

Informe Técnico CAY 1988-4

CONVERSIÓN DE FICHEROS ANAE EN FITS

Mario Tafalla, Josefina Montalbán, Pere Planesas.

El Señor bajó a ver la ciudad y la torre que estaban construyendo los hombres; y se dijo:

-Son un pueblo con una sóla lengua. Si esto no es más que el comienzo de su actividad, nada de lo que decidan hacer les resultará imposible. Vamos a bajar y a confundir su lengua, de modo que uno no entienda la lengua del prójimo.

Génesis 11,5-7

ÍNDICE

1. Introducción
2. El formato ANAE
  - \* Cabecera
  - \* Espectro
3. El formato FITS
  - \* Sintaxis FITS
  - \* Palabras clave
  - \* Ejemplo
4. El programa ANAFITS
  - \* Descripción de los comandos
  - \* La subrutina ANAFI
    - a. Cabecera
    - b. Espectro
5. Apéndices
  - I. Fichero en formato ANAE
  - II. Fichero en formato FITS
  - III. Listado del programa ANAFITS
  - IV. Equivalencias FITS-ANAE
  - V. Programas auxiliares

## 1. INTRODUCCIÓN

La conveniencia de poder utilizar los programas de análisis y tratamiento de imágenes procedentes del Grupo de Astrofísica de Grenoble (CLASS, LAS, GREG, GILDAS) para la reducción e interpretación de datos obtenidos en el radiotelescopio del CAY, así como la necesidad, tanto para observadores del CAY como visitantes, de poder llevar a otros centros datos procedentes de este telescopio en un formato estándar, han motivado la creación de un programa de conversión de ficheros de espectros en formato ANAE (propio del CAY) a formato FITS.

Los programas LAS y CLASS no utilizan este formato, pero disponen de comandos (en el caso de LAS), o programas auxiliares (en el de CLASS) que permiten automáticamente la conversión FITS-LAS/CLASS. Además, el hecho de tener que pasar por el formato intermedio FITS garantiza que una posible modificación en el formato LAS/CLASS no afectará a la transmisión de datos ya que el primero no sufre cambios.

El proceso inverso (esto es, la traducción de formato FITS en formato ANAE), necesario para poder analizar con el programa ANAE los espectros procedentes del radiotelescopio del IRAM en Pico Veleta, es posible mediante el programa VEREC (Pere Planesas, Informe Técnico CAY 1986-2).

En este informe se resumen algunas características de los formatos ANAE y FITS y se presenta el programa ANAFITS, para la conversión del primero en el segundo.

Este programa se inscribe dentro de los trabajos de informática destinados a la transmisión de datos observacionales entre distintos ordenadores, dentro del Proyecto de investigación número 477/83 de la CAICYT.

## 2. EL FORMATO ANAE

El formato ANAE (ANÁLISIS de Espectros) es propio del CAY y es el utilizado para almacenar los espectros obtenidos con el radiotelescopio de este centro y analizarlos posteriormente con el programa del mismo nombre (sobre el programa ANAE, véase P.Planesas y L.Delgado, Informe Técnico 1984-3).

Cada fichero en formato ANAE ocupa un único registro escrito sin formato. Las primeras 128 palabras (una palabra, en el ordenador HP1000, son 2 bytes de ocho bits cada uno) contienen todos los datos acerca de la observación y constituyen la cabecera. A continuación se almacena el espectro, utilizando una palabra por canal. Así, en el caso de un espectro de 2 vías, cada una con 128 canales, (resultado normal de una observación en conmutación de frecuencia con el banco de filtros de 256 canales) tendremos que un espectro necesita, para ser almacenado, un registro de 3x128 palabras dividido en tres partes: cabecera (128 palabras con información alfanumérica), vía 1 (128 palabras con valores numéricos) y vía 2 (igual que vía 1). Como máximo se pueden tratar espectros de una vía de 1024 canales o bien de 2 vías de 512 canales cada una, ocupando en este caso 9x128 palabras el registro completo.

La división cabecera-espectro es sólo lógica, y no física, ya que la única separación de este tipo es el fin de registro, que indica el final del espectro, sin que haya ninguna marca entre la cabecera y el espectro.

### \* Cabecera.

Dado que uno de los objetivos de este formato es la economización de espacio en el disco del ordenador, el almacenamiento de los datos en la cabecera se hace sin formato, y ocupando cada concepto el número de bytes adecuado a su tamaño. Algún dato ocupa un único byte (como la banda lateral que es un sólo carácter: S o I). Otros dos (enteros), otros cuatro (reales y enteros\*4) e incluso seis (reales\*6, como F1 y F2). La distribución de datos en la cabecera es, por tanto, necesariamente rígida e invariable. La distribución de los bytes de la cabecera puede verse en el ya citado Informe Técnico 1986-2.

\* Espectro

Los valores del espectro (esto es, los valores de  $T_a$  para cada canal) se almacenan como enteros\*2. Para ello existen, en la cabecera, un factor de escala y un desplazamiento, tales que al multiplicar cada entero por la escala y sumar el desplazamiento obtenemos el valor de  $T_a$  (en K). El número de canales y el de vías también figuran en la cabecera. Nótese que un espectro de sólo la vía 2, tiene la vía 1 a 32767 y el tercer grupo de 128 palabras ocupado. Sin embargo, uno con sólo vía 1 tiene únicamente dos grupos de 128 palabras y no almacena espacio para la vía 2.

En el apéndice I se muestra un ejemplo de un fichero en formato ANAE, descompuesto en los diferentes conceptos de la cabecera tras una conversión apropiada de formatos. También se muestra una vía almacenada en enteros. Obsérvese que el canal 10 se ha hecho mudo, y por tanto el valor entero que le corresponde es el -32768

### 3. EL FORMATO FITS

FITS (Flexible Image Transport System) es un formato estándar para la transmisión de imágenes astronómicas sobre cinta magnética. Es un sistema de amplias aplicaciones: interferometría, cámaras CCD, espectroscopía radio, etc. Nos interesa aquí en su aspecto de transmisión de espectros radio (para conocer sus diversas posibilidades, véase Wells et al (1981), Ap. J. Suppl. Ser. 44, 363).

#### \* Sintaxis FITS

Un fichero en formato FITS (en el apéndice II se muestra un ejemplo) consta de un cierto número de registros de 23040 bits (2880 bytes de 8 bits) cada uno en formato ASCII. Los primeros registros contienen la cabecera, y los siguientes el espectro.

La cabecera está formada por cadenas de 80 caracteres cada una, de la forma

clave = valor / comentario

La palabra clave (necesariamente debe ir en mayúsculas) es una cadena de 8 caracteres ajustada a la izquierda entre las columnas 1 a 8. El signo '=' se encuentra en la columna 9 y la columna 10 se deja en blanco. Tras el valor de la variable, y separada por un blanco, puede haber una diagonal y un comentario para indicar el tipo y características de la variable. Pueden existir líneas que contengan únicamente comentarios. En este caso sólo es necesario que aparezca una clave especial que indique que se trata de tal tipo de línea, y las restantes columnas no están sujetas a ningún formato.

La cabecera puede ocupar un número arbitrario de registros (como mínimo debe ser uno). La última cadena de la cabecera debe tener únicamente la palabra clave END. El resto del registro se completa con blancos. Los siguientes registros (si existen, ya que no es necesario) contienen el espectro. Éste se almacena en forma de enteros, en binario, donde cada uno ocupa un número de bits (8, 16 o 32) indicado previamente en la cabecera. Si el conjunto de valores no llena el registro, el resto queda en blanco.

Los espectros van separados por un fin de fichero

## \* Palabras clave

La primera línea de la cabecera debe ser necesariamente de la forma

SIMPLE = variable lógica

donde la variable lógica puede indicar si se trata de un fichero que se adapta al formato FITS estándar (SIMPLE = T) o no (SIMPLE = F).

La segunda línea debe contener la palabra clave BITPIX (bits per pixel) e indica el número de bits que ocupa cada entero que almacena un valor del espectro.

