

СООБЩЕНИЯ
ОБЪЕДИНЕННОГО
ИНСТИТУТА
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ
ДУБНА

4623/2-80

22/9-80
P11-80-417

И.Байла, Г.А.Ососков

ОБ ИСПОЛЬЗОВАНИИ
ПРЕОБРАЗОВАНИЯ УОЛША-АДАМАРА
ДЛЯ ВЫЧИСЛЕНИЯ БЫСТРОГО
СЛЭНТ-ПРЕОБРАЗОВАНИЯ

1980

Байла И., Ососков Г.А.

P11-80-417

Об использовании преобразования Уолша-Адамара для вычисления быстрого слэнт-преобразования

Рассматривается функциональная связь между быстрым слэнт-преобразованием /БСП/ и быстрым преобразованием Уолша-Адамара /БПУА/ размерности $N = 2^n$, $n = \overline{2,6}$ для разработки алгоритмов вычисления коэффициентов перехода между ними. Приводится текст и описание программы HASLAN, вычисляющей матрицы перехода в случае прямого и обратного преобразования. Излагаются результаты тестовых вычислений по программе TEST. Расчеты показали выгоду применения БПУА для аппаратной реализации прямого БСП и нецелесообразность этого метода для обратного БСП.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна 1980

Bajla I., Ososkov G.A.

P11-80-417

On Utilization of Walsh-Hadamard Transform for Fast Slant Transform Computation

The functional relation between the Fast Slant Transform

ВВЕДЕНИЕ

Одной из целей построения быстрых алгоритмов дискретных ортогональных преобразований /ДОП/, предназначенных для обработки цифровых сигналов, является использование этих алгоритмов в некоторой он-лайн системе сжатия данных. Благодаря своей простоте и быстродействию особое место среди существующих аппаратных реализаций быстрых алгоритмов ДОП принадлежит алгоритму преобразования Уолша-Адамара^{1-3/} /ПУА/.

В работе^{4/} были рассмотрены вопросы построения быстрых алгоритмов и программ прямого и обратного слэнт-преобразования /СП/ произвольной размерности $N = 2^n$, которое успешно используется при цифровой обработке двумерных изображений^{1,2,5-7/}. Предложенные в этой работе программы служат непосредственной основой для аппаратной реализации СП. Однако, поскольку между ПУА и СП можно при помощи некоторых матриц перехода установить функциональную связь, возможна реализация также и другого подхода к созданию аппаратной системы быстрого СП, основанного на использовании вышеупомянутых преимуществ быстрого ПУА.

Если располагать аппаратной версией быстрого ПУА, построенной, например, на основе параллельного алгоритма, предложенного в работе^{8/}, или алгоритма, приведенного в работе^{9/}, то для получения слэнт-спектра любого исходного вектора данных достаточно:

- 1/ заранее вычислить /при помощи некоторой офф-лайн программы/ матрицу перехода между ПУА и СП и
- 2/ реализовать аппаратно только блоки умножения и сложения преобразованных данных /спектра Уолша-Адамара/ на коэффициенты перехода.

В предлагаемой работе выводятся соотношения между матрицами H_N для ПУА и S_N для СП / $N = 2^n$, $n = \overline{2,6}$ / и на их основе построена программа HASLAN, вычисляющая коэффициенты перехода от ПУА к СП для прямого и обратного случаев.

ПОСТАНОВКА ЗАДАЧИ

А. Прямые преобразования

Пусть X есть вектор-столбец входных данных, подлежащих преобразованию. Пусть через S и S^{-1} обозначены соответствующие матрицы прямого и обратного СП, а через H обозначена мат-

