ORAS : opérateur rapide d'analyse spectrale

P. Gille

CENTRE NATIONAL D'ETUDES
DES TELECOMMUNICATIONS

CENTRE NATIONAL DE LA
RECHERCHE SCIENTIFIQUE

CENTRE DE
RECHERCHES
EN PHYSIQUE DE
L'ENVIRONNEMENT
TERRESTRE
ET PLANETAIRE

CRPE

NOTE TECHNIQUE
CRPE /106

ORAS : Opérateur rapide d'analyse spectrale

Par
P. GILLE

CENTRE NATIONAL D'ETUDES
DES TELECOMMUNICATIONS

CENTRE NATIONAL DE
LA RECHERCHE SCIENTIFIQUE

RP 182 (49)

CENTRE DE RECHERCHE EN PHYSIQUE DE
L'ENVIRONNEMENT TERRESTRE ET PLANETAIRE

NOTE TECHNIQUE CRPE/106

ORAS : OPERATEUR RAPIDE D'ANALYSE SPECTRALE

par

P. GILLE

C.R.P.E./P.C.E.

45045 - ORLEANS CEDEX

(Août 1981)

Le Directeur du C.R.P.E.

J. HIEBLOT

ORAS : OPERATEUR RAPIDE D'ANALYSE SPECTRALE

RESUME

L'interprétation des mesures géophysiques peut être réalisée avec efficacité au moyen d'un mini-ordinateur conversationnel associé à un périphérique spécialisé dans les opérations de traitement de signal : fonctions scalaires ou vectorielles, réelles ou complexes, calcul de spectres directs et croisés sur un nombre variable de signaux numériques.

L'ORAS a été notamment conçu pour doter un calculateur HP 1000 F d'une capacité de calcul rapide des matrices spectrales relatives aux mesures d'ondes TBF à 9 composantes. Par exemple, le calcul d'une transformée de Fourier complexe avec accumulation sur 1024 échantillons s'effectue en 20 millisecondes.

Le logiciel de l'ordinateur assure la gestion des opérations et des transferts de données qui s'effectuent au rythme maximum de 1 million de mots par seconde. L'utilisateur a la possibilité de définir son application de façon interactive sous la forme d'un macroprogramme.

L'ORAS pourra être utilisé à différentes fins, telles que l'étude des bruits électromagnétiques et il pourra servir de prototype à un dispositif fonctionnant en temps réel.

Ce texte a été présenté à la XXème Assemblée Générale de l'U.R.S.I. à Washington le 11 Août 1981, au cours de la session C2 ("Telecommunications and digital signal processing"), sous le titre : "Fast processor for spectral analysis".

PROPOSED PAPER TO U.R.S.I., XXth GENERAL ASSEMBLY

Washington, August 1981

Commission C : Signals and Systems

Author : P. GILLE, C.R.P.E. Orléans, France

Subject : Fast Processor for Spectral Analysis

Summary

Interpretation of geophysical measurements may be efficiently achieved with the held of an interactive mini computer coupled to a fast processor devoted to signal processing operations : linear/vectorial, real/complex functions, auto/cross-spectra of an adjustable number of channels. The ORAS ("Operateur Rapide d'Analyse Spectrale") has been mainly designed to give a HP 1000 F computer the capability to quickly calculate and integrate the spectral matrices of VLF satellite data with 9 (or less) components. For instance, the complex Fourier transform with summation takes 20 ms for 1024 samples. The computer software controls the sequence of operations and the data transfers (up to 1 M w/s). It allows the user to interactively define his application as a macroprogram. The ORAS will be used for different purposes, such as electromagnetic noise analysis and could lead to a real time device.

Presentation of ORAS
a fast processor for signal analysis

by P. Gille (CRPE)

at session C 2

URSI XXth General Assembly

Washington, August 1981

^^^^^^^^^^^

CONTENTS

ORAS at a glance

WHAT ?

1. meaning: ORAS = Operateur Rapide d'Analyse Spectrale
   = Fast Processor for Spectral Analysis

2. definition: digital peripheral for a mini-computer
   micro-programmed and macro-programmable
   dedicated to signal analysis operations

WHY ?

3. purposes:
   - linear operations with blocks (or vectors)
   - handling of both complex & real data
   - handling of multi-component digitized signals
   - fast Fourier transform

4. typical application: 6 component electomagnetic field
   telemetered from Geos satellite
   - reorganization of data channels
   - weighted FFT with sliding window averaging
   - complex auto & cross-spectra
   - phase & spin corrections

HOW ?

5. typical performances
   - processing unit cycle time: 200 ns
   - memory access to complex data (2x24 bits): 1000 ns
   - FFT with square mod. and summation: 20 ms for 1024 samples
   - FFT butterfly :5500 ns
   - data transfer: 16 M bits/sec

6. present limitations
   - memory core: 16 K complex data (2 x 24 bits)
   - vector size: 8 to 8096 (32 to 1024 for FFT)
   - number of vectors: 1 to 256 according to size
   - number of signal components: 1 to 16

7. possible extensions
   - new macro-instructions ,e.g. correlation
   - amount of memory core
   - development of real-time or on-board version

WHO ?