La tercera cadena debe tener como palabra clave NAXIS, que indica el número de ejes (índices) de la matriz que contiene el espectro.

Las cadenas siguientes indican (palabra clave NAXISn n=1,...,NAXIS) los valores máximos que pueden tomar los índices de la matriz del espectro. Así, éste viene almacenado en una matriz que en un programa FORTRAN estaría especificada por la sentencia

DIMENSION ESPECTRO(NAXIS1,NAXIS2,...,NAXISnaxis)

Las líneas anteriores (y la que sólo contiene END, que es la última de la cabecera) son necesarias y deben ir en el orden mencionado. Puede darse el caso de que el número de ejes sea cero y el fichero no contenga ningún espectro. Las siguientes palabras clave no son estrictamente necesarias y pueden ir en cualquier disposición. Los ficheros FITS que se crean con el programa ANAFITS tienen un orden fijo, que hemos tomado de los que son creados por CLASS, pero esto no es ningún requisito esencial.

Las variables BSCALE y BZERO convierten los valores enteros transmitidos en la cinta a los auténticos valores del espectro (en la escala indicada por la variable BUNIT) según la fórmula

VALOR REAL = (VALOR ENTERO \* BSCALE) + BZERO

CTYPEn, CRVALn, CDELtN y CRPIXn se refieren al eje n-ésimo. El primero indica el tipo de variable que corresponde a este eje. CRVALn indica el valor de esa variable en el "pixel" de referencia. CDELtN es el incremento de la variable al pasar de un "pixel" al siguiente. CRPIXn indica el "pixel" de referencia. Así el valor de la variable CTYPEn en el "pixel" i-ésimo viene dada por

VALOR DE CTYPEn(I) = CRVALn + (I - CRPIXn) \* CDELtN

Las palabras clave COMMENT, HISTORY, o las primeras 8 columnas en blanco indican que se trata de una línea de comentario y que puede contener cualquier tipo de información.

El resto de las variables que nos interesan son

fácilmente identificables y contienen información sobre el lugar y las condiciones en que se realizó la observación.

\* Ejemplo

El apéndice II muestra un fichero en formato FITS escrito por el programa ANAFITS, una vez que se ha separado cada registro de la cabecera en líneas de 80 caracteres (recuérdese que ésta no es una separación física sino convencional).

SIMPLE = T muestra que se adapta al estándar de FITS  
 BITPIX = 16 indica que el espectro se almacena en enteros de 16 bits (enteros\*2).

El número de ejes es 4 (NAXIS=4) y los ejes son

- . Frecuencias (canales) (NAXIS1=128)
- . Ascensiones Rectas (NAXIS2=1)
- . Declinaciones (NAXIS3=1)
- . Parámetros de Stokes (NAXIS4=1)

Aunque el número de ascensiones rectas y declinaciones es 1, podemos, sin variar el formato, almacenar con una única cabecera un conjunto de espectros procedentes de diversas posiciones (mapa), sin más que permitir otros valores de NAXIS2 y NAXIS3.

CTYPE1 = 'FREQ' indica que el primer índice de la matriz que almacena el espectro representa la frecuencia y la diferencia de frecuencias entre dos "pixels" (canales) es CDELT1, que corresponde a la resolución del espectrómetro. El canal CRPIX1 (=64) es el canal de referencia y está asociado a una frecuencia CRVAL1, que en este caso es cero porque está referida a la frecuencia en reposo RESTFREQ, y no se ha introducido ningún desplazamiento respecto a ella. El valor de la frecuencia en reposo para el principio del canal i-ésimo viene dado por

$$F(I) = \text{RESTFREQ} + \text{CRVAL1} + (I - \text{CRPIX1}) * \text{CDELT1}$$

La velocidad Doppler correspondiente al principio del mismo canal es

$$V(I) = \text{VLSR} + (I - \text{CRPIX1}) * \text{DELTAV}$$

donde DELTAV es la resolución en velocidades (y por tanto tiene siempre signo opuesto a CDELT1).

Los ejes 2 y 3 representan la ascensión recta y la declinación (como viene indicado por CTYPE2 = 'RA', y CTYPE3 = 'DEC'). CRVAL2, y CRVAL3 son la A.R. y DEC. del origen. CDELT2 y CDELT3 son los desplazamientos ("offsets"), respecto a esa posición central, del punto a que corresponde el

espectro.

Las demás variables proporcionan información como el nombre de la fuente, el nombre de la línea, etc., que permiten una más fácil gestión de los espectros a la hora del análisis.

Nótese que las últimas líneas son de comentario. Algunas empiezan con la clave HISTORY, y otras con la clave en blanco. El programa que traduzca ficheros FITS a otro formato deberá ignorar este tipo de cadenas.

La última clave es END, requisito imprescindible de este formato, quedando el resto del registro en blanco.

#### 4. EL PROGRAMA ANAFITS

Es un programa interactivo que permite la conversión de ficheros con formato ANAE en ficheros con formato FITS.

Los ficheros FITS se grabarán directamente en una cinta magnética que se encuentre en la unidad de cinta de 1600 bpi. Por ello es necesario, antes de ejecutar el programa, encender esta unidad, colocar una cinta (con anilla de grabación) y pulsar los botones "Load" y "Online". Cuando la luz roja de "Online" esté estabilizada, será la indicación de que la unidad de cinta está disponible para ser usada. También debemos comprobar que la impresora está encendida y en línea, ya que el programa escribirá información a través de ella.

Para llamar al programa se debe teclear ANAFITS. Aparecerá el cursor

ANAFITS>

que nos indicará que nos encontramos dentro del programa y que podemos introducir los comandos. Estos son los siguientes:

- \* ADIOS
- \* AYUDA/HELP
- \* COMENTARIO
- \* DATO
- \* DIRECTORIO
- \* ESCRIBE
- \* REBOBINA

#### \* Descripción de los comandos

ADIOS. Despide el programa.

AYUDA. Da acceso al fichero de ayuda.

COMENTARIO. Permite introducir, opcionalmente, en cada fichero FITS, un conjunto de líneas (hasta un máximo de 36) en forma de comentario. Este texto debe encontrarse previamente escrito en un fichero cuyo nombre se indica a continuación del comando. Por ejemplo

ANAFITS> COMENTARIO FICHERO.COME

El fichero de comentario no tiene restricción en cuanto al

contenido ni a su formato. Sí existe un límite a su tamaño: debe constar de a lo sumo 36 líneas y cada una de éstas no debe superar los 72 caracteres. El programa añade a cada línea, 8 blancos al comienzo, constituyendo las cadenas de 80 caracteres del formato FITS. En caso de exceder estos límites se perderán fragmentos de líneas (a partir de la columna 73) o líneas enteras (si hay más de 36). La existencia de esta limitación, por otra parte arbitraria, es debida a que el fichero de comentarios se guarda en una variable carácter que debe tener un tamaño máximo previamente establecido. Este modo de actuar permite un ahorro de tiempo, ya que el fichero debe abrirse una única vez, pero debe ser copiado en cada espectro (lo que puede suponer varios cientos de escrituras).

Si hemos estado utilizando un fichero de comentarios para unos espectros, y queremos que los siguientes no contengan ninguno, comandaremos

```
ANAFITS> COMENTARIO NO
```

Nota: Dado que la cadena de caracteres "NO" es considerada como una indicación de que se prescindirá del fichero de comentario, debe evitarse dar este nombre al fichero. Sí es permisible utilizar cualquier otro nombre, aunque comience por "NO", pero que tenga un tercer carácter distinto del blanco.

Si introducimos el comando

```
ANAFITS> COMENTARIO ?
```

el programa nos devolverá el nombre del fichero de comentarios, en caso de tener abierto uno, o nos avisará que no hay fichero si ese es el caso.

DATO. Sirve para indicar el fichero que contiene los espectros en formato ANAF. Tras la palabra clave se debe escribir el nombre de dicho fichero. Es necesario ejecutar este comando antes del comando "ESCRIBE":

```
ANAFITS> DATO FICHERO.ANAF
```

Al igual que en el comando "COMENTARIO", si en lugar de un nombre de fichero introducimos "?", el programa nos contestará el nombre del fichero de datos.

