in must produce all-zeros out.
9. Rerun the measurements using exactly the same data values of step 1, but change the sign on each pixel.

To Dr. Haskell
From S. Okubo

October 24, 1988

Figures attached to my letter (Our Ref: SGXVL570, Oct. 14, 1988)

Note: Sheet 3' is added, which shows the results with same conditions as Sheet 3 except the number of test blocks. It has been clarified that the nx384 Flexible Hardware design (Coefficient = 14 bits, Taransfer = 16 bits truncation as in Doc. #249) does not meet the current IDCT specification.

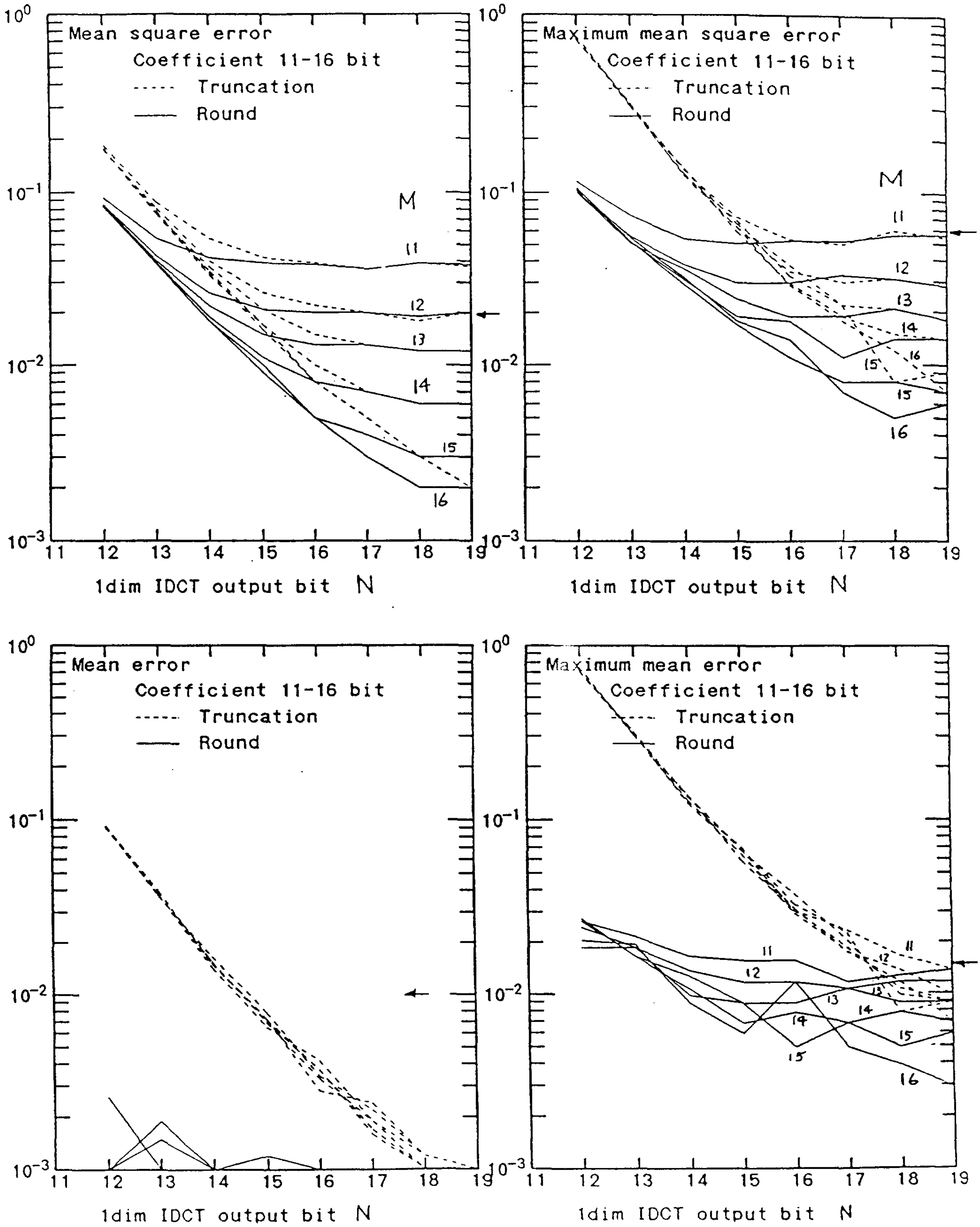
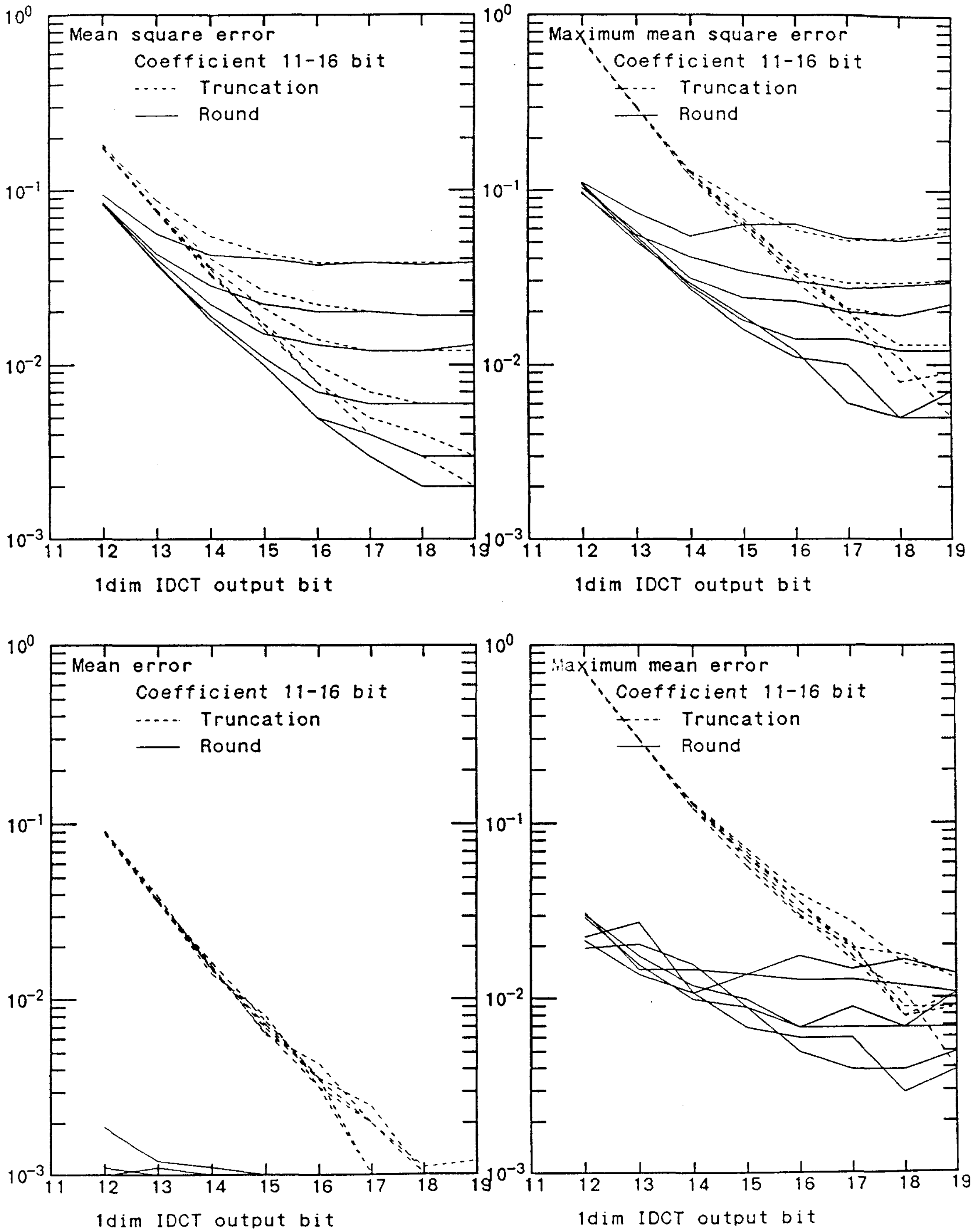
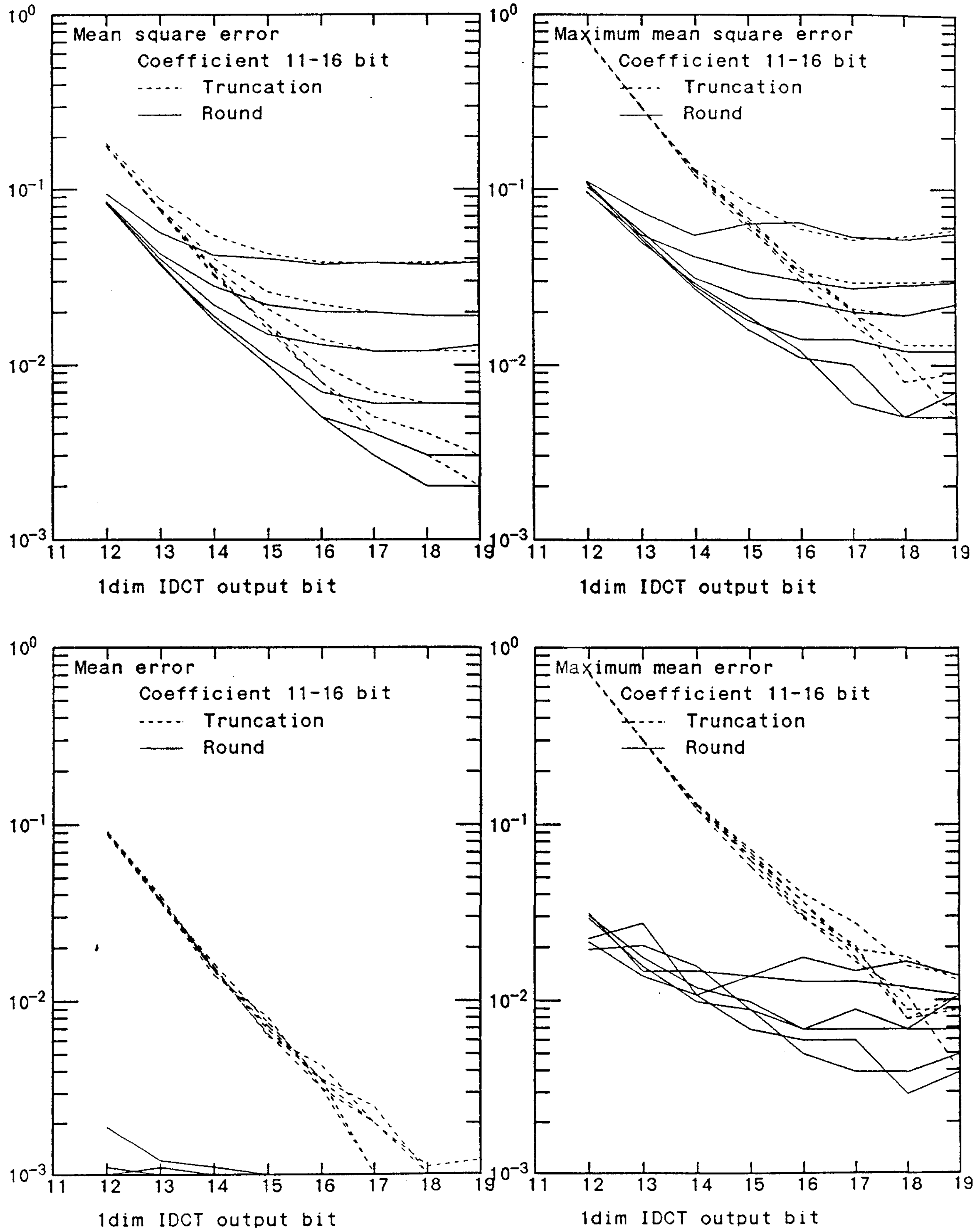