8. CRPE (CNET+CNRS): specifications & acceptance
   software for interfacing,tests & applications
9. LETI (CENG): hardware design & firmware

WHEN ?

10. kick-off :sep 76 ,study :77-78 ,development: 78-80
    interface:may 80 ,delivery:jan 81 ,extensions:aug 81

ORAS typical functions

1. Block handling

-operations performed item to item ,data replaced by result
-arithmetic mode : floating point per block on 24 bit mantissa
-automatic normalization after most signif. bits in the block
-automatic adjustment of "block factor" : 8 bit common exponent

2. Fourier transforms & spectra

options: real/complex , direct/reverse , fft/spectrum

3. Arithmetic vector operations

3.1 Real block & real block
- addition,subtraction,multiplication,division
- scalar product (Xk) = (Xk * Yk)
- (coherence) (Xk) = (Xk / Yk)
- weighted summation (Xk) = (Uk * Xk) + (Vk * Yk) + ...

3.2 Real block & real constant
- addition,substaction,multiplication
- sum of block items s = S[Xk]
- linear operations (Xk) = a * (Xk) + b * (Yk) + ...
- multidimensional rotation
- translation,centering (Xk) = (Xk- a) * b

3.3 Signal analysis
- correlation (Xk) = S[X(k) * X(k+j)]
- power,density a = S[Xk * Xk]
- cross power a = S[Xk * Yk]

3.4 Stastitical moments
- average a = S[Xk] / N
- deviation d = S[(Xk - a) * (Xk - a)] / N

3.5 Smoothing,derivatives (Xk) = a * X[k-1] + b * X[k] + c * X[k+1]

4. Complex operations

4.1 Complex block & complex block
- addition,subtraction, direct mult'n ,conjugate mult'n
- square module (Xk, 0) = (Xk,Yk) * (Xk,-Yk)
- 2-dimens. rotation correction (Xk,Yk) = (Xk,Yk) * (cos[a*k],sin[a*k])

4.2 Complex block & real constant
- multiplication

4.3 Complex block & complex constant
- 2-dimens. phase correction (Xk,Yk) = (Xk,Yk) * (cos a , sin a)

4.4 Linear complex operations : similar to real ones

4.5 Processing 2 real vectors as one complex vector

5. Transfer operations

Some data handling are performed via the interface in both trans'n & recep'n
- multi-component reorganization : exchange lines/rows (up to 16 rows)
- combination of 2 real signals in one complex , & vice versa
- binary inversion of addresses (before fft)
- data word matching : 16/24/32 bits with 0 to 16 bits shifted

Short bibliography

1. Signal analysis

- BRIGHAM E.O. :The Fast Fourier Transform ,Prentice Hall,1974
- LEFEUVRE F.,STOREY L.R.O. :The analysis of 6-component measurements of a random electromagnetic wave field in a magnetoplasma,vol II Model idendification ,Note technique CRPE/41, 1977
- LIFERMAN J. :Theorie et application de la transformee de Fourier rapide ,Masson 1977
- MAX J. :Methodes et techniques de traitement du signal et application aux mesures physiques ,Masson 1972
- PELLANDINI :Methodes et structures specialisees pour le traitement digital des signaux ,Journees d'electronique EPFL,Lausanne 1975
- WELCH P.D. :The use of the fast Fourier transform for the estimation of power spectra: a method based on time averaging over short modified periodograms , IEEE Trans. Audio Electroacoust.,15,70,1967

2. Geophysical applications

- BUCHALET LJ.,LEFEUVRE F.: One and two direction models for VLF electromagnetic waves observed on-board GEOS.1 ,J.G.R. vol 86,no A4,1981
- LEFEUVRE F. :Unresolved questions on power line radiation coupling to the ionosphere and magnetosphere ,URSI XXth General Assembly session EH1 opening remarks ,1981
- LEFEUVRE F. :Effets dans la magnetosphere des radiations emises par les lignes electriques , Journees nationales sur la compatibilite electromagnetique , Lille 1981
- LEFEUVRE F.,NEUBERT T.,PARROT M. :Wave normal directions and wave distribution functions for ground based transmitter signals observed on GEOS.1 , J.Geophys.Res.,1981 (to be publ.)
- LEFEUVRE F.,PARROT M.,DELANNOY C. : Wave distribution functions estimation of VLF electromagnetic waves observed on-board GEOS.1 , J.Geophys.Res. vol 86,no A4,april 1981
- SAMSON JC.,OLSON JV. :Some comments on the description of the polarization-states of waves ,Geophys.J.R. astr.Soc.,61,115,1980