DIRECTORIO. Permite conocer o cambiar el directorio de trabajo. Para cambiar de lugar se debe escribir, a continuación del comando, el nombre del nuevo directorio, si no especificamos ninguno, el comando nos indica en qué

directorio estamos.

ESCRIBE. Es el comando fundamental. Llama a la subrutina ANAFI (ver más abajo) que escribe físicamente, sobre la cinta, los espectros en formato FITS. Para su ejecución es necesario abrir un fichero de datos y, opcionalmente, uno de comentarios. Tiene varias formas de actuar: si seguidamente de "ESCRIBE" (y tras un blanco, por supuesto) aparece la palabra "DESPUES", el programa buscará un final de cinta (dos finales de fichero consecutivos), retrocederá una posición, y escribirá a partir de ahí. Si no aparece esta clave, comenzará a escribir directamente en la posición en que se encuentre.

Nota: La señal de fin de cinta se genera al escribir con este programa, de manera que una cinta utilizada para almacenar otro tipo de datos no la contiene necesariamente. Si desean escribirse espectros FITS al final de una cinta que contiene otra información deberá grabarse el fin de cinta, por ejemplo utilizando el comando "CN 8 EO" del CI.

El cualificador "DESPUES" es opcional. Tras él (o en su lugar, si no aparece) debe figurar una indicación de los identificadores de los espectros a transcribir. Ésta puede ser:

TODOS. Copia, en formato FITS, la totalidad de los espectros que contiene el fichero de datos.

DE ID1 A ID2, siendo ID1 e ID2 dos números enteros. Copia todos los espectros que se encuentran en el intervalo comprendido entre el que tiene identificador ID1 y el que tiene ID2, ambos inclusive. Nótese que no se trata de un intervalo por orden creciente, sino que escribe, dada la lista de los espectros ANAE, todos los espectros entre desde que encuentra el ID1 hasta leer el ID2. Este criterio parece más conveniente que el orden por tamaño, ya que normalmente uno estará interesado en transcribir espectros que se encuentran seguidos en el fichero ANAE, por corresponder a un intervalo temporal de observaciones sin importar que el orden de los identificadores sea creciente y éstos sean consecutivos.

IDEN, siendo IDEN un número. Copia únicamente el espectro con identificador indicado.

El comando "ESCRIBE" puede ser utilizado todas las veces que sea necesario, moviéndose hacia delante o hacia atrás en la tabla de identificadores del fichero de datos. Durante su ejecución va indicando por la pantalla el número del espectro que está transcribiendo. Si deseamos parar la escritura, sin abortar el programa, pulsamos cualquier tecla hasta que aparezca

S=l COMMAND ?

tras lo que respondemos "BREAK". El programa termina de escribir el espectro que estaba transmitiendo, y vuelve a pedir un nuevo comando. Esta operación no produce ningún problema posterior: podemos seguir escribiendo, rebobinar, etc.

REBOBINA. Rebobina la cinta magnética.

---

RESUMEN DE LOS COMANDOS

---

AYU[da]                    también se acepta HEL[p]

DAT[o] fichero  
 DAT[o] ?

COM[entario] fichero  
 COM[entario] NO  
 COM[entario] ?

ESC[cribe]		[DES[pues]]		TOD[os]
				DE xxx A xxx
				xxx

DIR[ectorio] directorio

DIR[ectorio]

REB[obinar]

ADI[os]                    también se acepta EX

---

## \* La subrutina ANAFI

Es la parte fundamental del programa: realiza la conversión de formato y añade, si así se ha indicado, el fichero de comentarios. Es llamada por el comando "ESCRIBE". Lee inicialmente un espectro (registro) ANAE y lo almacena en un vector entero IC de 9x128 elementos (tamaño máximo de un registro ANAE, véase §2). Esto simplificará tanto la transmisión del espectro, almacenado en este tipo de variables (entero\*2) como el de muchos valores de la cabecera. Para otros datos de ésta es más conveniente disponer de variables reales, por lo que hacemos equivalentes las primeras 128 palabras de IC (cabecera del espectro ANAE) a un vector real (real\*4) denominado C, de 64 elementos. Para la correcta transmisión de datos almacenados en otro tipo de variables (carácter, real\*6, etc.) utilizamos variables del tipo adecuado que se hacen equivalentes al segmento de IC donde se encuentran dichos parámetros.

Para transmitir el fichero en formato FITS, creamos una variable carácter (\*2880, es decir, 1440 palabras de 2 bytes) que se irá rellenando con datos a partir del fichero ANAE, hasta construir uno de los registros que forman un fichero FITS. El vector IBUF, de 1440 elementos enteros (\*2), y equivalente a BUF es una variable auxiliar para hacer más cómodas las operaciones de escritura del espectro, así como la puesta a cero. También existe una variable carácter (COMBUF, \*2880) que almacena un fichero de comentarios opcional.

## a. Cabecera

Para escribir la cabecera del fichero FITS (la mayor parte de la subrutina trata de esta operación) utilizamos una variable carácter, L, de 80 bytes que representará una de las cadenas en las que puede dividirse la cabecera de un fichero en este formato (véase §3).

Las líneas que tienen información fija se escriben directamente sobre L y se vuelcan en BUF mediante la subrutina LWRITE. Por ejemplo, la primera línea es invariable, de modo que escribimos

```
SIMPLE =                               T           / comentario
```

Las que tienen datos que dependen de la observación y que se encuentran, por tanto, almacenados en IC, C, o alguna variable auxiliar equivalente, se escriben en dos fases. Primero se crea un "molde" que consta de todos los elementos constantes de la cadena: el nombre de la variable, un signo

igual, una diagonal, comentarios. Por ejemplo

BSCALE = / comentario

Luego se llama a la variable que contiene la información del espectro ANAE que equivale a esa palabra clave (para ver la relación entre las variables FITS y ANAE, ver Apéndice IV) y tras un posible cambio de unidades, se escribe sobre el molde rellenando algunos de los espacios dejados en blanco. En el ejemplo anterior, escribiendo con el formato adecuado, obtenemos

BSCALE = .2409803700000E-04 / comentario

Algunos valores requieren mayor cálculo, como el nombre de la línea y la fecha. Una subrutina reconoce, a partir de la frecuencia, la transición correspondiente y otra calcula la fecha del día en curso.

Una vez completada la variable L, se vuelca en BUF mediante una llamada a la subrutina LWRITE. En caso de haberse completado el registro, la propia LWRITE lo escribe en la cinta y pone BUF a cero (en carácter nulo) mediante una llamada a la subrutina vectorial VMOV.

En caso de introducirse un fichero de comentarios, éste se almacena inicialmente en una variable llamada COMBUF, de 2880 caracteres y se escribe sobre BUF inmediatamente antes de la última cadena (que tiene la clave END). El hecho de que el fichero de comentarios se almacene en una variable supone una limitación de su tamaño, que se ha fijado en 36 líneas de 72 caracteres cada una.

Tras escribir en L la última línea de la cabecera y volcarla sobre BUF, se escribe un registro en la cinta magnética, aunque pueda estar incompleto (el resto en blanco) ya que el siguiente deberá contener el espectro.

#### b. Espectro

Tras los registros de la cabecera, escribimos un único registro que contiene los valores de la intensidad para cada canal. Como se vió en los apartados 2 y 3, la forma de almacenar dichos valores es semejante en los formatos ANAE y FITS, por lo que no tenemos más que igualar la variable equivalente IBUF con los elementos adecuados de IC (cambiando previamente el valor entero asociado a los canales que se eliminan), y escribir IBUF (o BUF, que es equivalente) en la cinta. Seguidamente se escribe un fin de fichero, indicación de que los próximos registros, si los hay, corresponderán a la cabecera del siguiente espectro.