Figure 3 Error between IDCT outputs after 9-bit rounding (Zt - Zr) for ideal IDCT (M.N >> 20) as reference. L=12

2

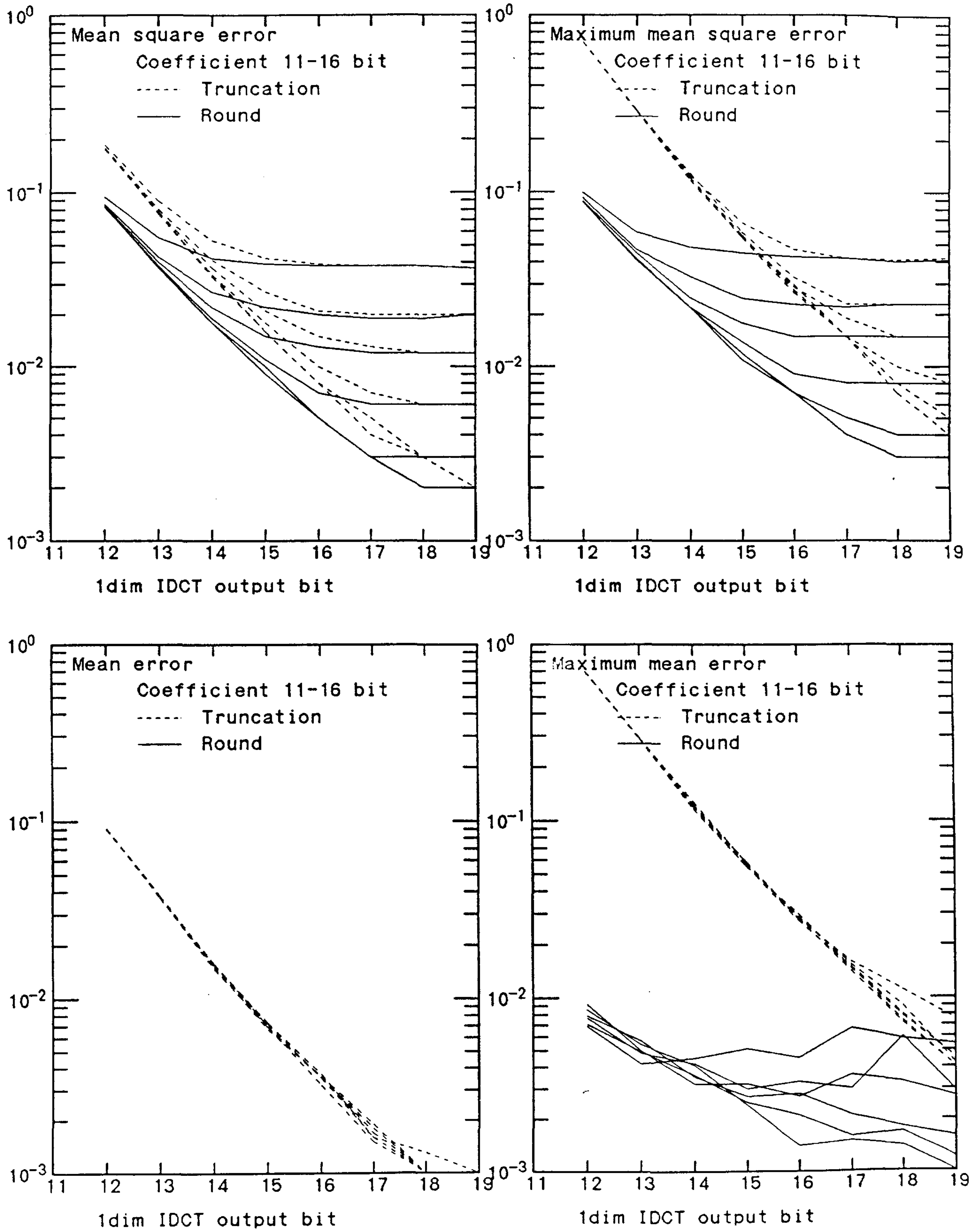


Ref=Floating, new_rand()