3.Technical notes

- CHABERT B. :Operateur microprogrammable pour le traitement du signal, GRETSI 7eme colloque traitement de signal & applications,Nice,1979
- GILLE P.:Presentation de l'ORAS, doc. de trav. CRPE/1061 , Journees sur les machines de traitement de signal en temps reel, Lannion 1980
- PARROT M.: Methodes numeriques d'analyse spectrale ,Note technique CRPE/35 ,1976

ORAS Hardware structure

3 printed circuit boards

1. Processing unit

1.1 Arithmetic & logic processor
- sliced microprocessor (AMD 2903)
- internal registers (ram): 2 x 16 words x 24 bits
- I/O registers : 24 bit input , 24 bit output
- bus 24 bits

1.2 Multiplier : special circuit
[TRW MPY16HJ or MPY24HJ ???]

1.3 Firmware
- microprogram rom : 1 K x 64 bits [extension to 2 K]
- microprogram sequencer (AMD 2910)
- micro-instruction register 64 bits
- instruction register: 24 bits

2. Memory
- memory core: 2 x 16 K words x 24 bits (dynamic memory)
- interface registers (AMD 2907) :2 inputs,2 outputs,1 address
- processing registers (AMD 2907) :2 inputs,2 outputs,1 address
- special multiplexer for binary inversion of addresses

3. ORAS interface to computer
- I/O registers :24 bit input, 32 bit output
- transfer code reg. : 16 bits
- memory address reg. : 24 bits
- word counter: 16 bits
- status register: 6 bits
- logic for command & interrupt system

4. Computer connexion
- interface HP12930 with direct memory access facility (DCPC)
- data output reg.:16 bits , result input reg. :16 bits
- control output reg. :6 bits , status input reg.:16 bits
- tranfer rate: 16 M bits/sec
- cable & connectors :4x37 pins

5. Electrical & mechanical
- standard rack 19" ,5 U
- power consumption :90 watts DC (19 Amps x 5 V)

ORAS-computer configuration

```
               #======#  #======#     bus 24 bits
               "      "  "      "
               "      "  "      "
            ,---"---,-"---"-,---"---,
 ORAS       :       :       :       :
3 circuit   :proces-: memory: inter-:
  boards    :  sor  :  32 K :  face :
            :AMD2903: x 24 b:       :
            :       :       :       :
            :       :       :       :
            ,-------,-------,-::-::-,
                             ::  ::
             6 bits command  ^^  ::    16 bits status
            16 bits data     ::  vv    16 bits result
                             ::  ::
                             \\  //
                               "
                               "
                               "
            ,------------------"----,
            :        Interface      :
            :        HP 12930       :
            :          DCPC         :
            ,-----------------------,
Mag tape    :                       :      Disc unit
 800 bpi  --:                       :----   2x5 Mw
HP7970B     :                       :       HP7906
            :       192 K words     :
            :                       :
Line        :        Computer       :      Console +
printer   --:                       :----  cartridges
HP2631      :        HP 1000 F      :      HP2645
            :                       :
            ,-----------------------,
                  :          :        \
                  :          :         \
                Graphic    Graphic      \
Hard copy       console    console         Plotter
Tek4631         Tek4006    Tek4025         HP9872T
   : :------------:          :
   :-------------------------:
```

ORAS Memory organization

1. Principles

1.1 Different areas
- data & results (e.g. 30 K words) [see below]
- special tables or "system area" (e.g. 2 K words)
  - block factors or exponents
  - cosines for FFT
  - variables & indexes
  - parameters for configuration & run
  - macro-program

1.2 Flexibility & reconfiguration
- location & size of all areas are described in "parameter" table
- special tables (if needed) have to be reloaded from computer at every boot-up
- they may be checked or modified during real-time operations

1.3 Memory access
- word per word or direct memory access
- simultaneous sharing by ORAS proc'g unit & computer interface

1.4 Word length
- 24 bits in ORAS
- 16 bits (integer) or 32=24+8 (floating point) in HP computer
- matching is assumed by interface (missing or extra byte)

2. Data structure

2.1 Block organization
- internal references inside ORAS are block number counts
- one block (or vector) is a data set with same exponent
- all blocks at a given time have same length power of 2 ,from 8 to 8096
- block length is imposed by a specific macroinstruction at any time (e.g. block split in sub-blocks)

2.2 Block exponent (or block factor)
- one 8 bit byte is the binary exponent common for a whole block
- it is considered & updated by arithmetic operations on block
- the block exponent area has a special geometry
- number of blocks & block expo's is limited to 256

2.3 Data types & addressing [see figure].
- data are consid'd as "real" or "complex" according to the macroinst'n
- memory core is divided into 2 word sets:
  - first half (0-16K) or even part,or real part of complex
  - second half (16-32K) or odd part,or imag. part of complex
- the current block length corresponds to a number of double words
- "complex" (or double word) lay-out is the regular one
- "real" type data are put alternatively in both memory halves