Si se ha terminado el fichero de datos (o ya se han transmitido todos los espectros que se indicaron en la sentencia "ESCRIBE") grabamos un segundo fin de fichero, constituyendo un fin de cinta. Esta marca es reconocible, y tras haber rebobinado la cinta, es posible localizarla para escribir a continuación. Si se siguen escribiendo espectros (en esta o en una posterior ejecución del programa), antes de hacerlo, la cinta retrocederá un registro (el segundo fin de fichero) para grabar a continuación.

La versión actual de ANAFI no permite la transmisión de espectros con dos vías, de manera que si se presenta un espectro con esas características, se perderá la segunda vía, convirtiéndose a formato FITS sólo la primera.

Por cada espectro transcrito se escribirá, por la impresora, una línea que indique algunas de sus características. Por ejemplo:

NUM	FUENTE	OFFX	OFFY	IDEN	FECHA	LINEA
1	AFGL6366(0,0)	0	0	5464	88/ 2/18	CS(1-0)
2	AFGL6366(0,2)	0	120	5578	88/ 2/18	CS(1-0)
3	AFGL6366(-2,0)	-120	0	5636	88/ 2/18	CS(1-0)

Este listado contendrá un resumen de todo lo escrito en la cinta. Nótese que la longitud del nombre de la fuente puede llegar a ser de hasta 20 caracteres (la máxima permitida por ANAE) y no de los 12 de FITS. Si la longitud del nombre supera esta cifra, en la transcripción se perderán algunos caracteres quedando al menos el nombre completo en el listado de la impresora.

Apéndice I. Fichero en formato ANAE

Nombre de la fuente :	5140(0,1)
Fecha de observacion :	870206
T.U. inicial :	11:20:32.0
Ascension recta :	22:17:42.0
Declinacion :	63: 4: 45.0
Identificador :	3882
Numero de canales :	128
Numero de vias :	1
Banda lateral :	5
Frecuencia en reposo :	48990963
Anchura de los canales :	.050
Canal raya via 1 :	64
Canal raya via 2 :	64
Tiempo integracion via 1 :	1:45:20.0
Tiempo integracion via 2 :	0:52:40.0
Numero espectros via 1 :	40
Numero espectros via 2 :	0
Temperatura receptor via 1 :	742
Temperatura receptor via 2 :	742
Temperatura calibrador via 1 :	620.612
Temperatura calibrador via 2 :	620.612
Velocidad LSR :	-4.0
Velocidad hel. :	0.0
Tiempo sidereo inic. :	20: 9:29.4
Elevacion final prom. via 1 :	64:41:53.7
Elevacion final prom. via 2 :	64:41:53.7
Acimut final prom. via 1 :	17:17:29.5
Acimut final prom. via 2 :	17:17:29.5
Angulo paralactico prom. via 1 :	-149: -1:-15.9
Angulo paralactico prom. via 2 :	-149: -1:-15.9
Opacidad :	.180
Correccion a la opacidad :	0.000
Temperatura ambiente :	300.00
Temperatura atmosfera :	203.00
Ganancia intermedia :	.500
Acoplamiento a la atmosfera :	.900
F1 :	30.9694079077
F2 :	0.0000000000
Armonico N1 :	25
Armonico N2 :	12
Valor mudo :	-33333.
Resolucion :	.3060
Sigma via 1 :	.1485
Sigma via 2 :	.2301
Factor de escala via 1 :	.2410E-04
Factor de escala via 2 :	.2732E-04
Desplazamiento en via 1 :	.3937E+00
Desplazamiento en via 2 :	.2665E+00
Offset A.R. :	0
Offset DEC. :	60

  

-22215	-16999	-12331	-21340	-15499	-529	-6991	-23809	-8074
-32768	-22302	-18087	-23774	-7506	-18179	-10781	-15736	-22136
-19181	-10302	-6812	-22808	-29377	-29616	-23440	-23054	-23161
-21655	-19352	-20198	-20102	-9163	-8960	-14955	-18849	-17875
-3426	-4795	-16488	-12593	-10868	-14145	-12636	-21867	-13042
-18820	-9376	-12737	-11980	-14696	-12305	-19889	-18318	-15566
-18128	-11751	-21975	-13525	-21928	-17776	-16912	-14764	-26662
-23921	-24807	-14217	-10438	-11986	-3527	-4791	-3387	13148
32765	28462	21263	23874	26548	17780	5732	-2913	-11003
-13831	-18680	-29842	-22293	-17289	-27251	-18130	-29479	-23376
-18976	-13326	-7570	-10027	-15032	-13791	-21605	-10276	-29365
-12932	-13360	-32765	-15466	-8746	-13401	-11774	DETA	DETA

## Apéndice II. Fichero en formato FITS

```

SIMPLE =                               T           / If simple=T, FITS standard format
BITPIX =                               16         / Bits per pixel
NAXIS  =                               4           / Number of axis
NAXIS1 =                               128        / Number of channels
NAXIS2 =                               1           / Number of A.R.
NAXIS3 =                               1           / Number of DEC.
NAXIS4 =                               1           / Stokes parameters
BSCALE = .24098037000000E+04           / Channel value=tape value*bscale+bzero
BZERO  = .39367443000000E+00           /
DATAMIN = -.39597005000000E+00         / Datamin = -32768*bscale+bzero
DATAMAX = .11832707000000E+01         / Datamax = 32766*bscale+bzero
BUNIT  = 'K'                            / Brightness units
CTYPE1 = 'FREQ'                          /
CRVAL1 = 0.00000000000000E+00           / Offset frequency
CDELTA1 = .50000000000000E+05           / Frequency resolution
CRPIX1 = .64000000000000E+02           / Reference channel
CTYPE2 = 'RA'                            /
CRVAL2 = .33442493000000E+03           / R.A. origin (degrees)
CDELTA2 = .00000000000000E+00           / R.A. offset (degrees)
CRPIX2 = 0.00000000000000E+00           /
CTYPE3 = 'DEC'                            /
CRVAL3 = .63079163000000E+02           / Dec. origin (degrees)
CDELTA3 = .16666666666667E-01           / Dec. offset (degrees)
CRPIX3 = 0.00000000000000E+00           /
CTYPE4 = 'STOKES'                        /
CRVAL4 = 1.00000000000000E+00           /
CDELTA4 = 0.00000000000000E+00           /
CRPIX4 = 0.00000000000000E+00           /
TELESCOP = 'CAY-DAN-IGN'                 /
OBJECT  = 'S140(0,1)'                   /
GLAT   = 0.00000000000000E+00           / Galactic latitude
GLON   = 0.00000000000000E+00           / Galactic longitude
EPOCH  = 0.19500000000000E+04           /
BLANK  = 32767                          / Blanking value = 32767
LINE   = 'CS(1-0)'                      / Line name
RESTFREQ = .48990962000000E+11          / Rest frequency
VLSR   = -.40000000000000E+04          / Velocity of reference channel
DELTA V = -.30597671600000E+03         / Velocity spacing of channels
IMAGFREQ = .46392410112000E+11         / Image frequency
TSYS   = .74200000000000E+03           / Receiver temperature I
OBSTIME = .63200000000000E+04          / Integration time
SCAN-NUM = .38820000000000E+04         / Scan number
TAU-ATM = .18000007000000E+00          / Atmospheric opacity
ORIGIN = 'ANAE-Yebes-HP'                /
DATE   = '8/ 5/88'                      / Date written
DATE-OBS = '8/ 2/87'                    / Date observed
DATE-RED = '8/ 5/88'                    / Date reduced
HISTORY REL .64698257000000E+02         / Telescope elevation
HISTORY RAZ .17291530000000E+02         / Telescope azimuth
HISTORY RUT 11:20:31.973 Universal time at start of observation
HISTORY RST 20: 9:29.449 Sideral time at start of observation
ESTA ES UNA DEMOSTRACION DE UN FICHERO DE COMENTARIOS.
    
```

AQUI PUEDE IR CUALQUIER TIPO DE INFORMACION SOBRE LOS ESPECTROS,

COMO EL TIPO DE CALIBRACION, ETC. SE PUEDEN INCLUIR FORMULAS O

TABLAS DE DATOS:

PARA CS(1-0)

$$TCAL = T_0 + T_1 * \exp(\tau * A)$$

$$T_0 = 492$$

$$T_1 = 99$$

$$\tau = .24$$

ESTOS DATOS IRAN UNIDOS A CADA ESPECTRO FITS.