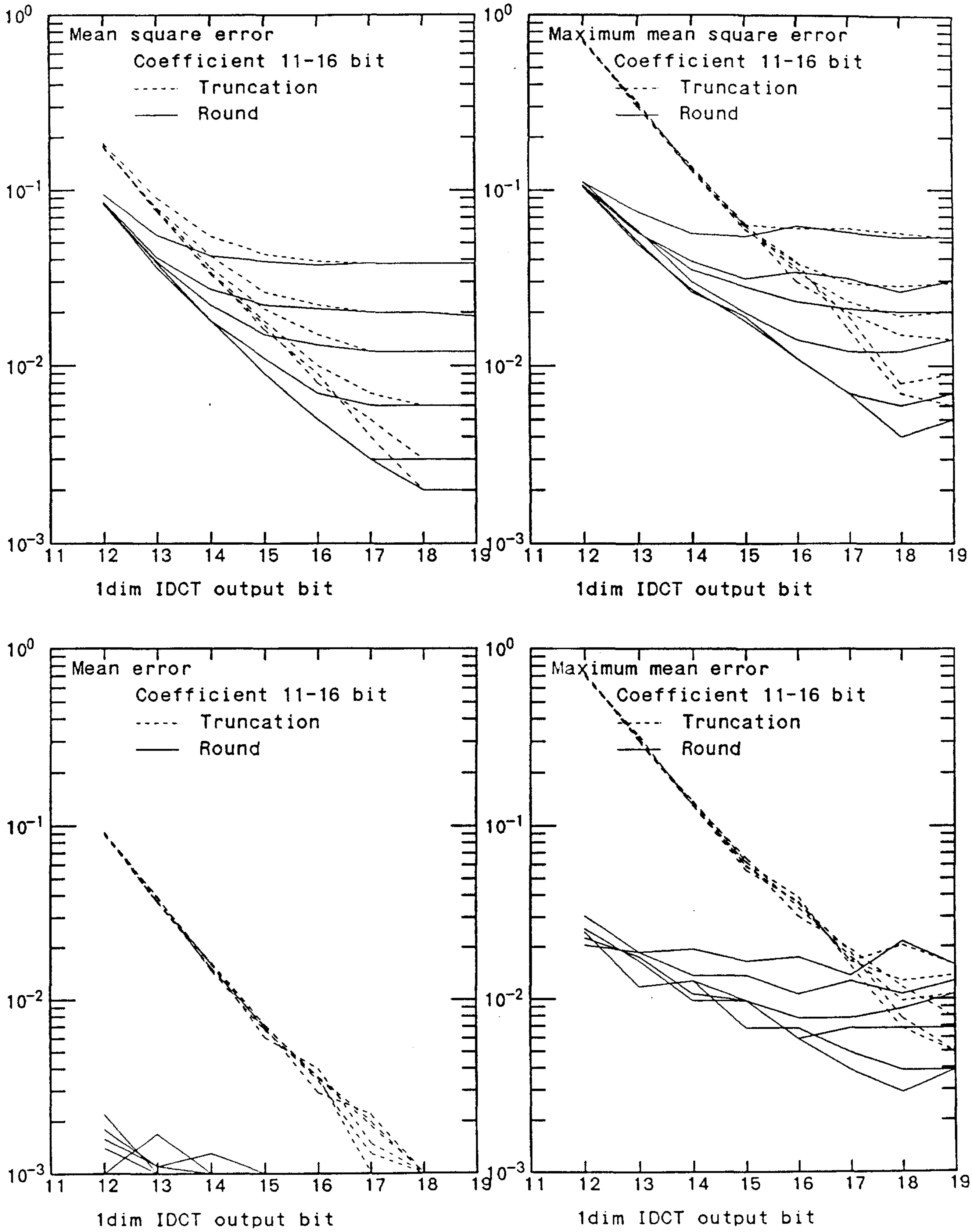
Ref=Floating, new_rand(), H

③' Oct. 22, 1988



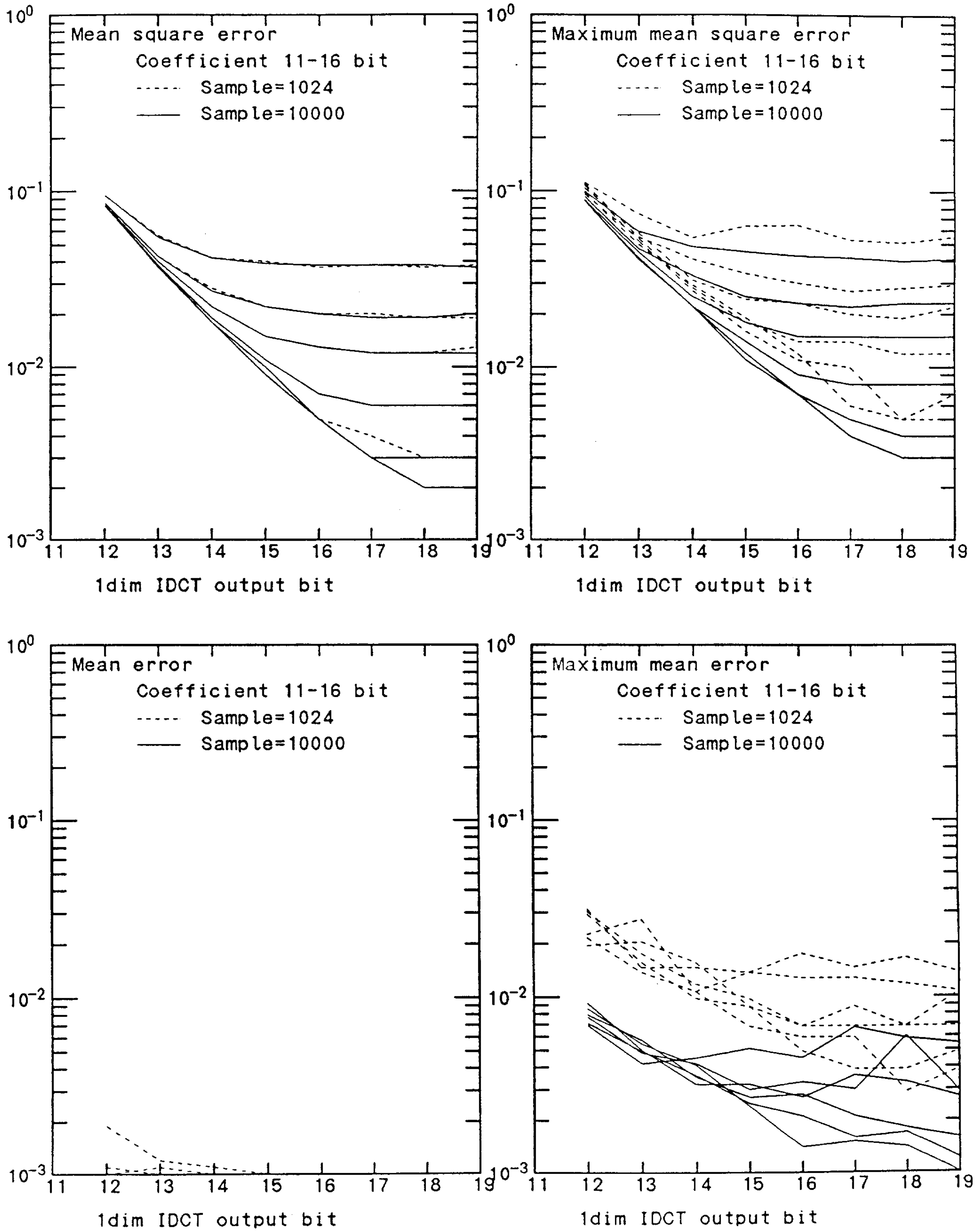
N=10,000 new_rand()