ORAS Memory lay-out

1. General lay-out

```
                              addr               addr
 ..............       0.------.  16384 .------....
              ^       1|      :  16385 :      |
              :       2|      :  16386 :      |
 30 K         :        |      :        :      |
              :    ....|......:........:......|....
 data         :    :   | even :        : odd  |   :   one data block
              :    :   | half :        : half |   :
 block        :    ....|......:........:......|....
              :        |      :        :      |
 area         :        |      :        :      |
              :        |      :        :      |
              :  15359 |      :  31743 :      |
 .........X........|------:  .......:------|...
              :  15360 | - - -:  31744 : - - -|
 2 K          :        | - - -:        : - - -|
 system       :        | - - -:        : - - -|
 table        :        | = = =:  . . . : = = =|  . . . block factor
 area         :  16383 | - - -:  32767 : - - -|
 .........v..(16 K).------...(32 K).------....
                       .      .        .      .
                       <......>        <......>
                       24 bits         24 bits
```

2. Data block types Example with block size = 128 double words

2.1 Real block

```
.-----.-----.
| X0   . X1  |
| X2   . X3  |
| X4   . X5  |
| ...  . ... |
| X250. X251|
| X252. X253|
| X254. X255|
.-----.-----.
```

2.2 Complex block

```
.-----.-----.
| Y0   . Z0  |
| Y1   . Z1  |
| Y2   . Z2  |
| ...  . ... |
| Y125. Z125|
| Y126. Z126|
| Y127. Z127|
.-----.-----.
```

2.3 Even block

```
.-----.-----.
| Y0   .  0  |
| Y1   .  0  |
| Y2   .  0  |
| ...  . ... |
| Y125.  0  |
| Y126.  0  |
| Y127.  0  |
.-----.-----.
```

2.4 Odd block

```
.-----.-----.
|  0   . Z0  |
|  0   . Z1  |
|  0   . Z2  |
| ...  . ... |
|  0   . Z125|
|  0   . Z126|
|  0   . Z127|
.-----.-----.
```

3. Binary inversion of addresses

```
.-----.-----.        .-----.-----.
| X0   . X1  |        | Y0   . Z0  |
| X128. X129|        | Y64  . Z64 |
| X64  . X65 |        | Y32  . Z32 |
| ...  . ... |        | ...  . ... |
| X190. X191|        | Y95  . Z95 |
| X126. X127|        | Y63  . Z63 |
| X254. X255|        | Y127. Z127|
.-----.-----.        .-----.-----.
```

Data transfer special features

1. <u>Transfer code</u>
   a 16 bit word specifying all options for a transfer of data & parameters
   The same data block may be transf'd differently in transm'n or recep'n

2. <u>Word management</u>
   Options:-computer word length :16 b (integer) ,32 b (mantissa 24 + expo 8)
   -binary shift: 0 to 16 ,only to the left & in data transmission

```
Examples:                 computer        --->      ORAS        ---> computer

 - 16/24,no shift         | a   b|              | a   b   0|       | a   b|
                          |==,==|               |==,==,--|         |==,==|

 - 16/24,shift=8          | a   b|              | b   0   0|       | b   0|
                          |--,==|               |==,--,--|         |==,--|

 - 32/24,no shift         | a   b   c   d|      | a   b   c|       | a   b   c   0|
    (expo lost)           |==,==,==,--|         |==,==,==|         |==,==,==,--|

 - 32/24,shift=8          | a   b   c   d|      | b   c   d|       | b   c   d   0|
    (msb lost)            |--,==,==,==|         |==,==,==|         |==,==,==,--|
```

3. <u>Block management</u> [see :Memory lay-out]
   Options: - addressing step in ORAS
   2=double word for real & complex (same transfer type)
   1=simple word for even & odd (according to first word address)
   - binary inversion of addresses prior to fft

4. <u>Reorganization of multicomponents</u>

4.1 No reorganization if data already classified channel per channel
e.g. (X0,X1,...X127),(Y0,Y1,...,Y127), ...

4.2 Needed reorganization :multi-channel time sampling
Limitation: 16 channels ,real or complex,with/without binary inversion
Example: 3 channels (real or complex) , block size = 128 double words

```
Real:                                   Complex:
---- computer <---> ORAS                -------        computer  <--->   ORAS
                    ,----,----,                                             ,----,----,
                    |X0  :X1  |                                             |X0  :U0  |
( X0  ,Y0  ,Z0      |X2  :X3  |         ( X0  ,U0  ,Y0  ,V0  ,Z0  ,W0       |X1  :U1  |
(,X1  ,Y1  ,Z1      |    :    |         (,X1  ,U1  ,Y1  ,V1  ,Z1  ,W1       |    :    |
(,X2  ,Y2  ,Z2      |    :    |         (,X2  ,U2  ,Y2  ,V2  ,Z2  ,W2       |    :    |
(,...               |X254:X255|         (,...                               |X127:U127|
(,...               ,----,----,         (,X127,U127,Y127,V127,Z127,W127)    ,----,----,
(,...               |Y0  :Y1  |                                             |Y0  :V0  |
(,X255,Y255,Z255)   |    :    |                                             |    :    |
                    |Y254:Y255|                                             |Y127:V127|
                    ,----,----,                                             ,----,----,
                    |Z0  :Z1  |                                             |Z0  :W0  |
                    |    :    |                                             |    :    |
                    |Z254:Z255|                                             |Z127:W127|
                    ,----,----,                                             ,----,----,
```