NO SE ADMITEN MAS DE 36 LINEAS DE COMENTARIO NI LINEAS DE MAS DE 72

CARACTERES YA QUE EN CASO CONTRARIO EL PROGRAMA LAS TRUNCARA AUTOMATICAM

```

END
-22215 -16999 -12331 -21340 -15499 -529 -6991 -23809 -8074
32767 -22302 -18087 -24774 -7506 -18179 -10781 -15736 -22136
-19181 -10302 -6812 -22808 -29377 -29616 -23440 -23054 -23161
-21655 -19352 -20198 -20102 -9163 -8960 -14955 -18849 -17875
-3426 -4795 -16488 -12593 -10868 -14145 -12636 -21867 -13042
-18820 -9376 -12737 -11980 -14696 -12305 -19889 -18318 -15566
-18128 -11751 -21975 -13525 -21928 -17776 -16912 -14764 -26662
-23921 -24807 -14217 -10438 -11986 -3527 -4791 -3387 13148
32765 28462 21263 23874 26548 17780 5732 -2913 -11003
-13831 -18680 -29842 -22293 -17289 -27251 -18130 -29479 -23376
-18976 -13326 -7570 -10027 -15032 -13791 -21605 -10276 -29365
-12932 -13360 -32765 -15466 -8746 -13401 -11776 -9584 -23152
-9576 -9287 -14681 -12658 -13131 -9149 -12385 -13224 -22311
-21535 -14338 -30125 -19763 -17517 -12025 -14748 -22878 -11215
    
```

Apéndice III. Listado del programa ANAFITS

```

0001 $FILES 0,4
0002 C
0003 PROGRAM ANAFITS
0004 C
0005 CHARACTER*60 LINEA,COMBUF*2880
0006 CHARACTER*20 INSTR(6),FICCOM,FICDAT
0007 LOGICAL SINTAXIS,FICHERO,COMENT,EOT
0008 LOGICAL TODOS,INTER,IDEN
0009 C
0010 COMMON/INSTR/INSTR
0011 COMMON/TODOS/SINTAXIS,FICHERO,COMENT,EOT,FICCOM,FICDAT
0012 COMMON/COMEN/COMBUF,ITER
0013 COMMON/ESCR/TODOS,INTER,IDENIN,IDENFIN,IDEN,NUMF
0014 C
0015 10 WRITE(1,*)'ANAFITS)_ '
0016
0017 C Leemos la instruccion y separamos en palabras con RREAD
0018
0019 READ(1,'(A)',END=100)LINEA $ CALL UPPER(LINEA)
0020 CALL RREAD(LINEA,INSTR,6)
0021 C
0022 C Segun sea la instruccion comandada, llamamos a una subrutina
0023 C
0024 IF(INSTR(1)(1:3).EQ.'REF')CALL REFUBINAR
0025 IF(INSTR(1)(1:3).EQ.'DAT')CALL DATO
0026 IF(INSTR(1)(1:3).EQ.'COM')CALL COMENTARIO
0027 IF(INSTR(1)(1:3).EQ.'ADI')CALL ADIOS
0028 IF(INSTR(1)(1:3).EQ.'EX' )CALL ADIOS
0029 IF(INSTR(1)(1:3).EQ.'ESC')CALL ESCRIBE
0030 IF(INSTR(1)(1:3).EQ.'AYU')CALL AYUDA
0031 IF(INSTR(1)(1:3).EQ.'DIR')CALL DIRECTORIO
0032 IF(INSTR(1)(1:3).EQ.'HEL')CALL AYUDA
0033 IF(INSTR(1)(1:3).EQ.' ')GOTO 10
0034 IF(.NOT.SINTAXIS)CALL ERROR
0035 C
0036 SINTAXIS=.FALSE. ! SINTAXIS indica si el comando existe
0037 GOTO 10
0038 100 WRITE(1,*) '^D' $ CALL ADIOS
0039 END
0040
0041 *****
0042 SUBROUTINE DIRECTORIO
0043 INTEGER FmpSetWorkingDir,FmpWorkingDir
0044 CHARACTER INSTR(6)*20,MENSAJE*63
0045 LOGICAL SINTAXIS
0046 COMMON /INSTR/INSTR
0047 COMMON/TODOS/SINTAXIS,FICHERO,COMENT,FOT,FICCOM,FICDAT
0048
0049 SINTAXIS=.TRUE.
0050 IF(INSTR(2).EQ.' ') THEN
0051 IF(FmpWorkingDir(MENSAJE).NE.0) GOTO 99
0052 WRITE(1,'(A)') MENSAJE
0053 ELSE
0054 IF(FmpSetWorkingDir(INSTR(2)).NE.0) GOTO 99

```

```

0055         ENDIF $ RETURN
0056
0057 99      CALL FmpError (IERR,MENSAJE) $ WRITE(1,*) MENSAJE(1:30)
0058         END
0059
0060 *****
0061         SUBROUTINE REBOBINAR
0062 *
0063 *      Esta subrutina rebobina la cinta magnetica
0064 *
0065         CHARACTER*20 FICCOM,FICDAT
0066         LOGICAL SINTAXIS,FICHERO,COMENT,EOT
0067 C
0068         COMMON/TODOS/SINTAXIS,FICHERO,COMENT,EOT,FICCOM,FICDAT
0069 C
0070         SINTAXIS=.TRUE.          ! Comando correcto
0071 C
0072         REWIND(8)                ! Rebobina la cinta
0073         EOT=.FALSE.              ! Ya no estamos en el final de la cinta
0074         END
0075
0076 *****
0077         SUBROUTINE DATO
0078 *
0079 *      Subrutina para abrir el fichero de datos. Si '?' el programa
0080 *      indica el nombre del fichero de datos abierto.
0081 *
0082         LOGICAL SINTAXIS,FICHERO,COMENT,EOT
0083         CHARACTER*20 INSTR(6),FICCOM,FICDAT
0084 C
0085         COMMON/INSTR/INSTR
0086         COMMON/TODOS/SINTAXIS,FICHERO,COMENT,EOT,FICCOM,FICDAT
0087 C
0088         SINTAXIS=.TRUE.          ! Comando correcto
0089 C
0090         IF(INSTR(2)(1:1).EQ.'?')THEN
0091             IF(.NOT.FICHERO)WRITE(1,*)'Fichero de datos no abierto'
0092             IF(FICHERO)WRITE(1,*)'Fichero de datos ',FICDAT
0093         ELSE
0094             OPEN(34,FILE=INSTR(2),STATUS='OLD',ERR=9999,Iostat=IOS)
0095             FICHERO=.TRUE.      $ FICDAT=INSTR(2)
0096         ENDIF
0097         RETURN
0098 9999 WRITE(1,*)' ERROR IOS= ',IOS,' al abrir el fichero de datos'
0099         END
0100
0101 *****
0102         SUBROUTINE COMENTARIO
0103 *
0104 *      Subrutina para abrir el fichero de comentarios (opcional) y leerlo.
0105 *      Se almacena en la variable COMBUF. Si 'NO', no se escribieran mas
0106 *      comentarios. Si '?' el programa indica nombre del fichero de comen-
0107 *      tarios abierto.
0108 *      Variables principales