рица ПУА. Рассматриваемые преобразования запишем следующим образом:

$$X \rightarrow SX = Y_S, \quad /1/$$

$$X \rightarrow \frac{1}{N}HX = Y_H, \quad /2/$$

где Y_S - слэнт-спектр, а Y_H - спектр Уолша-Адамара вектора X . Матрицу перехода ПУА \rightarrow СП определим как некоторую матрицу A , для которой справедливо

$$Y_S = A \cdot Y_H. \quad /3/$$

Если в /3/ подставить /2/, то

$$Y_S = \frac{1}{N}A \cdot H \cdot X, \quad /4/$$

и следовательно, на основе /1/

$$S = \frac{1}{N}A \cdot H. \quad /5/$$

Поскольку элементы $N \times N$ матриц S и H в /5/ можно вычислить на основе их определений /2/, равенство /5/ можно рассматривать как матричное уравнение для неизвестной матрицы A , i -той строке которой $[a_{i1}, a_{i2}, \dots, a_{iN}]$ ($i = 1, N$) соответствует система линейных уравнений:

$$\begin{aligned} a_{i1}h_{11} + a_{i2}h_{21} + \dots + a_{iN}h_{N1} &= N \cdot s_{i1}, \\ a_{i1}h_{12} + a_{i2}h_{22} + \dots + a_{iN}h_{N2} &= N \cdot s_{i2}, \\ &\vdots \\ a_{i1}h_{1N} + a_{i2}h_{2N} + \dots + a_{iN}h_{NN} &= N \cdot s_{iN}. \end{aligned} \quad /6/$$

Следовательно, решив все N системы /6/, получим элементы матрицы A_N .

Располагая матрицей перехода A_N и значениями компонент вектора $Y_H = [y_{h1}, y_{h2}, \dots, y_{hN}]'$ - образа исходного вектора X при преобразовании Уолша-Адамара, слэнт-образ $Y_S = [y_{s1}, y_{s2}, \dots, y_{sN}]'$ вектора X получаем следующим образом:

$$y_{s1} = a_{11}y_{h1} + a_{12}y_{h2} + \dots + a_{1N}y_{hN},$$

$$y_{s2} = a_{21}y_{h1} + a_{22}y_{h2} + \dots + a_{2N}y_{hN},$$

\vdots

$$y_{sN} = a_{N1}y_{h1} + a_{N2}y_{h2} + \dots + a_{NN}y_{hN}$$

/7/

/через $[]'$ будем обозначать вектор-столбец, соответствующий вектор-строке $[]$./

Эти формулы служат непосредственной основой для создания блоков, дополняющих аппаратуру быстрого преобразования Уолша-Адамара для получения на выходе компонент слэнт-спектра Y_S . Следует, однако, отметить, что с точки зрения быстродействия такой он-лайн версии слэнт-преобразования очень важной является структура матрицы перехода A_N .

Формулы /7/, выведенные в /7/ для частного случая $N = 4, 8$, показывают, что матрицы A_4, A_8 состоят в основном из нулей, и ответить на вопрос, как ведут себя матрицы A_N для $N > 8$, можно только, решив матричное уравнение /5/ для соответствующих значений N . Сказанное относится в равной степени и к матрице B_N для перехода между обратными ПУА и СП, тем более, что, насколько известно авторам, коэффициенты матрицы обратного перехода еще не исследовались.

Б. Обратные преобразования

Для матрицы H преобразования Уолша-Адамара имеют место /2/ следующие соотношения:

$$H'_N = H_N, \quad /8/$$

$$H_N H'_N = H'_N H_N = N \cdot I_N,$$

где I_N - единичная $N \times N$ матрица, и

$$H^{-1} = \frac{1}{N}H. \quad /9/$$

Исходя из формулы $Y_H = \frac{1}{N}H \cdot X$, определяющей прямое ПУА, с учетом /8/ и /9/ для обратного ПУА получаем $X = H \cdot Y_H$, где Y_H - спектр Уолша-Адамара.

Вычисление элементов матрицы B_N перехода от обратного ПУА к обратному СП желательно провести при помощи уже вычисленной матрицы A_N , избегая тем самым решения отдельного матричного уравнения.