List of macro-instructions

1. Interruption management
   - 1 MSYN send "synchronization" signal (status bit 4) to computer
   - 2 MCAL send "end of computation" signal (status bit 3)
   - 3 MEND send "end of computation" and stop macro-program
   - 4 MSEM wait for "end of group transfer" from interface
   - 5 MSSY wait for "synchro. signal" from computer

2. Data block management
   - 6 MTAI set current block length
   - 7 MBRA decrementation of variable & skip if negative
   - 8 MRAZ reset one block to zero & proper factor to -128
   - 9 MTRF transfer one block factor into another factor
   - 10 MEXF copy one block factor into following ones
   - 11 MTRA transfer one block (with factor) into another block
     opt 0:direct copy , opt 2:copy with binary inversion of addresses
     opt 4:compress even part of 2 blocks into one "real" block

3. Arithmetic operations on blocks
   - operation item to item, result in the first block,factor updated
   - 12 MADD addition block to block
   - 13 MSUB subtraction
   - 14 MMUL real multiplication
   - 15 MMLC complex multiplication
   - 16 MMCC complex conjugate multiplication
   - 17 MDIV real division

4. Block & scalar operations
   - 18 MADC addition of real variable to block
   - 19 MSUC subtraction of real variable to block
   - 20 MMUC multiplication of block by real variable

5. Fourier transforms [see special chapter]
   - 21 MFFT real,direct, 22 MTFF real,reverse, 23 MMOD real,dir. & square
   - 24 MCFT co'x,direct, 25 MCFF co'x,reverse, 26 MCMO co'x,dir. & square

6. Miscellaneous [* indicates new instructions]
   - 27 MCHV load integer or floating constant into one variable
   - *28 MDEC shift words inside a real block
     options:left/right,open/close,shift number
   - *29 MINT summation of all block items
   - *30 MCCC multiplication of complex block by complex variable
   - *31 MTVV (opt 0) transfer one variable into another one
   - *31 MGSU (opt 1) call to subroutine
   - *31 MRET (opt 2) return from subroutine
   - *31 MBAV (opt 3) copy one block item into one real variable

7. Pseudo-instructions
   - (44)COM comment
   - (45)VAL definition of constant
   - (46)EQU definition of symbolic label
   - (47)ORG definition of initial address
   - (48)FIN end of coding

ORAS FFT special features

1. <u>Principles</u>

1.1 Result block takes location of data block,with same length
they may be "real" or "complex" or "even" or "odd"

1.2 Real signal may be proc'd either as "real" or "even" block
according to related calculations

1.3 Direct and reverse FFT are performed with the same algorithm
the results appear always in natural order

1.4 Direct auto-spectrum (FFT followed by square module) may be performed in
a single instruction .Result is even :complex with imaginary part null

1.5 Reverse FFT gets data & results in natural order

1.6 Real FFT takes care of hermitian symmetry
2*N real data give N complex results for positive frequencies only
imagin. part of initial value (frequency zero) should be zero
but it is replaced by real part of medium value (frequency N)

1.7 Direct FFT needs a preliminary pass for address inversion
- either by interface during data transm'n ,due to special tr. code
- or by internal transfer performed by a macroinstruction

1.8 Limitations are due to hardwired preliminary binary inversion
typically 16 to 1024 complex (or 32-2048 reals)

1.9 Cosine-sine table has to be loaded from computer
with maximum size (513 pairs for half a period:0-pi)

2. <u>Special macro-instructions</u>

The number of samples corresponds to the same block size "N" (double words)