```

```

0109 *          COMBUF      Acumula el contenido del fichero de comentario
0110 *          COMENT      Variable logica. Si =T, hay fichero de coment.
0111 *          FICCOM      Caracter. Contiene el nombre del fichero de com.
0112 *
0113          LOGICAL SINTAXIS,FICHERO,COMENT,EOT
0114          CHARACTER COMBUF*2880
0115          CHARACTER*20 INSTR(A),FICCOM,FICDAT
0116 C
0117          COMMON/INSTR/INSTR
0118          COMMON/TODOS/SINTAXIS,FICHERO,COMENT,EOT,FICCOM,FICDAT
0119          COMMON/COMEN/COMBUF,ITER
0120 C
0121          SINTAXIS=.TRUE.      ! Comando correcto
0122 C
0123          IF(INSTR(2)(1:3).EQ.'NO ')THEN
0124              COMENT=.FALSE.
0125              RETURN
0126          ELSEIF(INSTR(2)(1:1).EQ.'?')THEN
0127              IF(.NOT.COMENT)WRITE(1,*)'Fichero de comentarios no abierto'
0128              IF(COMENT)WRITE(1,*)'Fichero de comentarios ',FICCOM
0129          ELSE
0130              OPEN(26,FILE=INSTR(2),STATUS='OLD',IOSTAT=IOS,ERR=9999)
0131              COMENT=.TRUE.      $ FICCOM=INSTR(2)
0132              DO ITER=1,36
0133                  COMBUF((ITER-1)*80+1:(ITER-1)*80+8)='
0134                  READ(26,'(A)',END=9998,ERR=9997,IOSTAT=IOST)
0135          +          COMBUF((ITER-1)*80+9:80*ITER)
0136              ENDDO
0137          ENDIF
0138          9998  CLOSE(26)
0139          RETURN
0140          9999  WRITE(1,*)' ERROR IOS= ',IOS,' al abrir el fichero de comentarios'
0141          RETURN
0142          9997  WRITE(1,*)' ERROR IOS= ',IOST,' al leer el fichero de comentarios'
0143          END
0144
0145          *****
0146          SUBROUTINE ESCRIBE
0147 *
0148 *          Subrutina para llama a ANAFI de la forma adecuada a cada caso.
0149 *
0150          CHARACTER*20 INSTR(6),FICCOM,FICDAT
0151          CHARACTER*2880 BUF
0152          LOGICAL SINTAXIS,FICHERO,COMENT,EOT
0153          LOGICAL TODOS,INTER,IDEN
0154 C
0155          COMMON/INSTR/INSTR
0156          COMMON/ESCR/TODOS,INTER,IDENTIN,IDENTFIN,IDEN,NUME
0157          COMMON/TODOS/SINTAXIS,FICHERO,COMENT,EOT,FICCOM,FICDAT
0158 C
0159          SINTAXIS=.TRUE.
0160          TODOS=.FALSE. $ INTER=.FALSE. $ IDEN=.FALSE.
0161 C
0162          IF(INSTR(2)(1:4).EQ.'DESP')THEN

```

```

0163         IF (.NOT. EOT) THEN
0164 10        DO WHILE (.TRUE.)
0165           READ(8, '(A)', END=20) BUF           !Buscamos dos fines de fichero
0166           ENDDO                               !seguidos
0167 20        READ(8, '(A)', END=30) BUF
0168           GOTO 10
0169 30        EOT=.TRUE.
0170           ENDIF
0171           INSTR(2)=INSTR(3) * INSTR(3)=INSTR(4) * INSTR(4)=INSTR(5)
0172           INSTR(5)=INSTR(6)
0173         ENDIF
0174         IF (INSTR(2)(1:3).EQ. 'TOD') THEN
0175           TODOS=.TRUE.
0176           CALL ANAFI
0177         ELSEIF (INSTR(2)(1:2).EQ. 'DE') THEN
0178           IF (INSTR(4)(1:3).NE. 'A ' .AND. INSTR(4)(1:3).NE. 'HAS') THEN
0179             WRITE(1,*) ' ERROR ', INSTR(4)(1:4), ' desconocido'
0180             RETURN
0181           ENDIF
0182           INTER=.TRUE.
0183           READ(INSTR(3)(1:5), '(I5') IDENIN
0184           READ(INSTR(5)(1:5), '(I5') IDENFIN
0185           CALL ANAFI
0186         ELSE
0187           IDEN=.TRUE.
0188           READ(INSTR(2)(1:5), '(I5') NUME
0189           CALL ANAFI
0190         ENDIF
0191         END
0192
0193 *****
0194 SUBROUTINE ANAFI
0195
0196 *      Esta subrutina lee un espectro en formato ANAE (un registro)
0197 *      y lo almacena en la variable IC. A partir de esta variable,
0198 *      va construyendo el espectro FITS (varios registros de 2880
0199 *      bytes) de la siguiente manera: la cabecera se va generando
0200 *      linea por linea ( en la variable L) y esta es volcada en BUF.
0201 *      Tras los registros FITS de la cabecera se encuentra el espectro
0202 *      propiamente dicho, en formato analogo al de ANAE, por lo que
0203 *      no requiere modificacion.
0204 *
0205 *      Variables:
0206 *
0207 *      IC(1152)   vector donde se almacena el espectro ANAE leido
0208 *                 sin formato. (1152=9*128, tamanyo maximo del
0209 *                 fichero).
0210 *      C(64)     vector real equivalente a las primeras 128 pala-
0211 *                 bras de IC, para permitir el manejo de los datos
0212 *                 de la cabecera.
0213 *      BUF      variable caracter(*2880) que constituya un re-
0214 *                 gistro del fichero FITS.
0215 *      IBUF(1440) vector entero, equivalente a BUF, utilizado
0216 *                 para poner a cero BUF y transmitir los valores

```

```

0217 *           del espectro.
0218 *           COMBUF   variable caracter(*2880) donde se almacena el
0219 *                   fichero de comentarios.
0220 *           L         caracter(*80) utilizada para ir llenando BUF
0221 *                   con los datos de la cabecera.
0222 *
0223 *
0224 CHARACTER CAR*20, LIN*12, SI*1, BUF*2880, COMBUF*2880, LL*40, L*80
0225 CHARACTER*20 FICCOM, FICDAT
0226 DIMENSION IAUX(1440), IC(1152), C(64), IBUF(1440), IPAR(5), LOCK(2)
0227 INTEGER*4 FRECREST, FRMHZ
0228 LOGICAL SINTAXIS, FICHERO, COMENT, EOT
0229 LOGICAL TODOS, INTER, IDEN, FIN, DENTRO
0230 REAL*6 RESTFREQ, IMAGFREQ
0231 C
0232 COMMON L, BUF
0233 COMMON/A/IDIA, MES, IYEAR
0234 COMMON/TODOS/SINTAXIS, FICHERO, COMENT, EOT, FICCOM, FICDAT
0235 COMMON/COMEN/COMBUF, ITER
0236 COMMON/ESCR/TODOS, INTER, IDENIN, IDENFIN, IDEN, NUME
0237 EQUIVALENCE (IC, C), (C(12), FRECREST), (IC(22), SI), (C, CAR)
0238 +(IBUF, BUF)
0239 *
0240 IF(.NOT.FICHERO)THEN
0241     WRITE(1,*)' ERROR   Fichero de datos no abierto'
0242     RETURN
0243 ENDIF
0244 DENTRO=.FALSE.  # FIN=.FALSE.
0245 CALL LGBUF(IAUX,1440)
0246 LU=1
0247 LL='
0248 C     LOCK(1)=6 # LOCK(2)=8 # NLOCK=2 # CALL LURQ (1,LOCK,NLOCK)
0249 38   WRITE(6,666)
0250 666  FORMAT(2X, 'NUM', 4X, 'FUENTE', 18X, 'OFFX', 4X, 'OFFY', 4X, 'IDEN', 5X,
0251 +'FECHA', 6X, 'LINEA', /)
0252     BACKSPACE(8)  !Si hay dos BUF retrocedemos uno
0253     DO I=1,32700
0254     IF(IFBRK(K).LT.0) GOTO 9999
0255 *
0256 *   Leemos el espectro en formato ANAF y lo almacenamos en IC
0257 *
0258 30   READ(34,ERR=33,IOSTAT=IOS,END=9999)IC
0259 33   IF(IOS.NE.0 .AND. IOS.NE.496) GOTO 99
0260     IF(IDEN.AND.NUME.NE.IC(19))GOTO 30
0261     IF(IDEN.AND.NUME.EQ.IC(19))FIN=.TRUE.
0262     IF(INTER.AND.IDENIN.EQ.IC(19))DENTRO=.TRUE.
0263     IF(INTER.AND.IDENIN.NE.IC(19).AND..NOT.DENTRO) GOTO 30
0264     IF(INTER.AND.IDENFIN.EQ.IC(19))FIN=.TRUE.
0265     WRITE(LU,*) 'inicio espectro #',I,'
0266     L(1:40)='SIMPLE   =           T           //'
0267     L(41:80)=' If simple=f, FITS standard format
0268     CALL LWRITE
0269     L(1:40)='BITPIX   =           16           //'
0270     L(41:80)=' Bits per pixel