Обозначая через X_H вектор, полученный в результате обратного ПУА вектора Y_S , т.е. $[x_{h1}, x_{h2}, \dots, x_{hN}] = X_H = H \cdot Y_S$ имеем для обратного СП вектора Y_S

$$X = B_N X_H = B_N H Y_S \quad (X = [x_1, x_2, \dots, x_N]')$$
 /10/

В то же время для обратного СП справедливо

$$X = S' Y_S,$$
 /11/

поскольку $S' = S^{-1}$.

Сравнение /10/ и /11/ дает соотношение

$$S' = B_N H.$$
 /12/

С другой стороны, транспонированием матрицы S в /5/ получаем $S' = \frac{1}{N} H' A'_N = \frac{1}{N} H A'_N$, откуда с учетом /12/ имеем окончательную формулу:

$$B_N = \frac{1}{N^2} H A'_N H$$
 /13/

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

Назначение

Программа HASLAN, построенная на основе формул /5/ и /13/, а также формул, при помощи которых определены слэнт-преобразование и преобразование Уолша-Адамара^{/2/}, предназначена главным образом для вычисления коэффициентов перехода между этими преобразованиями для $N = 2^n$, $n = 2, 6$.

Программа HASLAN реализована на языке FORTRAN IV, снабжена развернутой программной печатью результатов и опробована на ЭВМ CDC-6500.

Инструкция по использованию

Кроме быстрого алгоритма вычисления элементов слэнт-матрицы S_N , взятого из работы^{/4/}, и алгоритма вычисления матрицы H_N для ПУА, программа HASLAN включает в себя и алгоритмы, реализующие ПУА исходного вектора /массив X / и обратное ПУА некоторого вектора-образа /массив Y /. Результаты этих преобразований - выходной спектр или первоначальный вектор засылаются в те же массивы X и Y .

Вычисляемые коэффициенты перехода помещаются в массивы SC /матрица A_N / и SCIN. /матрица B_N /, параметрам N и n соответствуют в программе переменные MM и M.

Выбор соответствующих значений параметров печати: INM, IPR1, IPR2, IPR3, пояснение к которым дано в комментариях программы, позволяет печатать коэффициенты перехода для промежуточных значений N /подпрограмма TPRINT /.

Для решения матричного уравнения /5/ программа HASLAN использует библиотечную подпрограмму LINEQ1.

Обращение

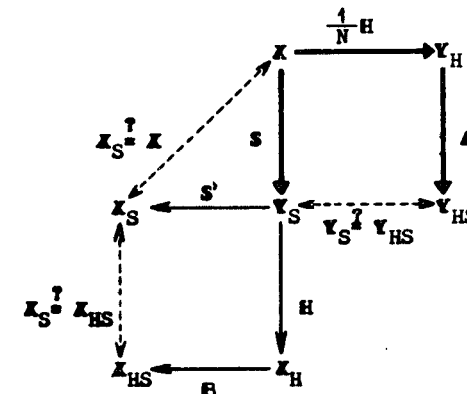
CALL HASLAN (X, Y, SC, SCIN, M, MM, IT, INM, INV, IPR1, IPR2, IPR3).

Ресурсы и быстроедействие

Общий объем оперативной памяти ЭВМ CDC-6500, необходимый для счета по программе HASLAN, составляет 41175₈ слов. Отметим, что при этом основная часть памяти отводится массивам SS, H /8192 слов/ и массиву P /8192 слов/, необходимому для вызова подпрограммы LINEQ1. Поэтому, если это необходимо для $N < 64$, пользователь может резко сократить упомянутые ресурсы при помощи соответствующей небольшой модификации размеров этих массивов и модификации оператора вызова LINEQ1. Быстроедействие программы HASLAN на ЭВМ CDC-6500, вычисленное в тестовых примерах для $N = 64$, составляет 23 с.