function name	MFFT	MTFF	MMOD	MCFT	MCTF	MCMO
type	real direct	real reverse	real direct square	complex direct	complex reverse	complex direct square
data number	2*N	N	2*N	N	N	N
type	real	complex	real	complex	complex	complex
dispo	inverted	ordered	inverted	inverted	ordered	inverted
result number	N	2*N	N	N	N	N/2
type	complex	real	even	complex	complex	even
dispo	ordered	ordered	ordered	ordered	ordered	ordered
frequency step	1/2N	1/2N	1/2N		1/N	
freq range (Re)	0, N	0,N-1	0,N-1		-N/2 , N/2-1	
(Im)	1,N-1	0,N-1	0,N-1		-N/2 , N/2-1	

```
----+----,----+----,----+----,----+----,----+----,----+----,----+----,----+----
      F o u r i e r   t r a n s f o r m s   &   s p e c t r a  oras URSI 81  page 12/18
      ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^
1. Complex data ( Y0,Z0,Y1,Z1, ... ,Y127,Z127 )
   ------------                /  \
   transfer:          no inv/    \ inv
                            /      \
           co'x,ord'd /             \ co'x,inv'd
           ,----,----,               ,----,----,
           :Y0   .Z0  :              :Y0   .Z0  :
           :Y1   .Z1  : MTRA(2)     :Y64  .Z64 :
           : ..  . .. : ------>      : ..  . .. :
           :Y126.Z126: intern       :Y63  .Z63 :
           :Y127.Z127: bin inv      :Y127.Z127:
           ,----,----,               ,----,----,
                ^                       / \
            MCTF \           MCFT  /   \  MCMO
           reverse\         direct /     \ direct & spec
                    ^               V       V
                  \Fourier tr/          \auto-spec                 compr.autosp.
                   ,----,----,           ,----,----,               ,----,----,
                   :F0   .G0  :          :A0   .  0  : MTRA(4)   :A0   .A1  :
                   :F1   .G1  :   MMUL   :A1   .  0  : ------>    :      .    :
                   : ..  . .. : ------>  : ..  . .. :            :A126.A127:
                   : ..  . .. :  square  : ..  . .. :            ,----,----,
                   :F126.G126:           :A126.  0  :
                   :F127.G127:           :A127.  0  :
                   ,----,----,           ,----,----,
2.Real data
  ---------
             ( X0,X1,X2,      ......         ,X253,X254,X255 )
              /   \                                  /     \
             /     \                                /       \
 real ord'd /       \ real inv'd          even ord'd /         \ even inv'd
 ,----,----,          ,----,----,          ,----,----,            ,----,----,
 :X0   .X1  :         :X0   .X1  :         :X0    .  0  :          :X0    .  0  :
 :X2   .X3  : MTRA(2) :X128.X129:         :X1    .  0  : MTRA(2) :X128.  0  :
 : ..  . .. : ------> : ..  . .. :         : ..   . ..  : ------>  : ..   . ..  :
 :X252.X253: intern  :X126.X127:          :       .     :          :       .     :
 :X254.X255: bin inv :X254.X255:          :       .     :          :       .     :
 ,----,----,          ,----,----,          :       .     :          :       .     :
    ^                     / \              :X254.  0    :          :X127.  0    :
     \                   /   \             :X255.  0    :          :X255.  0    :
 MTFF \       MFFT /       \ MMOD          ,----,----,            ,----,----,
 rev   \      dir /         \ spec              ^            /        :
        \        /           \           MCTF  :      MCFT/         : MCMO
         ^      /             v                 ^          /          v
 half Feurier v           auto spec       Hermit.Four.      v       Herm.autosp
 ,----,----,              ,----,----,      ,----,----,               ,----,----,
 :F0   .F128:**           :A0   .  0  :*   :F0   .G0   :             :A0   .  0  :
 :F1   .G1   :            :A1   .  0  :    :F1   .G1   :   MMUL      :A1   .  0  :
 : ..  . ..  : --//--> : ..  . .. :    : ..  . ..   : ------>    : ..  . ..  :
 :F127.G127:              :A127.  0  :     :F127.G127:                :A127.  0  :
 ,----,----                ,----,----,    *:F128.G128:*             *:A128.  0  :
**(F128)                                    :F127.-G127               :A127.  0  :
                                            : ..  . ..  :              : ..  . ..  :
                                            :F2   .-G2  :              :A2   .  0  :
                                            :F1   .-G1  :              :A1   .  0  :
                                            ,----,----,                ,----,----,
----+----,----+----,----+----,----+----,----+----,----+----,----+----,----+----
```

ORAS & computer: task organization

1. General principles

- ORAS and HP1000 computer work simultaneously
- synchonization is made by means of interruptions and flags
- all data exchanges are computer-directed

2. Operations inside computer

2.1 Foreign relations

- external data management :disc,tape,etc
- auxiliary computations e.g. calibrations
- digital & graphic display
- interactive functions: start-up,break,etc

2.2 Preparation tasks

- building up auxiliary tables
  - configuration parameters
  - trigon. tables:cosines,weighting
- compilation of symbolic macro-program
- initializations, ORAS reset

2.3 Run time tasks

- control of all input/output op'ns
- control of task activation & completion in ORAS

3. Operations inside ORAS processing unit

- macro-instruction decoding
- micro-instr'n execution, management & arithmetic functions
- update of parameters,variables,status,error flag, etc
- (no I/O)

4. Operations performed by ORAS Interface

4.1 Bidirectional I/O general tasks

- data & result transfers ,as required by computer
- emission & echo of interruptions

4.2 Special features due to data structure

- complex or real (double or simple) data addressing
- word format matching :16 or 32 bits in HP ,24 bits in ORAS
- reorganization of up to 16 multicomponents per channel block
- binary inversion of addresses (anticipation of next FFT)
- word shift ,0 to 15 bits to the left (in data transm'n only)