```

```

0271      CALL LWRITE
0272      L(1:40)='NAXIS  =          4          //'
0273      L(41:80)=' Number of axis
0274      CALL LWRITE
0275      NAXIS1=IC(20)
0276      L(1:40)='NAXIS1 =          //
0277      WRITE (L(28:30),'(I3)') IC(20)
0278      L(41:80)=' Number of channels
0279      CALL LWRITE
0280      L(1:40)='NAXIS2 =          1          //'
0281      L(41:80)=' Number of A.R.
0282      CALL LWRITE
0283      L(1:40)='NAXIS3 =          1          //'
0284      L(41:80)=' Number of DEC.
0285      CALL LWRITE
0286      L(1:40)='NAXIS4 =          1          //'
0287      L(41:80)=' Stokes parameters
0288      CALL LWRITE
0289      L(1:40)='BSCALE =          //
0290      L(41:80)=' Channel value=tape value*bscale+bzero
0291      WRITE(L(11:30),1111)C(43)
0292      1111  FORMAT(E20.13)
0293      CALL LWRITE
0294      L(1:40)='BZERO  =          //
0295      L(41:80)=LL
0296      WRITE(L(11:30),1111)C(45)
0297      CALL LWRITE
0298      L(1:40)='DATAMIN =          //
0299      WRITE(L(11:30),1111)-32768*C(43)+C(45)
0300      L(41:80)=' Datamin = -32768*bscale+bzero
0301      CALL LWRITE
0302      L(1:40)='DATAMAX =          //
0303      WRITE(L(11:30),1111)32768*C(43)+C(45)
0304      L(41:80)=' Datamax = 32768*bscale+bzero
0305      CALL LWRITE
0306      L(1:40)='BUNIT  = ''K          //
0307      L(41:80)=' Brightness units
0308      CALL LWRITE
0309      L(1:40)='CTYPE1 = ''FREQ      //
0310      L(41:80)=LL
0311      CALL LWRITE
0312      L(1:40)='CRVAL1 = 0.00000000000000E+00 //
0313      L(41:80)=' Offset frequency
0314      CALL LWRITE
0315      L(1:40)='CDEL11 =          //
0316      WRITE(L(11:30),1111)C(13)*1E6
0317      L(41:80)=' Frequency resolution
0318      CALL LWRITE
0319      L(1:40)='CRPIX1 =          //
0320      L(41:80)=' Reference channel
0321      WRITE(L(11:30),1111)IC(27)
0322      CALL LWRITE
0323      L(1:40)='CTYPE2 = ''RA          //
0324      L(41:80)=LL

```

```

0325      CALL LWRITE
0326      L(1:40)='CRVAL2' =
0327      L(41:80)=' R.A. origin (degrees)
0328      WRITE(L(11:30),1111)C(8)*180
0329      CALL LWRITE
0330      L(1:40)='CDELTA2' =
0331      L(41:80)=' R.A. offset (degrees)
0332      WRITE(L(11:30),1111)IC(93)/3600.D0
0333      CALL LWRITE
0334      L(1:40)='CRPIX2' = 0.000000000000000E+00
0335      L(41:80)=LL
0336      CALL LWRITE
0337      L(1:40)='CTYPE3' = 'DEC'
0338      L(41:80)=LL
0339      CALL LWRITE
0340      L(1:40)='CRVAL3' =
0341      L(41:80)=' Dec. origin (degrees)
0342      WRITE(L(11:30),1111)C(9)*180
0343      CALL LWRITE
0344      L(1:40)='CDELTA3' =
0345      L(41:80)=' Dec. offset (degrees)
0346      WRITE(L(11:30),1111)IC(94)/3600.D0
0347      CALL LWRITE
0348      L(1:40)='CRPIX3' = 0.000000000000000E+00
0349      L(41:80)=LL
0350      CALL LWRITE
0351      L(1:40)='CTYPE4' = 'STOKES'
0352      CALL LWRITE
0353      L(1:40)='CRVAL4' = 1.000000000000000
0354      CALL LWRITE
0355      L(1:40)='CDELTA4' = 0.000000000000000
0356      CALL LWRITE
0357      L(1:40)='CRPIX4' = 0.000000000000000
0358      CALL LWRITE
0359      L(1:40)='TELESCOP=' 'CAY-DAN-IGN'
0360      CALL LWRITE
0361      L(1:40)='OBJECT' = ''
0362      WRITE(L(12:23),'(A12)')CAR
0363      CALL LWRITE
0364      L(1:40)='GLAT' = 0.000000000000000E+00
0365      L(41:80)=' Galactic latitude
0366      CALL LWRITE
0367      L(1:40)='GLON' = 0.000000000000000E+00
0368      L(41:80)=' Galactic longitude
0369      CALL LWRITE
0370      L(1:40)='EPOCH' = 0.195000000000000E+04
0371      L(41:80)='
0372      CALL LWRITE
0373      L(1:40)='BLANK' = 32767
0374      L(41:80)=' Blanking value = 32767
0375      CALL LWRITE
0376      L(1:40)='LINE' = ''
0377      CALL LINEA(FRECRESL,LIN)
0378      L(12:23)=LIN

```

```

0379      L(41:80)=' Line name
0380      CALL LWRITE
0381      L(1:40)='RESTFREQ=
0382      WRITE(L(11:30),1111)FRECRES[*1E3
0383      L(41:80)=' Rest frequency
0384      CALL LWRITE
0385      L(1:40)='VLSR      =
0386      WRITE(L(11:30),1111)C(20)*1E3
0387      L(41:80)=' Velocity of reference channel
0388      CALL LWRITE
0389      L(1:40)='DELTAV  =
0390      WRITE(L(11:30),1111)(-1)*C(40)*1E3
0391      L(41:80)=' Velocity spacing of channels
0392      CALL LWRITE
0393      IF (SI.EQ.'S') THEN
0394          IMAGFREQ=FRECRES[*1E3-2*1299.275E6
0395      ELSE
0396          IMAGFREQ=FRECRES[*1E3+2*1299.275E6
0397      ENDIF
0398      L(1:40)='IMAGFREQ=
0399      WRITE(L(11:30),1111)IMAGFREQ
0400      L(41:80)=' Image frequency
0401      CALL LWRITE
0402      L(1:40)='TSYS      =
0403      WRITE(L(11:30),1111)IC(33)      !ES LA TEMPERATURA DEL RECEPTOR
0404      L(41:80)=' Receiver temperature !
0405      CALL LWRITE
0406      L(1:40)='OBSTIME  =
0407      WRITE(L(11:30),1111)IC(29)
0408      L(41:80)=' Integration time
0409      CALL LWRITE
0410      L(1:40)='SCAN-NUM=
0411      WRITE(L(11:30),1111)IC(19)
0412      L(41:80)=' Scan number
0413      CALL LWRITE
0414      L(1:40)='TAU-ATM  =
0415      WRITE(L(11:30),1111)C(29)
0416      L(41:80)=' Atmospheric opacity
0417      CALL LWRITE
0418      C      L(1:40)='NPHASE  =          2
0419      C      CALL LWRITE
0420      C      L(1:40)='DELTA F1 = -0.50000000000000E+07
0421      C      CALL LWRITE
0422      C      L(1:40)='PTIME1  =  0.37500000000000E+02
0423      C      CALL LWRITE
0424      C      L(1:40)='WEIGHT1  =  0.10000000000000E+01
0425      C      CALL LWRITE
0426      C      L(1:40)='DELTA F2 =  0.50000000000000E+07
0427      C      CALL LWRITE
0428      C      L(1:40)='PTIME2  =  0.37500000000000E+02
0429      C      CALL LWRITE
0430      C      L(1:40)='WEIGHT2  = -0.10000000000000E+01
0431      C      CALL LWRITE
0432      L(1:40)='ORIGIN   =  'ANAF-Yebes-HP