ТЕСТОВОЕ ОБЕСПЕЧЕНИЕ И РЕЗУЛЬТАТЫ ВЫЧИСЛЕНИЙ

Для проверки работы программы HASLAN была специально разработана программа TEST, которая после вычисления модельных векторов данных ($N = 64$) реализует следующую схему преобразований:




```

SUBROUTINE HASLANIX,Y,SC,SCIN,M,MH,IT,INN,INV,IPR1,IPR2,IPR3)
*****
THIS ROUTINE CALCULATES THE COEFFICIENT MATRIX FOR MAPPING*
THE WALSH-HADAMARD TRANSFORM VECTOR TO SLANT TRANSFORM ONE*
X- INPUT VECTOR, ON THE OUTPUT MAY BE MHT-VECTOR
Y- INPUT SLANT TRANSFORM VECTOR
ON THE OUTPUT MAY BE THE INVERSE MHT-VECTOR
SC- FOUNDED COEFFICIENT MATRIX OF RANGE 64 X 64
SCIN- FOUNDED INVERSE COEFFICIENT MATRIX OF RANGE 64 X 64
M- EXPONENT, MM=2**M
MH- NEEDED MATRIX RANGE
IT- MHT TEST PARAMETER
IF IT.EQ.0 NO CALCULATION
IF IT.NE.0 THE MHT-TRANSFORM OF VECTOR X IS CALCULATED*
INN- INTERMEDIATE PRINTING PARAMETER
IF INN.EQ.0 INTERMEDIATE PRINTINGS OMITTED
IF INN.NE.0 INTERMEDIATE PRINTINGS ACCEPTED
INV- PARAMETER OF THE INVERSE TRANSFORMATION
IF INV.EQ.0 NO INVERSE MATRIX CALCULATION
IF INV.NE.0 INVERSE COEFFICIENT MATRIX IS CALCULATED*
IPR1,IPR2,IPR3- PRINTINGS PARAMETER.
*****
IPR1  IPR2  IPR3
1      1      1      PRINTED- 1) HEADLINE 2) TRANSFORM
1      1      0      PRINTED- 1) HEADLINE,2) COEF MATRIX SC
1      0      0      PRINTED- 1) HEADLINE,2) TRANSFORM
0      0      0      PRINTED- ONLY HEADLINE
0      0      1      NO PRINT
0      1      1      PRINTED- ONLY COEF MATRIX SC
1      0      1      PRINTED- 1) TRANSFORM MATRICES
1      1      0      PRINTED- 2) COEF MATRIX SC
1      1      0      PRINTED- 1) HEADLINE, 2) SC MATRIX
1      1      0      PRINTED- TRANSFORM MATRICES
SS- SLANT MATRIX OF RANGE 64 X 64
H- WALSH-HADAMARD MATRIX OF RANGE 64 X 64
*****
DIMENSION SC(64,64),SCIN(64,64),SS(64,64),H(64,64)
X(64),Y(64),Z(64),ZL(64),HL(64),P(64,128),PS(64,2),INDEX(64)
SQ=1./SQRT(2.)
A=1.
PRINT 10
FORMAT(1H1)
PRINT 20,MH
20  FORMAT(1/2X,*,TOTAL TRANSFORM RANGE MM = *,I3)
SS(1,1)=SQ          SS(1,2)=SQ
SS(2,1)=SQ          SS(2,2)=-SQ
H(1,1)=1.           H(1,2)=1.
H(2,1)=1.           H(2,2)=-1.
DO 30 N=1,M
MM=2**N
IF(NN.EQ.2) GO TO 40
NN=MM/2            N1=MM+1
N2=N1+1           N3=N2+1
S1=1./SQRT(1.+A**2)  S2=1./SQRT(1.+B**2)  S A=2.*B*A
*****
C *****
C *****
CALCULATE THE (MN X MN) SLANT MATRIX *
*****
DO 50 K=1,NH
PS(1,1)=SQ*SS(1,K)  $ PS(1,2)=PS(1,1)
PS(2,1)=SQ*(A*SS(1,K)+B*SS(2,K))
PS(2,2)=SQ*(A*SS(1,K)+B*SS(2,K))
IF(NN.EQ.3) GO TO 60
DO 70 J=3,NH
PS(J,1)=SQ*SS(J,K)  $ PS(J,2)=PS(J,1)
CONTINUE
CONTINUE
PS(N1,1)=SQ*SS(2,K)
PS(N1,2)=PS(N1,1)
PS(N2,1)=SQ*(B*SS(1,K)+A*SS(2,K))
PS(N2,2)=SQ*(B*SS(1,K)+A*SS(2,K))
IF(NN.LT.3) GO TO 80
DO 90 J=N3,NH
NN=J-NH
PS(J,1)=PS(JN,1)
PS(J,2)=PS(J,1)
CONTINUE
70  CONTINUE
80  CONTINUE