4



Ref=Floating, new_rand(), V

5



Ref=Floating, Test=Round

```
Data range -256  to  255

enter no. of blocks to be simulated:no. of blocks simulated = 10000

pel means
-0.000100 0.000900 0.002400 -0.001000 0.000400 -0.001100 -0.000400 -0.000900
-0.000800 -0.000200 0.001700 -0.000600 -0.000300 0.000400 0.002400 0.000500
0.001300 0.000200 0.001300 0.000600 0.000900 -0.001500 0.000800 -0.001500
0.001200 -0.000800 -0.000300 -0.000500 0.000400 0.000300 -0.000200 0.001500
-0.001400 0.000000 0.001600 0.000400 -0.000800 -0.000600 0.001100 0.001300
-0.000800 -0.001400 0.001100 0.000000 -0.001500 0.001100 0.000300 0.000300
-0.000500 -0.001500 0.001000 -0.000200 -0.000800 -0.000800 0.001000 -0.000100
0.001800 -0.000500 0.002100 -0.000200 -0.000100 0.000300 -0.000800 0.001200
peak pel mean error = 0.002400

pel mse's
0.010100 0.009700 0.008800 0.009400 0.011400 0.011300 0.011400 0.010300
0.010600 0.010600 0.010100 0.011000 0.011100 0.012800 0.011000 0.010500
0.010500 0.010600 0.012100 0.008800 0.009900 0.010700 0.009800 0.010900
0.009400 0.009800 0.010500 0.009900 0.009600 0.010100 0.010200 0.008900
0.009600 0.010000 0.011000 0.008600 0.009200 0.009800 0.009500 0.009300
0.011600 0.012000 0.009300 0.008400 0.010700 0.010500 0.011300 0.007700
0.008900 0.012500 0.010200 0.007800 0.010200 0.010200 0.010000 0.009900
0.009400 0.009900 0.009500 0.010000 0.010500 0.010500 0.011000 0.009800
peak pel mse = 0.012800
overall mse = 0.010166
overall mean error = 0.000150
peak error = 1
```

BARRY G. HASKELL HO-4C538
VISUAL COMMUNICATIONS RESEARCH DEPT.
AT&T BELL LABORATORIES
HOLMDEL, NEW JERSEY 07733

```
Data range -5  to  5

enter no. of blocks to be simulated:no. of blocks simulated = 10000

pel means
0.000800 0.001000 -0.001500 -0.000600 -0.001500 0.001900 0.000100 -0.000700
0.000100 -0.000500 0.000000 0.001100 0.001100 0.000100 0.000000 0.000600
0.001100 -0.001200 -0.000300 0.000100 0.003400 0.001100 -0.000500 0.001300
0.001200 0.000400 -0.000900 -0.000100 -0.000800 0.000500 -0.001400 -0.001700
-0.001100 0.000400 -0.000100 0.001800 -0.000600 0.000800 0.000000 0.001000
0.001800 0.002600 -0.001100 0.000400 -0.001000 -0.000900 0.000500 0.000600
0.002300 0.000600 0.000000 0.000200 0.000800 0.001500 -0.000800 -0.001400
-0.000900 -0.001300 0.000200 0.000300 -0.000300 -0.001100 0.000000 -0.001400
peak pel mean error = 0.003400

pel mse's
0.009200 0.012000 0.011300 0.011800 0.008500 0.010700 0.010300 0.009700
0.010100 0.010900 0.010000 0.010100 0.010500 0.011900 0.009800 0.011000
0.009100 0.010600 0.011500 0.008700 0.009000 0.011100 0.012100 0.011900
0.009000 0.011400 0.010700 0.008700 0.011600 0.009500 0.010600 0.009700
0.009700 0.010000 0.010900 0.011200 0.009800 0.012000 0.009600 0.010800
0.010400 0.012400 0.011700 0.012400 0.011000 0.012700 0.011900 0.010400
0.010300 0.012400 0.010600 0.010600 0.011600 0.010500 0.013000 0.009400
0.009100 0.009500 0.011200 0.009900 0.008900 0.009500 0.011600 0.010000
peak pel mse = 0.013000
overall mse = 0.010594
overall mean error = 0.000125
peak error = 1
```

```
Data range -300  to  300

enter no. of blocks to be simulated:no. of blocks simulated = 10000

pel means
0.001200 0.002000 -0.000300 -0.000800 -0.002700 -0.001500 -0.000100 -0.001000
-0.001200 0.001200 -0.000300 -0.000100 0.001000 0.000000 0.000500 0.000600
0.001800 -0.000900 -0.000400 0.001900 0.000000 -0.002100 -0.000200 -0.002300
-0.000400 -0.001100 -0.001900 0.000800 0.000900 -0.000100 0.000700 -0.000900
-0.001200 -0.001000 0.001200 -0.000700 0.001000 -0.000500 0.000200 -0.000400
-0.000300 0.000600 0.000600 -0.001100 0.000700 0.000000 -0.000900 0.002100
-0.002300 0.000100 0.000000 0.000400 0.001300 -0.001200 -0.000800 0.001800
-0.000200 -0.000100 -0.000600 0.000400 -0.000600 0.000200 -0.000100 -0.000300
peak pel mean error = 0.002700

pel mse's
0.009000 0.009000 0.009100 0.009600 0.008100 0.007900 0.008500 0.008200
0.006000 0.010200 0.008300 0.008500 0.011200 0.008000 0.009500 0.009800
0.008800 0.009900 0.009200 0.007500 0.008800 0.010500 0.009400 0.007700
0.007400 0.008700 0.008100 0.008200 0.007100 0.007900 0.010300 0.009500
0.008400 0.011600 0.009200 0.008500 0.008200 0.010300 0.009000 0.007400
0.008500 0.008000 0.010000 0.008700 0.008500 0.008200 0.009700 0.008500
0.008500 0.007700 0.008200 0.007400 0.010700 0.009200 0.007800 0.010400
0.008000 0.009700 0.007800 0.007200 0.009800 0.009800 0.009900 0.008300
peak pel mse = 0.011600
overall mse = 0.008797
overall mean error = -0.000116
peak error = 1
```

FACSIMILE TRANSMISSION

Fax:+1 201 949 3697
TO:Dr.Barry G.Haskell
Room HO-4C538
ATT Bell Labs.
Crawfords Corner Road
Holmdel, NJ 07733
USA

FROM: Fumio Sugiyama
Video Sys. & Tech. Lab.
R & D Center TOSHIBA
1 Komukai Toshiba-cho
Saiwai-ku Kawasaki 210
JAPAN
phone: +81 044 549 2151
fax: +81 044 555 2074
NOVEMBER 18th 1988

No. of pages including this one: 6

Content:On the results for IDCT new criterion in annexC of Doc.#396.
Please contact me on this problem.

November 18,1988
TOSHIBA

IDCT Specification Simulation Results

1. Abstract

In order to examine the number of bits necessary for DCT chip to satisfy the criterion on Doc.#397, two simulations have been carried out. One is for Matrix DCT and the other is for Toshiba's fast DCT.

2. Condition of simulation

Simulations have been carried out on the following condition:

(1) Matrix DCT

* number of bits for transform coefficients and the output of the first 1-D IDCT are the same

* caluculation results are rounded in a fixed bit length after accumulation

(2) Toshiba's fast DCT

* number of bits for transform coefficients and the output of the first 1-D IDCT are the same

* caluculation results are rounded in a fixed bit length after all multiplication and accumulation

3. Simulation results and conclusions

Simulation results are shown in Table 1 to Table 4 for Matrix DCT and Table 5 to Table 8 for fast DCT.
It has been clarified from these results that,

(1) at least 14 bits are necessary for Matrix DCT

(2) at least 16 bits are necessary for Toshiba's fast DCT to satisfy the criterion on Doc.#397.

END.