Computer software structure for ORAS

1. Libraries

1.1 Elementary control of input/output
- SOCOM command word output
- ENSTA interface status word input
- ENDAT/SODAT word per word i/o
- DMAES direct memory access i/o

1.2 Protocol for data block exchange
- ENSOB bidirectional buffer transfer to a given address according to data type & structure
- EXTRA transfer of special tables: cosines,parameters,macroprog.,etc
- EXECO management of commands & status for run-time control: macro-pointer,error flag,start/stop,wait for synch.,etc

1.3 Utilities
- FAMAC macro compiler & des-assembler
- FAPON build weighting for FFT
- FABRI build synthetic data & transfer code: Dirac,sine,saw tooth,etc
- FAVAR editing of any data or data set :decimal,octal,hexad.
- GRADO graphic display according to data type (r/c ,multicomponent)

2. Interactive general control (PORAS program)

2.1 Purposes
- activation of any function at any level
- access to any data or parameter
- check-out test :transfers,interruptions,macro-inst'n
- initializations & control of application program

2.2 Operations
- 80 keywords covering the different functions or function sets
- functions may be run step by step or linked or repeated
- command keywords may be entered via:
  - -either a terminal keyboard
  - -or a symbolic command file

3. Application program (e.g. POWEL)

3.1 Double coding: two descriptions have to correspond:
- command file including macroprogram & associated functions (initial'n)
- fortran sequence including CALLs to ORAS libraries for data formatting ,transfers & synchronizations

3.2 Parallel work :flip-flop operations are feasible with help of sync's typically:transfer of next buffer while current one is processed

3.3 Auxiliary control: computer time-sharing enables to run simultaneously from different terminals the Applic'n and the Interactive cont'l program

Application to GEOS Survey data

1. Purpose

A contribution to study of power-line radiations after spectral analysis of VLF magnetic field via eigen-values of spectral matrices
This is the first operational use of ORAS system although it needs some additional computations (eg eigen-values)

2. Characteristics of data

- origin: GEOS.1 satellite ,"Survey mode" ,recorded on magnetic tape
- total bandwidth :9335 Hz ,swept in 31 steps with frequency overlay
- step bandwidth :744 Hz ,of which 296.16 are undependant
- one "high speed format"(hsf) = 128 samples with constant gain per antenna
- one step = 688 ms = 8 hsf = 1024 real data * 3 antennae

3. Task summary

3.1 Data selection

- select the 3 magnetic components (Bx,By,Bz) from the tape
- gain correction for the 8 hsf
- averaging to zero : suppression of DC component
- rotation correction on X and Y axis (Z has a fixed attitude)

3.2 Spectral analysis (via ORAS)

- sliding window per step for each of the 3 real channels ,ie:
  split the 1024 in 5 segments of 256 ,with overlay of 64
- weighting of each data segment (Parzen window)
- 3 real direct fft per segment : 128 Hermitian complex
- calculate 3 real auto-spectra and 3 complex cross-spectra
- correction of antenna phase on 2 cross-spectra
- accumulation of the 6 corrected spectra for the 5 segments

3.3 Eigen-values

- select useful bandwidth : 51 of 128 frequency points
- re-arrange the spectral matrices in triangular real array (6*6/2)
- compute eigen-values by pair : 3 free E.V. per frequency

3.4 Graphic display

options:raw data,fft,spectra,eigen-values

5. Performance

timing : 360 ms per step of spectral analysis (see above 3.2)
compatible with the real-time sampling (688 ms)
i.e. conversion of 3*1024 real data into 128 spectral matrices * 9 items
this includes mainly 15 weighted FFT and spectrum calculations
all information transfer ORAS-computer
the task synchronization is not optimized (no parallelism)

4. Possible extensions & improvements

- handle the 6 electromagnetic components
- accomodate the sliding window inside ORAS via block shifting
- build a flip-flop processing :transfer parallel with computation

ORAS-computer parallel operations

an example of sequential operations (no flip-flop)