```

```

DO 100 J=1,NH
SS(J,K)=PS(1,1)
SS(J,K+NH)=PS(J,2)
CONTINUE
100 CONTINUE
C *****
C *****
CALCULATE THE (NN X NN) - WALSH-HADAMARD MATRIX *
*****
DO 110 J=1,NH
DO 110 K=N1,NH
L=K-NH
H(J,K)=H(J,L)  $ H(K,J)=H(L,J)
CONTINUE
110 CONTINUE
DO 120 J=N1,NH
DO 120 K=N1,NH
JJ=J-NH      KK=K-NH
H(J,K)=-H(JJ,KK)
CONTINUE
120 CONTINUE
IF(N.NE.M) GO TO 130
IF(IT.EQ.0) GO TO 140
*****
C *****
C *****
CALCULATE THE W-H TRANSFORM OF VECTOR X *
AND INVERSE W-H TRANSFORM OF VECTOR Y *
*****
DO 125 IK=1,2
DO 150 I=1,NH
SUM=0
DO 160 J=1,NH
IF(IK.EQ.1) SUM=SUM+H(I,IJ)*X(IJ)
IF(IK.EQ.2) SUM=SUM+H(I,IJ)*Y(IJ)
CONTINUE
160 IF(IK.EQ.1) Z(I)=SUM/FLOAT(NH)
IF(IK.EQ.2) Z(I)=SUM
CONTINUE
150 DO 170 I=1,NH
IF(IK.EQ.1) X(I)=Z(I)
IF(IK.EQ.2) Y(I)=Z(I)
CONTINUE
170 CONTINUE
125 CONTINUE
140 CONTINUE
130 CONTINUE
40  CONTINUE
IF(IPR1.EQ.0) GO TO 180
IF(INN.EQ.0.AND.M.LT.M) GO TO 180
PRINT 190,NH
190  FORMAT(1/2X,*,SLANT TRANSFORM RANGE NN = *,I2)
PRINT 200,A,B
200  FORMAT(1/2X,*,COEFFICIENTS A = *, F8.6,*, B = *, F8.6//)
180  CONTINUE
IF(NN.LE.8) NT=NN
IF(NN.GT.8) NT=8
NTR=NN/NT
IF(IPR2.EQ.0) GO TO 210
IF(INN.EQ.0.AND.M.LT.M) GO TO 210
PRINT 220
220  FORMAT(1/2X,*,SLANT TRANSFORM MATRIX *//)
CALL TPRINT(SS,NN,NT,NTR)
PRINT 230
230  FORMAT(1/2X,*,WALSH-HADAMARD TRANSFORM MATRIX *//)
CALL TPRINT(H,NN,NT,NTR)
210 CONTINUE
C *****
C *****
CALCULATE THE W-H COEFFICIENT MATRIX SC *
*****
NH=2*MM  $ NL=NN+1
DO 240 I=1,NH
DO 240 J=1,NH
P(I,J)=H(J,I)/FLOAT(NH)
CONTINUE
240 CONTINUE
DO 250 I=1,NH
DO 250 J=NL,NH
JJ=J-NH
P(I,J)=SS(JJ,I)
CONTINUE
250 CONTINUE
CALL LINE01(P,64,NN,64,NN,INDEX,NER,DET)
DO 260 I=1,NH
DO 260 J=1,NH
SC(I,J)=P(J,I)
260 CONTINUE