Table 1 12bits(Matrix)

	PEAK-ERROR	MSE(PIXEL)	MSE(BLOCK)	ME(PIXEL)	ME(BLOCK)
(-256,255)	1	0.0893	0.0857	0.0056	0.000064
	1	0.0893	0.0857	0.0076	-0.000006
(-5,5)	1	0.0888	0.0828	0.0078	-0.000561
	1	0.0888	0.0828	0.0066	0.000561
(-300,300)	1	0.0806	0.0739	0.0079	-0.000180
	1	0.0804	0.0738	0.0081	0.000213

Table 2 14bits(Matrix)

	PEAK-ERROR	MSE(PIXEL)	MSE(BLOCK)	ME(PIXEL)	ME(BLOCK)
(-256,255)	1	0.0224	0.0193	0.0031	-0.000066
	1	0.0224	0.0192	0.0039	0.000069
(-5,5)	1	0.0206	0.0181	0.0027	-0.000231
	1	0.0206	0.0181	0.0032	0.000231
(-300,300)	1	0.0202	0.0168	0.0032	0.000228
	1	0.0203	0.0168	0.0033	-0.000225

Table 3 16bits(Matrix)

	PEAK-ERROR	MSE(PIXEL)	MSE(BLOCK)	ME(PIXEL)	ME(BLOCK)
(-256,255)	1	0.0062	0.0471	0.0015	-0.000033
	1	0.0062	0.0471	0.0020	0.000034
(-5,5)	1	0.0068	0.0449	0.0024	-0.000041
	1	0.0068	0.0449	0.0015	0.000041
(-300,300)	1	0.0057	0.0424	0.0020	-0.000069
	1	0.0057	0.0423	0.0017	0.000069

Table 4 18bits(Matrix)

	PEAK-ERROR	MSE(PIXEL)	MSE(BLOCK)	ME(PIXEL)	ME(BLOCK)
(-256,255)	1	0.0019	0.00115	0.0009	0.0000140
	1	0.0019	0.00115	0.0012	-0.0000140
(-5,5)	1	0.0024	0.00119	0.0007	-0.0000375
	1	0.0024	0.00119	0.0009	0.0000375
(-300,300)	1	0.0022	0.00103	0.0008	-0.0000578
	1	0.0022	0.00102	0.0006	0.0000578

Table 5 14bits(Fast)

	PEAK-ERROR	MSE(PIXEL)	MSE(BLOCK)	ME(PIXEL)	ME(BLOCK)
(-256,255)	1	0.109	0.0826	0.0067	-0.000142
	1	0.109	0.0825	0.0090	0.000180
(-5,5)	1	0.107	0.0800	0.0056	0.000211
	1	0.107	0.0800	0.0069	-0.000211
(-300,300)	1	0.0906	0.0702	0.0053	0.000228
	1	0.0906	0.0702	0.0045	-0.000192

Table 6 15bits(Fast)

	PEAK-ERROR	MSE(PIXEL)	MSE(BLOCK)	ME(PIXEL)	ME(BLOCK)
(-256,255)	1	0.0493	0.0386	0.0030	-0.000338
	1	0.0493	0.0385	0.0055	0.000369
(-5,5)	1	0.0464	0.0366	0.0037	0.000217
	1	0.0464	0.0366	0.0026	-0.000217
(-300,300)	1	0.0421	0.0329	0.0049	0.000100
	1	0.0422	0.0329	0.0045	-0.000064

Table 7 16bits(Fast)

	PEAK-ERROR	MSE(PIXEL)	MSE(BLOCK)	ME(PIXEL)	ME(BLOCK)
(-256,255)	1	0.0236	0.0183	0.0020	-0.000411
	1	0.0235	0.0183	0.0031	0.000431
(-5,5)	1	0.0230	0.0176	0.0028	0.000089
	1	0.0230	0.0176	0.0027	-0.000089
(-300,300)	1	0.0207	0.0158	0.0024	0.000144
	1	0.0207	0.0158	0.0028	-0.000125

Table 8 17bits(Fast)

	PEAK-ERROR	MSE(PIXEL)	MSE(BLOCK)	ME(PIXEL)	ME(BLOCK)
(-256,255)	1	0.0124	0.00911	0.0021	-0.0000875
	1	0.0124	0.00911	0.0023	0.0000906
(-5,5)	1	0.0126	0.00915	0.0028	-0.0000188
	1	0.0126	0.00915	0.0017	0.0000188
(-300,300)	1	0.0108	0.00780	0.0014	-0.0000422
	1	0.0108	0.00779	0.0034	0.0000547

bgh last

FROM : Alain Artieri
TO   : Barry G. Haskell

DEAR SIR

SGS THOMSON IS INTRODUCING A MULTI-WINDOW DCT CHIP. THIS COMPONENT IS TODAY AVAILABLE IN OUR VIDEO DIVISION.
WE HAVE CAREFULLY EXAMINED THE CCITT PROPOSITION.
ACCORDING TO THE PROPOSED SPECIFICATION METHOD FOR IDCT 8*8, THE CHIP MEET THE FOLLOWING FIGURES :

- PEAK ERROR : 1
- PEAK MSE : 0.112
- OVERALL MSE : 0.052
- PEAK MEAN ERROR : 0.019
- OVERALL MEAN ERROR : 0.06

FOR A FUTURE SPECIFIC 8*8 VERSION OF THE CHIP, INCLUDING SOME ACCURACY OPTIMIZATION, WE COULD MEET THE FOLLOWING FIGURES :

- PEAK ERROR : 1 ✓
- PEAK MSE : 0.042 ✓
- OVERALL MSE : 0.029 X 0.02
- PEAK MEAN ERROR : 0.013 ✓
- OVERALL MEAN ERROR : 0.005 X 0,0015

THIS MEET THE PROPOSED CCITT REQUIREMENTS, EXCEPT FOR THE OVERALL MEAN ERROR. WHY IS THE REQUIRED VALUE FOR THE OVERALL MEAN ERROR 10 TIMES SMALLER THAN THE REQUIRED VALUE FOR THE PEAK MEAN ERROR ???
WE INSIST ON THE FACT THAT THE COST TO MEET THIS LAST CONSTRAINT, IN TERM OF SILICON AREA, COULD BE RATHER HIGH. COULD THE CCITT RELAX THIS VALUE ? IN OUR OPINION, THIS COULD BE AN ACCEPTABLE TRADE-OFF.
FOR MORE INFORMATION, PLEASE CONTACT :

Alain Artieri
SGS THOMSON MICROELECTRONICS
VIDEO DEDICATED PRODUCTS DIVISION
17 AVENUE DES MARTYRS BP 217
38019 GRENOBLE CEDEX FRANCE
TELEPHONE : (33) 76 49 36 00
TELEFAX : (33) 76 49 36 57

From: Barry G. Haskell
AT&T Bell Labs
Holmdel, NJ 07733
USA

phone: +1 201 949 5459
fax: +1 201 949 3697

To: Mr. Alain Artieri
SGS Thomson Microelectronics
Video Dedicated Products Division
17 Avenue des Martyrs BP 217
38019 Grenoble Cedex FRANCE

tel: +33 76 49 36 00
fax: +33 76 49 36 57

Dear Mr. Artieri,

Thank you for your letter of October 18 and the accompanying data. As you are no doubt aware, the IDCT specifications have been evolving on the basis of computer simulations and real-time hardware experiments. As more results become available, we will be better able to decide the tradeoffs between IDCT accuracy, picture quality and transmission bit-rate. From what we understand now, however, the specifications probably cannot be relaxed a lot more.

In particular, the picture quality seems to be rather sensitive to overall mean error. I suppose this is because it is less randomized than error that has zero mean. Our best evidence from hardware studies is that an overall mean error of 0.003 is acceptable, and a maximum pixel mean error of 0.03 is acceptable. I share your concern that this latter figure may still be too large.

Apportioning the error half to the coder and half to the decoder gives us the current specifications of 0.0015 and 0.015, respectively. Other manufacturers seem to have no trouble with this, as they indicated at Martlesham Heath last May.

I am puzzled by the statement that your data meets the CCITT requirements except for overall mean error. Your overall mse figures of 0.052 and 0.029 are both larger than the proposed CCITT value of 0.020

We look forward to more experimental evidence, perhaps from CNET? Please let us know of any further results, plans or concerns. Thanks.