COMPUTER	INTERFACE	ORAS
1. General interactive control		
- initializations,reset	command	- stand by
- transfer system tables		
cosines		
parameters		
macroprogram		
- transmit auxiliary data	data	
weighting		
despin		
rephasing		
- check data & status	result	
2. Application program		
2.1 Initialization		
- activate macro-program (ACMA)	command	
		- start macro-program
2.2 Loop		
		- loop beginning
- wait for "ready for data"	status	*send "ready for data"(MSYN)
- transmit data segment	data	
	-reorg.	
	-bin.inv	
	-shift 8	
- send synch "end of data"(ACSY)	command	*wait for "end of data"
		*computation sequence
		+internal transfer
		+weighting
		+FFT,spectra
		+phase correction
		+accumulation
- wait for "end of computation"	status	*send "end of comput'n"(MCAL
- fetch results	result	*jump to loop beginning
		- send "stop"(MEND) and stop
2.3 Related computations		
- eigen-values		
- graphic display		
3. General interactive control		
- checkstatus (if needed)	status	

```
***     #POWEL:OR:12    oras-command file 30.6.81
TRFA,TRVA,TRPA          reset parameter area
TRFE                    build & transfer weighting block (15)
     256,1, 15
TRRO                    rephasing blocks (16,17,18)
     128,2, 16,   0, 186,3600
        , , 17,-450,2011,
        , , 18,-675,1825,
FATA                    specify data block size (0 to 14)
     128,2,   0,15
COMA                    compose macro program
        ORG,0
          MTAI,120,128,2 block size
          MRAZ,3         reset before accumulation
          MRAZ,4
          MRAZ,5
          MRAZ,6
          MRAZ,7
          MRAZ,8
          MCHV,1,999,0   iteration index
        COM              loop beginning
          MSYN           send sync signal ,ready for new data
          MSSY           wait for ACSY ,data transferred
          MTRA,0,12,0    internal data copy
          MTRA,1,13,0
          MTRA,2,14,0
          MMUL,12,15     weighting
          MMUL,13,15
          MMUL,14,15
          MFFT,12        real fft
          MFFT,13
          MFFT,14
          MTRA,12,9,0    duplicate before spectra
          MTRA,13,10,0
          MTRA,14,11,0
          MMCC,9,13      complex cross spectra
          MMCC,10,14
          MMCC,11,12
          MMCC,12,12     even auto spectra
          MMCC,13,13
          MMCC,14,14
          MMCC,13,17     phase corr yz,zx
          MMCC,14,18
          MADD,3,12      sum 6 spectra
          MADD,4,13
          MADD,5,14
          MADD,6,9
          MADD,7,10
          MADD,8,11
          MCAL           signal end of iteration
          MBRA,1,-60     end of loop
          MEND           signal end of oras processing
        FIN              end of symbolic macropgm
TRMA,ACMA               load & start  macropgm
TRIN                    end of command file
```

```
----+----,----+----,----+----,----+----,----+----,----+----,----+----,----+----
```

ORAS Fortran calling sequence

```
      PROGRAM POWEL ( ), oras-power lines 30.6.81
C   process 3 components of earth magnetic field
C   256 real data , 128 complex results
C   prerequisite: oras initialisation by PORAS program
C                 with #POWEL command file
      COMMON/CODIS/ LUCO(2),LIMG(8,5),TITR(24)
      COMMON/STOCK/ITABL(3200),MEIG(3,51,31)
      DIMENSION MATS(2, 768), MAGN(3,1024)
      DIMENSION IFICH(16),TETE(16) ,IFABL(16)
      EQUIVALENCE (ITABL(129),MATS,MAGN)
      DATA LU/1/,IMP/6/,ISEVR/1/
      DATA LOBLO/1024/,LOSEG/256/,LOVER/64/,NCOMP/3/
      DATA LOBUF/1024/,LOBUS/128/,LOSPE/128/
      DATA NBSEG/5/ ,NBLOS/3/,NBLOF/9/ ,NELEM/6/
      DATA NUFR1/28/,NUFR2/ 78/,NBFRQ/51/,NSTEF/31/ ,NTOTF/1581/
      DATA IFICH/2H#S,2HUR,2HV2,2HMP/
      DATA TETE/   15* 4H     ,4HORAS/
C
C 'ZERO' reset array for eigen-values
      DO 135 K=1,NSTEF
      DO 135 J=1,NBFRQ
      DO 135 I=1,NCOMP
135   MEIG(I,J,K)=0
C
      DO 2300 KSTE=1,NSTEF
C 'SURV' access to geos survey data
      CALL SURVA(LU,IFICH,ITABL,3200,IS4)
      WRITE(LU    ,141) (ITABL(I),I=1,5),(ITABL(I),I=126,128)
C 'OPER'  reset macroprogm & accumulators
C         transfer to oras (sliding window with overlay)
C         activation of macroprogram (ACSY)
      CALL GLIMA(IMP,ISEVR,MAGN(1,1),NCOMP,LOSEG,LOBLO,LOVER)
C 'RSPE'  wait for "end of computation" (ACMA synch)
C         retrieve average spectral matrices
C         overlay on raw data
      CALL RESPE(IMP,ISEVR,MATS(1,1),IFABL(1),LOSPE,NBLOS,NELEM)
C 'EIGN' computes Eigen-values in limited bandwidth
      CALL PEIGN(IMP,MEIG(1,1,IS4+1),MATS,NUFR1,NUFR2)
C 'PLGN' plot eigen-values for current frequency step
      CALL GRAGN(IMP,LUCO,MEIG(1,1,IS4+1),3,NBFRQ, IS4 ,1
     .           ,LIMG,TITR,TETE)
2300  CONTINUE
C
      END
```

```
----+----,----+----,----+----,----+----,----+----,----+----,----+----,----+----
```

CNET imp 75.1163