```

```

IF(IPR3.EQ.0) GO TO 275
IF(INH.EQ.0.AND.N.LT.M) GO TO 275
PRINT 280
FORMAT (/2X, * HADAMARD-SLANT COEFFICIENT MATRIX A *//)
CALL TPRINT(SC,NN,NT,NTR)
CONTINUE
275 IF(INV.EQ.0) GO TO 255
DO 265 I=1,NN
DO 265 J=1,NN
SCIN(I,J)=SC(J,I)
CONTINUE
265 DO 310 K=1,NN
DO 320 I=1,NN
SUM=0
DO 330 J=1,NN
SUM=SUM+H(I,J)*SCIN(J,K)
CONTINUE
330 HL(I)=SUM
CONTINUE
320 DO 340 I=1,NN
SCIN(I,K)=HL(I)
CONTINUE
310 DO 350 K=1,NN
DO 360 I=1,NN
SUM=0
DO 370 J=1,NN
SUM=SUM+SCIN(I,J)*H(J,K)
CONTINUE
360 HL(I)=SUM/FLOAT(NN**2)
CONTINUE
DO 380 I=1,NN
DO 285 J=1,NN
IF(N.LT.M) SC(I,K)=HL(I)
IF(N.EQ.M) H(I,K)=HL(I)
CONTINUE
380 CONTINUE
DO 285 I=1,NN
DO 285 J=1,NN
IF(N.LT.M) SCIN(I,J)=SC(I,J)
IF(N.EQ.M) SCIN(I,J)=H(I,J)
CONTINUE
285 CONTINUE
IF(INV.EQ.0) GO TO 270
IF(IPR3.EQ.0) GO TO 270
IF(INH.EQ.0.AND.N.LT.M) GO TO 270
PRINT 300
FORMAT (/2X, * INVERSE HADAMARD-SLANT COEFFICIENT MATRIX B *//)
CALL TPRINT(SCIN,NN,NT,NTR)
CONTINUE
270 CONTINUE
30 RETURN
END

SUBROUTINE TPRINT(SC,M,NT,NTR)
DIMENSION SC(64,64)
DO 10 K=1,NTR
KL=K*NT
KP=KL-NT+1
DO 20 J=1,M
PRINT 30 SC(I,J),J=KP,KL
FORNAT (/2X,8(2X,F12.8))
CONTINUE
IF(NTR.GT.1.AND.K.LT.NTR) PRINT 40
FORNAT (/2X,5H* * *//)
CONTINUE
10 RETURN
END

```

Тест I - проверяет работу программ FSLANT и ISLANT^{/4/} для прямого и обратного СП соответственно.

$$(X_S = X)$$

Тест II - проверяет правильность полученных коэффициентов перехода - элементов матрицы A_N , причем в качестве модели он-лайн версии быстрого преобразования Уолша-Адамара вектора X используется программа WHT, предложенная в работе^{/2/}.

Тест III - проверяет правильность вычисленных коэффициентов перехода для обратного случая /элементы матрицы $B_N/$, причем для моделирования он-лайн версии обратного быстрого ПУА использована опять-таки программа WHT.

Все эти тесты дали верные результаты. В качестве примера приведем формулы перехода для прямого и обратного случаев, полученные в результате вычислений по программе HASLAN ($N=2,4,8,16$).