Sincerely yours,

Barry Haskell

Barry Haskell

cc: Sakae Okubo

Date : 03 Nov 1988 12:30

From : Alain ARTIERI
SGS Thomson Microelectronics
Video Dedicated Products Division
17 Avenue des Martyrs BP 217 phone : +33 76 49 36 00
38019 Grenoble Cedex FRANCE fax : +33 76 49 36 57

Holmdel, NJ 07733 phone : +1 201 949 5459
USA fax : +1 201 949 3697

SUBJECT : IDCT SPECIFICATION

Dear Sir,

We have improved our DCT algorithm in order to meet the last CCITT proposal. The solution is to enlarge the data format to compute the one dimensional DCT. This leads to a definite improvement of accuracy for a negligible overhead of hardware (because of serial arithmetic used in the operative part of our architecture).
Thus, we are able to propose a chip with the following accuracy characteristics, according to the CCITT specification method for IDCT 8*8 :

		proposed :
Peak Error	: 1	(1)
Peak Mean Square Error	: 0.02590	(0.06)
Overall Mean Square Error	: 0.02000	(0.02)
Peak Mean Error	: 0.00410	(0.015)
Overall Mean Error	: 0.00001	(0.0015)

Mean Error Versus Pixel Location

-0.0013	0.0006	0.0002	-0.0022	0.0039	0.0006	0.0004	0.0000
0.0002	-0.0003	-0.0027	0.0009	-0.0025	-0.0014	0.0007	0.0005
0.0006	-0.0006	0.0021	0.0010	0.0000	-0.0007	-0.0007	0.0012
0.0009	-0.0003	-0.0019	0.0041	0.0000	-0.0022	-0.0030	0.0021
0.0003	-0.0020	0.0016	0.0031	0.0025	-0.0025	0.0002	-0.0002
0.0019	-0.0002	0.0001	0.0011	-0.0008	0.0016	0.0016	0.0000
-0.0024	0.0009	-0.0014	-0.0013	-0.0001	-0.0019	-0.0008	-0.0025
0.0006	-0.0012	0.0032	0.0015	0.0015	0.0003	-0.0001	0.0001

Mean Square Error Versus Pixel Location

0.0117	0.0156	0.0186	0.0188	0.0169	0.0180	0.0174	0.0136
0.0156	0.0167	0.0221	0.0233	0.0227	0.0210	0.0191	0.0151
0.0170	0.0226	0.0237	0.0254	0.0240	0.0229	0.0211	0.0200
0.0185	0.0225	0.0259	0.0255	0.0244	0.0244	0.0252	0.0205
0.0177	0.0212	0.0240	0.0249	0.0229	0.0219	0.0218	0.0194
0.0185	0.0220	0.0219	0.0225	0.0254	0.0232	0.0206	0.0172
0.0168	0.0197	0.0178	0.0221	0.0211	0.0207	0.0166	0.0155
0.0120	0.0164	0.0174	0.0217	0.0195	0.0189	0.0155	0.0139

Sincerely yours,

Alain ARTIERI

CC : Dr. Sakae Okubo

Teletta VIMERCATE (MI)-ITALY TRANSMISSION VIA FACSIMILE	DATE : 11-11-88	TIME :
	No of Pages : 1/1	This PAGE No : 1/1
	PRIORITY : ☐ URGENT ☐ NORMAL	
FROM : S. Cucchi	TO : AT&T HOLMDEL NJ (USA)	
MESSAGE No : 6593	ATT.N : MR. BARRY G. HASKELL	
FAX No : 039-68.14.83	FAX No : 201 - 949-3697	
SIGNATURE Cucchi Silvio	PHONE No :	

Dear Mr. G. Haskell

We did simulation with the specifications that you send me by fax 10.10.88 and also we tested the random number generator.
As conclusion we agree with the proposed specification for Inverse DCT chips.
If you need the program for chip simulation written in Pascal Language , I can send you.
Best regards

S. Cucchi

To: B G Haskell, AT&T, Bell Labs
cc: S Okubo, NTT Human Interface Labs

From: Colin Smith, Inmos, Bristol, UK

Date: 27th October 1988.

Subject: Proposed CCITT Specification for Inverse DCT chips

Dear Mr. Haskell,

We have reviewed the proposed specification for the inverse DCT and we have performed some tests on our simulator.

A summary of the worst case results for internal rounding to 14 bits is as follows :
Maximum Peak Error on any pixel = 1
Mean Square Error on any pixel = 0.0227
Average Mean Square Error = 0.0191
Mean Error on any pixel = 0.0049
Average Mean Error = 0.0002

If the internal rounding is to 16 bits the results are as follows :
Maximum Peak Error on any pixel = 1
Mean Square Error on any pixel = 0.0103
Average Mean Square Error = 0.0075
Mean Error on any pixel = 0.0026
Average Mean Error = 0.0002

We do not see any problem with the specification method or meeting the proposed specification but we do have a concern over the random number generator (C version). We are using Microsoft C ver 3 on IBM PC and we see two anomalies.
1/ It produces values of -L and +H with a probability of half that of other values.
2/ There is a very slight non-uniformity if the range does not divide 2^{31} exactly.

The second problem is very small and the first problem can be cured by modification to the program as follows :

```
LONG rand (L,H)
LONG L,H;
{
STATIC LONG randx = 1;
STATIC DOUBLE z = (DOUBLE)0x80000000;
LONG i,j;
DOUBLE x;
     randx = (randx * 1103515245)+12345;
     i = randx & 0x7FFFFFFF;
     x = ((DOUBLE)i)/z;
     x *= (L+H+1);
     j = x;
     RETURN (j-L);
}
```

I hope this information is useful, if there are any queries please contact me.

Regards,

Colin Smith

Manager, DSP Products.

NTT

NIPPON TELEGRAPH AND TELEPHONE CORPORATION
NTT
1-2356, TAKE, YOKOSUKA-SHI, KANAGAWA-KEN,
238-03 JAPAN

Dr. Barry G. Haskell
AT&T Bell Laboratories
Room 4C538
Holmdel, NJ

Subject : IDCT Specification
Date : November 7, 1988
Our Ref : SGXVL579
Number of Pages: 3 including this page

Dear Dr. Haskell,

Thank you for having accepted our visits to your lab the other day. Mr. Yamashita and I could enjoy the one week autumn stay in New Jersey.

Stimulated by the INMOS comment on the random number generator, my colleague Yutaka Suzuki tried their proposal. He found that our computer CONVEX has a problem with the number "0x80000000", because it produces a minus number. To cope with this problem, he tried two modifications; one keeping that number but adding "-" to one equation, the other keeping the original "0x7fffffff" but changing the third and second last equations in your program. Programs are listed in the attached two page paper. Random numbers generated are also listed in the attached paper.

I think the modification 2 can solve the problem 1/ Mr. Colin Smith pointed out.

Recent good news is that THOMSON (a major player according to Mr. Liou) can meet the CCITT specification with their latest design. Perhaps we may be able to finalize our specification in Florida.

Hoping this information be useful for you.

Best regards,

大久保栄

Sakae OKUBO
Visual Media Laboratory
NTT Human Interface Laboratories
1-2356 Take, Yokosuka-shi
238-03 Japan
Phone: +81 468 59 2814
Fax : +81 468 59 2829 or 2812
Telex: 03852480 YECL J

```c
#include        <stdio.h>
        main()
{
        int     i;
        long    k1, k2, k3, k4;
        long    new_rand(), new2_rand(), new3_rand(), new4_rand();

        printf("¥n/*");
        printf("¥n          No.     Haskell     INMOS     Modify     Modify");
        for( i = 0 ; i < 32 ; i++ )
        {
                k1 = new_rand(256, 255);
                k2 = new2_rand(256, 255);
                k3 = new3_rand(256, 255);
                k4 = new4_rand(256, 255);
                printf("¥n %10d%10d%10d%10d%10d", i, k1, k2, k3, k4);
        }
        printf("¥n*/");
}
/*------------------------------------------------------------------------*/
        long    new_rand(L, H)
        long    L, H;
{
        static long     randx = 1;
        static double   z = (double)0x7fffffff;

        long    i, j;
        double  x;

        randx = ( randx * 1103515245 ) + 12345;
        i = randx & 0x7fffffff;
        x = ( (double)i ) / z;
        x *= ( L + H );
        j = x + 0.5;
        return( j - L );
}
```

Haskell

```c
/*------------------------------------------------------------------------*/
        long    new2_rand(L, H)
        long    L, H;
{
        static long     randx = 1;
        static double   z = (double)0x80000000;

        long    i, j;
        double  x;

        randx = ( randx * 1103515245 ) + 12345;
        i = randx & 0x7fffffff;
        x = ( (double)i ) / z;
        x *= ( L + H + 1 );
        j = x;
        return( j - L );
}
```

Inmos

```
/*------------------------------------------------------------------------------*/
        long     new4_rand(L,H)
        long     L,H;
{
        static long      randx = 1;
        static double    z = (double)0x80000000;

        long     i,j;
        double   x;
                                                                  Mod 1
        randx = ( randx * 1103515245 ) + 12345;
        i = randx & 0x7fffffff;
        x = ( (double)i ) / z;
        x *= -( L + H + 1 );
        j = x;
        return( j - L );
}
/*------------------------------------------------------------------------------*/
        long     new3_rand(L,H)
        long     L,H;
{
        static long      randx = 1;
        static double    z = (double)0x7fffffff;

        long     i,j;
        double   x;                                               Mod 2

        randx = ( randx * 1103515245 ) + 12345;
        i = randx & 0x7fffffff;
        x = ( (double)i ) / z;
        x *= ( L + H + 1 );
        j = x;
        return( j - L );
}

/*
        No.     Haskell     INMOS     Modify 1     Modify 2
         0          7        -519          7            7
         1       -166        -345       -167         -167
         2        -98        -414        -98          -98
         3         17        -529         17           17
         4        228        -741        229          229
         5       -168        -343       -169         -169
         6        103        -615        103          103
         7       -140        -371       -141         -141
         8         -3        -509         -3           -3
         9       -192        -319       -193         -193
        10       -213        -298       -214         -214
        11        -57        -455        -57          -57
        12       -114        -397       -115         -115
        13        -68        -444        -68          -68
        14        247        -759        247          247
        15         18        -530         18           18
        16        135        -648        136          136
        17         74        -586         74           74
        18        136        -648        136          136
        19        143        -655        143          143
        20        165        -677        165          165
        21       -178        -333       -179         -179
        22         64        -576         64           64
        23        -95        -417        -95          -95
        24        -79        -433        -79          -79
        25        213        -725        213          213
        26         10        -522         10           10
        27        -51        -461        -51          -51
        28         54        -566         54           54
        29        145        -658        146          146
        30        220        -732        220          220
        31        189        -701        189          189
*/
```

```
                        /*L and H must be long, ie, 32 bits*/
long rand(L,H)
long         L,H;
{
    static  long     randx = 1;       /*long is 32 bits*/
    static double    z = (double)0x7fffffff;

    long    i,j;
    double x;                         /*double is 64 bits*/

    randx = (randx * 1103515245) + 12345;
    i = randx & 0x7ffffffe;                   /*keep 30 bits*/
    x = ( (double)i ) / z;                    /* range 0 to 0.99999... */
    x *= (L+H+1);                             /* range 0 to  < L+H+1 */
    j = x ;                                   /*truncate to integer*/
    return( j - L );                          /*range  -L  to  H  */
}
```

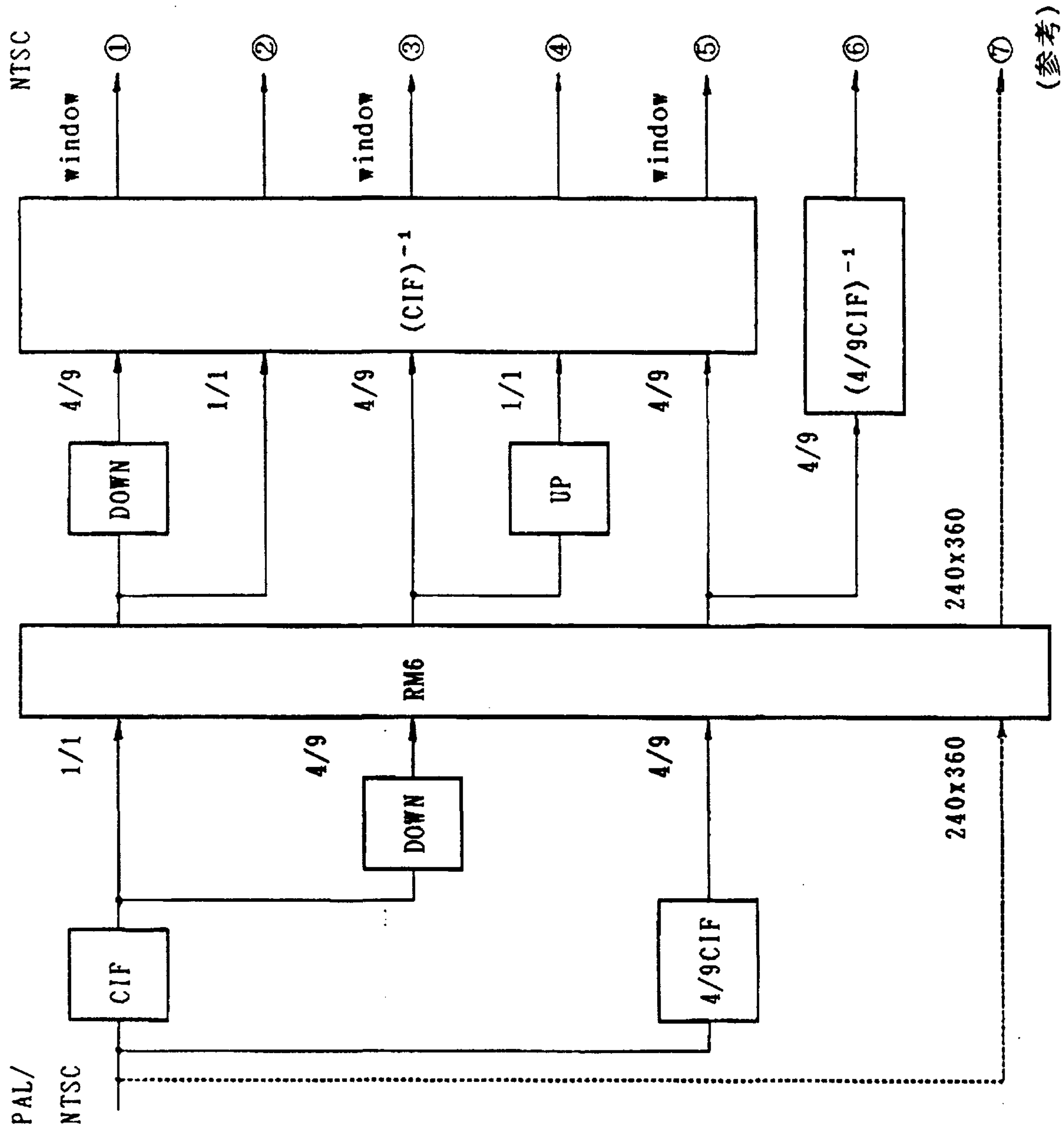
Colin:

Thanks for your response.

Our compiler cannot use your program since 0x80000000 is converted to a negative number. The above program works. What do you think?

Note that the least significant bits of randx are not very random anyway.

Barry


PAL/
NTSC
CIF
DOWN
4/9CIF
1/1
4/9
4/9
240x360
RM6
240x360
DOWN
UP
4/9
1/1
4/9
1/1
4/9
4/9
(CIF)$^{-1}$
(4/9CIF)$^{-1}$
window
window
window
NTSC
①
②
③
④
⑤
⑥
⑦
(参考)

To: B G Haskell, AT&T, Bell Labs

From: Colin Smith, Inmos, Bristol, UK

Date: 3rd November 1988.

Subject: Proposed CCITT Specification for Inverse DCT chips

Dear Barry,

The new program looks OK. We ran it through our simulator to see what difference the minor modifications made and we got marginally worse results but as expected the results were still within the proposed specification limits.

Regards,

Manager, DSP Products.

ANT Nachrichtentechnik

ANT Nachrichtentechnik GmbH · Postfach 1120 · D-7150 Backnang

Barry Haskell
AT&T Bell Labs - 4C538
Holmdel, NJ 07733
USA
Fax: +1 201 949 3697

Copy to:
Sakae Okubo
NTT Human Interface Labs

Fax: +81 468 59 2829

Ihre Zeichen/Nachricht vom	Unsere Zeichen/Bearbeiter	☎ (07191) 13-0 Durchwahl 13-	Backnang
	E324 Loh/eg - Dr.Lohscheller	2724	November 9, 1988

Dear Mr. Haskell,

examinating the modified specifications for Inverse DCT chips, we found the minor changes within the first six topics to be helpful and we can agree to it.