1. ПУА → СП

$$N = 2$$

$$y_{s1} = 1.41421356 \cdot y_{h1},$$

$$y_{s2} = 1.41421356 \cdot y_{h2}.$$

$$N = 4^*$$

$$y_{s1} = 2.00000000 \cdot y_{h1},$$

$$y_{s2} = 0.89442719 \cdot y_{h2} + 1.78885438 \cdot y_{h3},$$

$$y_{s3} = 2.00000000 \cdot y_{h4},$$

$$y_{s4} = 1.78885438 \cdot y_{h2} - 0.89442719 \cdot y_{h3}.$$

$$N = 8^*$$

$$y_{s1} = 2.82842712 \cdot y_{h1},$$

$$y_{s2} = 0.61721340 \cdot y_{h2} + 1.23442680 \cdot y_{h3} + 2.46885360 \cdot y_{h5},$$

$$y_{s3} = 2.82842712 \cdot y_{h4},$$

$$y_{s4} = 2.52982213 \cdot y_{h2} - 1.26491106 \cdot y_{h3},$$

$$y_{s5} = 1.26491106 \cdot y_{h6} + 2.52982213 \cdot y_{h7},$$

$$y_{s6} = 1.10410489 \cdot y_{h2} + 2.20820979 \cdot y_{h3} - 1.38013112 \cdot y_{h5},$$

$$y_{s7} = 2.82842712 \cdot y_{h8},$$

$$y_{s8} = 2.52982213 \cdot y_{h6} - 1.26491106 \cdot y_{h7}.$$

*Отметим, что приведенные здесь значения коэффициентов, проверенные по всем тестам, отличаются от приведенных в работе^{/10/}.

$$N = 16$$

$$\begin{aligned}
 y_{s1} &= 4.00000000 \cdot y_{h1} , \\
 y_{s2} &= 0.43386092 \cdot y_{h2} + 0.86772183 \cdot y_{h3} + 1.73544366 \cdot y_{h5} + 3.47088733 \cdot y_{h9} , \\
 y_{s3} &= 4.00000000 \cdot y_{h4} , \\
 y_{s4} &= 3.57770876 \cdot y_{h2} - 1.78885438 \cdot y_{h3} , \\
 y_{s5} &= 1.78885438 \cdot y_{h6} + 3.57770876 \cdot y_{h7} , \\
 y_{s6} &= 1.56144012 \cdot y_{h2} + 3.12288023 \cdot y_{h3} - 1.95180015 \cdot y_{h5} , \\
 y_{s7} &= 4.00000000 \cdot y_{h8} , \\
 y_{s8} &= 3.57770876 \cdot y_{h6} - 1.78885438 \cdot y_{h7} , \\
 y_{s9} &= 0.87287156 \cdot y_{h10} + 1.74574312 \cdot y_{h11} + 3.49148624 \cdot y_{h13} , \\
 y_{s10} &= 0.75740971 \cdot y_{h2} + 1.51481942 \cdot y_{h3} + 3.02963884 \cdot y_{h5} - 1.98820049 \cdot y_{h9} , \\
 y_{s11} &= 4.00000000 \cdot y_{h12} , \\
 y_{s12} &= 3.57770876 \cdot y_{h10} - 1.78885438 \cdot y_{h11} , \\
 y_{s13} &= 1.78885438 \cdot y_{h14} + 3.57770876 \cdot y_{h15} , \\
 y_{s14} &= 1.56144012 \cdot y_{h10} + 3.12288023 \cdot y_{h11} - 1.95180015 \cdot y_{h13} , \\
 y_{s15} &= 4.00000000 \cdot y_{h16} , \\
 y_{s16} &= 3.57770876 \cdot y_{h14} - 1.78885438 \cdot y_{h15} .
 \end{aligned}$$