Concerning to the performance measures summarized in topic 7, we appreciate the intensified overall mean error requirement. But we strongly ask for reconsideration of the overall mean square error requirement. It can be shown, that the mean square error performance for any pixel do not vary so much. Therefore, the inconsistent requirements of the mean square error and the overall mean square error should be settled.

Kind regards,

MX/V

Börner

E3

Dr.Welzenbach

IEEE STANDARDS OFFICE
THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.
345 EAST 47TH STREET, NEW YORK, N.Y. 10017-2394, U.S.A. TELEX 237936

DIRECT NUMBER
(212) 705-7960

June 24, 1988

Ming-Ting Sun
Bellcore
331 Newman Springs Road
Red Bank, NJ 07701-7020

Subject: P1180 Standards Specification for Implementations of 8X8 Discrete Cosine Transform and Inverse Discrete Cosine Transform (DCT/IDCT)

Dear Ming-Ting Sun:

The subject project submittal has been received by th IEEE Standards Office. The assigned project number is P1180. This PAR will now be circulated to all liaison representatives to the Standards Board from Groups/Societies and Technical Committees for review and comment.

Final action on P1180 will be taken at the October 18, 1988 IEEE Standards Board meeting.

Sincerely,

Jane Post

Jane Post
Secretary, NesCom

Bell
Communications
Research
331 Newman Springs Road
Box 7020
Red Bank, New Jersey 07701-7020
(201) 758-2000

Rm. NVC 3X-301
Ext. (201) 758-2869

June 7, 1988

Mr. Paul Lange
NesCom
IEEE Standards Office
345 East 47th Street
New York, New York 10017

Dear Paul:

Enclosed is a PAR form for consideration by the NesCom, and approval by the IEEE Standards Boards as an IEEE standards project. If you have any questions, please call me at (201) 758-2869.

Thank you very much for your assistance.

Sincerely,

Ming T. Sun
Chair, CAS AdHoc Committee on Standards

Attachment
As Above

Copy (with Att) to
Members of CAS Ad Hoc Committee on Standards
M. L. Liou

STANDARDS PROJECT AUTHORIZATION (PAR)

When completing this PAR refer to instructions in the PAR Submitter's Guide

1. 6/01/88
Date of Request

Revised PAR [ ] Yes [x] No

12. Standards Board Assigned Project No.
Approved:
(For Standards Office Use Only)

2. [x] Standard [ ] Recommended Practice [ ] Guide
[x] New [ ] Revision of

3. Project Title: Standards Specification for Implementations of 8x8 Discrete Cosine Transform and Inverse Discrete Cosine Transform (DCT/IDCT)

4. Scope of Proposed Standard (use attachment sheet if necessary):
(See attached)

5. Purpose of Proposed Standard (use attachment sheet if necessary):
(See attached)

6. Sponsor
Technical Committee: Standards Committee Society: CAS

7. Proposed Coordination: / Method of Coordination:

Proposed Coordination	Method of Coordination
Other IEEE Societies	Circulation of Drafts
CCITT Special Group on Visual Telephony	Joint Membership
ISO	Circulation of Drafts
ANSI	Circulation of Drafts

8. Name of Group that will write the Standard:
IEEE/CCITT Special Group on Visual Telephony

9. Are you aware of any patent issues? [ ] Yes [X] No (If yes, attach a sheet with a complete description.)

Are you aware of any standards or projects with a similar scope? [ ] Yes [x] No (If yes, attach a sheet with a complete description.)

10. Person Delegated to Receive Communications and Conduct Liaison with Interested Bodies:

Name Ming-Ting Sun
Company Bellcore
Street Address 331 Newman Springs Road
City Red Bank, NJ 07701-7020
State
Zip Code
Telephone No. (201) 758-2869
Telefax No. (201) 758-0889
Telex No. 275318

11. Submitted by:

Name Ming-Ting Sun
Company Bellcore
Street Address 331 Newman Springs Road
City Red Bank
State NJ
Zip Code 07701-7020
Telephone No. (201) 758-2869
Telefax No. (201) 758-0889
Telex No. 275318

Note: Copies of PAR Submitter's Guide and the IEEE Standards Manual are available from the IEEE Standards Office

11/13/85 :lg

Project Title:

Standards Specification for Implementation of 8x8 Discrete Cosine Transform and Inverse Discrete Cosine Transform (DCT/IDCT)

Scope of Proposed Standard:

The proposed standard is a set of specifications for 8x8 DCT/IDCT implementations to be used in video codecs and image data compression systems. The specification ensures the compatibility between different implementations of the DCT/IDCT. It also ensures that the picture quality will not be degraded by the limited accuracy of the various DCT/IDCT implementations.

Purpose of Proposed Standard:

The 8x8 DCT/IDCT is becoming a very popular compression technique for many video codecs and image data compression systems. However, different implementations of the DCT/IDCT would yield various degrees of accuracy. When different implementations of the DCT/IDCT are used in the coder and decoder, the accuracy mismatching may cause severe picture-quality degradation. In order to ensure a high-quality system, a standard specification for the 8x8 DCT/IDCT is necessary. The purpose of this proposed standard is to provide such a specification so that all implementations of 8x8 DCT/IDCT will have sufficient accuracy, eliminate the hardware compatibility problem and, in turn, reduce the cost of implementation as a result of competition.

CCITT Specialists Group on Visual Telephony

Proposed Specification for Inverse DCT Chips

1. Generate random integer pixel data values in the range -L to +H according to the attached random number generator (C version). Arrange into 8x8 blocks by allocating each set of consecutive 8 numbers in a row. Data sets of 10000 blocks each should be generated for (L=256, H=255), (L=H=5) and L=H=300).

2. For each 8x8 block, perform a separable, orthonormal, matrix multiply, Forward Discrete Cosine Transform (FDCT) using at least 64-bit floating point accuracy.

3. For each block, round the 64 resulting transformed coefficients to the nearest integer values. Then clip them to the range -2048 to +2047. This is the 12-bit input data to the inverse transform.

4. For each 8x8 block of 12-bit data produced by step 3, perform a separable, orthonormal, matrix multiply, Inverse Discrete Cosine (IDCT) using at least 64-bit floating point accuracy. Round the resulting pixels to the nearest integer, and clip to the range -256 to +255. These blocks of 8x8 pixels are the "reference" IDCT output data.

5. For each 8x8 block of 12-bit data produced by step 3, use the proposed IDCT chip or an exact-bit simulation thereof to perform an Inverse Discrete Cosine Transform. Clip the output to the range -256 to +255. These blocks of 8x8 pixels are the "test" IDCT output data.

6. For each of the 64 IDCT output pixels, and for each of the 10,000 block data sets generated above, measure the peak, mean and mean square error between the "reference" and "test" data.

7. For any pixel, the peak error should not exceed 1 in magnitude.
For any pixel, the mean square error should not exceed 0.06.
Overall, the mean square error should not exceed 0.02.
For any pixel, the mean error should not exceed 0.015 in magnitude.
Overall, the mean error should not exceed 0.0015 in magnitude.

8. All-zeros in must produce all-zeros out.

9. Rerun the measurements using exactly the same data values of step 1, but change the sign on each pixel.

```
                    /*L and H must be long, ie, 32 bits*/
long rand(L,H)
long      L,H;
{
   static long   randx = 1;    /*long is 32 bits*/
   static double   z = (double)0x7fffffff;

   long   i,j;
   double x;                  /*double is 64 bits*/

   randx = (randx * 1103515245) + 12345;
   i = randx & 0x7ffffffe;                  /*keep 30 bits*/
   x = ( (double)i ) / z;                   /* range 0 to 0.99999... */
   x *= (L+H+1);                            /* range 0 to  < L+H+1 */
   j = x ;                                  /*truncate to integer*/
   return( j - L );               /*range  -L   to   H  */
}
```