2. Инверсное ПУА → Инверсное СП

$$N = 2$$

$$x_1 = 0.70710678 \cdot x_{h1} ,$$

$$x_2 = 0.70710678 \cdot x_{h2} .$$

$$N = 4$$

$$x_1 = 0.47360680 \cdot x_{h1} + 0.02639320 \cdot x_{h2} + 0.11180340 \cdot x_{h3} - 0.11180340 \cdot x_{h4} ,$$

$$x_2 = -0.11180340 \cdot x_{h1} + 0.11180340 \cdot x_{h2} + 0.47360680 \cdot x_{h3} + 0.02639320 \cdot x_{h4} ,$$

$$x_3 = 0.11180340 \cdot x_{h1} - 0.11180340 \cdot x_{h2} + 0.02639320 \cdot x_{h3} + 0.47360680 \cdot x_{h4} ,$$

$$x_4 = 0.02639320 \cdot x_{h1} + 0.47360680 \cdot x_{h2} - 0.11180340 \cdot x_{h3} + 0.11180340 \cdot x_{h4} .$$

Проведенные для всех $n = 2, 6$ вычисления показывают, что, хотя все соответствующие матрицы A_N содержат в основном нули, матрицы B_N являются полными матрицами. Это означает, что умножение на матрицу перехода B_N не будет окупаться выигрышем в простоте и быстродействии, полученным за счет перехода от СП к ПУА.

Числа дополнительных арифметических операций / a_N - суммирование, m_N - умножение/, которые необходимо реализовать для перехода в прямом случае, приведены в таблице.

ЗАКЛЮЧЕНИЕ

Программа HASLAN позволяет получить коэффициенты перехода между преобразованием Уолша-Адамара и слэнт-преобразованием, необходимые для расширения некоторой существующей аппаратной реализации преобразования Уолша-Адамара до спецпроцессора, осуществляющего ПУА и СП. Кроме того, HASLAN может служить как

N	a_N	m_N
2	0	2
4	2	6
8	7	15
16	18	34
32	41	73
64	88	152

тестовое обеспечение для проверки других алгоритмов ПУА и СП.

Проведенные по программе HASLAN вычисления показали, что методику аппаратной реализации СП при помощи соответствующей системы ПУА и регистров с коэффициентами перехода можно рекомендовать только в случае прямого слэнт-преобразования. Для большинства случаев практического применения /например, к сжатию данных/ этого достаточно, так как обратное СП выполняется уже в режиме офф-лайн. Но в случае адаптивного он-лайн сжатия ^{4/}, когда с помощью немедленного он-лайн обращения урезанного спектра и быстрого сравнения реконструкции с оригиналом можно выбрать локально-оптимальную отсечку спектра, для реализации обратного слэнт-преобразования целесообразнее использовать алгоритм и программу, предложенную в работе ^{4/}.

ЛИТЕРАТУРА

1. Ярославский Л.П. Введение в цифровую обработку изображений. "Советское радио", М., 1979.
2. Ahmed N., Rao K.R. Orthogonal Transform for Digital Signal Processing. Springer-Verlag, Berlin-Heidelberg-New York, 1975.

3. Elliott A.R., Shum Y.Y. In: Proceedings of Applications of Walsh Functions Symposium and Workshop /Ed. R.W.Zeek et al./. Washington, March 1972.
4. Байла И., Ососков Г.А. ОИЯИ, P10-80-237, Дубна, 1980.
5. Pratt W.K. Digital Image Processing. A Wiley-Interscience Publication. John Wiley and Sons, New York-Chichester-Brisbane-Toronto, 1978.
6. Pratt W.K., Wen-Hsiung Chen, Welch L.R. IEEE Transactions on Communications, vol. COM-22, No 8, Aug. 1974.
7. Enomoto H., Shibata K. IEEE Transactions on Electromagnetic Compability, vol. EMC-13, pp.11-17, Aug. 1971.
8. Elliott A.R., Shum Y.Y. In: Proceedings of Applications of Walsh Functions Symposium and Workshop /Ed. R.W.Zeek et al./. Washington, March 1972.
9. Kunt M. IEEE Transactions on Computers, vol. C-24, No 11, Nov. 1975.
10. Enomoto H., Shibata K. In: Proceedings of Applications of Walsh Functions Symposium and Workshop /Ed. R.W.Zeek et al./. Washington, March 1972.

Рукопись поступила в издательский отдел
17 июня 1980 года.