```

```

0433      L(41:80)='
0434      CALL LWRITE
0435      CALL FECHA
0436      L(1:40)='DATE      = ' / / '
0437      WRITE(L(12:13),'(I2)')IDIA
0438      WRITE(L(15:16),'(I2)')MES
0439      WRITE(L(18:19),'(I2)')IYEAR
0440      L(41:80)=' Date written
0441      CALL LWRITE
0442      L(1:40)='DATE-OBS=' / / '
0443      IY=C(6)/10000
0444      IM=(C(6)-IY*10000.)/100
0445      ID=C(6)-(IY*10000.+IM*100)
0446      WRITE(L(12:13),'(I2)')ID
0447      WRITE(L(15:16),'(I2)')IM
0448      WRITE(L(18:19),'(I2)')IY
0449      L(41:80)=' Date observed
0450      CALL LWRITE
0451 *      La fecha de reduccion es igual a la corriente
0452      L(1:40)='DATE-RED=' / / '
0453      WRITE(L(12:13),'(I2)')IDIA
0454      WRITE(L(15:16),'(I2)')MES
0455      WRITE(L(18:19),'(I2)')IYEAR
0456      L(41:80)=' Date reduced
0457      CALL LWRITE
0458      L(1:42)='HISTORY RFL
0459      WRITE(L(13:32),1111)C(23)*160
0460      L(43:80)=' Telescope elevation
0461      CALL LWRITE
0462      HISTORAZ=C(25)*180
0463      IF(C(25).LT.0) HISTORAZ=360+HISTORAZ
0464      L(1:42)='HISTORY RAZ
0465      WRITE(L(13:32),1111)HISTORAZ
0466      L(43:80)=' Telescope azimuth
0467      CALL LWRITE
0468      L(1:25)='HISTORY ROT      :
0469      IHU=C(7)*12
0470      IMU=(C(7)*12-IHU)*60
0471      SU=((C(7)*12-IHU)*60-IMU)*60
0472      WRITE(L(14:15),'(I2)')IHU
0473      WRITE(L(17:18),'(I2)')IMU
0474      WRITE(L(20:25),'(F6.3)')SU
0475      L(26:80)=' Universal time at start of observation
0476      CALL LWRITE
0477      L(1:25)='HISTORY RST      :
0478      IHS=C(22)*12
0479      IMS=(C(22)*12-IHS)*60
0480      SS=((C(22)*12-IHS)*60-IMS)*60
0481      WRITE(L(14:15),'(I2)')IHS
0482      WRITE(L(17:18),'(I2)')IMS
0483      WRITE(L(20:25),'(F6.3)')SS
0484      L(26:80)=' Sideral time at start of observation
0485      CALL LWRITE
0486      IF(COMENT)THEN

```

```

0487      DO KIK=1,ITER-1
0488          L=COMBUF((KIK-1)*80+1;KIK*80)
0489          CALL LWRITE
0490      ENDDO
0491      ENDIF
0492      L(41:80)='
0493      ! L(1:40)='BEAMEFF = 0.28      //
0494      ! CALL LWRITE
0495      ! FORWEFF=C(34)
0496      ! L(1:40)='FORWEFF =      //
0497      ! WRITE(L(12:15),'(F4,2)')FORWEFF
0498      ! CALL LWRITE
0499      ! GAINIMAG=C(33)
0500      ! L(1:40)='GAINIMAG=      //
0501      ! WRITE(L(11:30),1111)GAINIMAG
0502      ! CALL LWRITE
0503      L(1:42)='END
0504      CALL LWRITE
0505      *
0506      *      El espectro se transmite directamente sin ninguna
0507      *      modificacion salvo la sustitucion del valor nudo
0508      *
0509      DO II=1,NAXIS1
0510          IVALOR=IC(II+128)
0511          IF(IVALOR.FQ.-32768) THEN * IBUF(II)=32767 !cambio de nudos
0512          ELSE * IBUF(II)=IVALOR * ENDIF
0513      ENDDO
0514      *
0515      *      Escribimos el espectro en la cinta, y tras el, un fin
0516      *      de fichero
0517      *
0518      WRITE(8,'(A)') BUF
0519      ENDFILE 8
0520      CALL VMDV(0.,0,IBUF,1,720)
0521      NORD=NORD+1
0522      WRITE(6,555)NORD,CAR,IC(93),IC(94),IC(19),IY,IN,ID,LIN
0523      555  FORMAT(2X,I3,4X,A,4X,I4,4X,I4,4X,I4,4X,I2,'//',I2,'//',I2,4X,A)
0524      IF(FIN)GOTO 9999
0525      ENDDO      ! Vamos a por el proximo fichero ANAF
0526      99  REWIND(34)
0527      WRITE(1,*)' ERROR IOS= ',IOS,' al leer el espectro ANAE'
0528      C      CALL LURQ (0,LOCK,NLOCK)
0529      9999  REWIND(34)
0530      WRITE(1,*)
0531      ENDFILE(8)      ! Dos EOF son un fin de cinta.
0532      EOT=.TRUE.
0533      RETURN
0534      END
0535      *****
0536      *      Subrutina para volcar, secuencialmete, la linea L en BUF.
0537      *      y en caso de haberse completado BUF se escribe en la
0538      *      unidad 8. Permite, si se lanza el programa adecuadamente,
0539      *      escribir por la impresora el contenido de la linea copiada.
0540      *

```

```

0541      SUBROUTINE LWRITE
0542      DIMENSION IBUF(1440)
0543      CHARACTER L*80,IBUF*2880
0544      COMMON L,BUF
0545      EQUIVALENCE(BUF,IBUF)
0546      DATA K/1/
0547 7373  FORMAT(' ',A)
0548      BUF(K:K+79)=L $ K=K+80
0549      IF(K.GT.2880.OR.L(1:3).EQ.'END')THEN ! Si hemos llenado BUF,
0550 *                                     lo escribe
0551          WRITE(B,'(A)')BUF
0552          CALL VMOV(0.,0,IBUF,1,720)
0553          K=1
0554      ENDIF
0555      END
0556 *****
0557 *          Calcula, a traves de una subrutina del sistema, el dia
0558 *          en curso y lo convierte a cifras
0559 *
0560      SUBROUTINE FECHA
0561      COMMON/A/IDIA,MES,IYEAR
0562      CHARACTER B*30
0563      DIMENSION BR(8),TMES(12)
0564      EQUIVALENCE (BR,B)
0565      DATA TMES/48HJAN.FEB.MAR.APR.MAY JUNEJULYAUG.SEPTOCT.NOV.DEC./
0566      CALL FTIME (BR) !toma fecha actual
0567      DO I=1,12 $ IF (BR(6).EQ.TMES(I)) MES=I $ ENDDO
0568      READ (B(17:18),*) IDIA !toma dia del mes
0569      READ (B(29:30),*)IYEAR
0570      END
0571 *****
0572 *          Subrutina para asignar el nombre a la transicion a partir
0573 *          de la frecuencia. Trabajamos en MHz para evitar problemas
0574 *          de redondeo
0575 *
0576      SUBROUTINE LINEA(FR,LIN)
0577      CHARACTER*12 LIN
0578      INTEGER*4 FR ,FRMHZ
0579      FRMHZ=(FR+500)/1E3
0580      LIN='*'
0581      IF (FRMHZ.EQ.48991) LIN='CS(1-0)'
0582      IF (FRMHZ.EQ.48207) LIN='C34S(1-0)'
0583      IF (FRMHZ.EQ.45379) LIN='C25(4,3-3,2)'
0584      IF (FRMHZ.EQ.46246) LIN='C35(8-7)'
0585      IF (FRMHZ.EQ.42821) LIN='S10-v=2'
0586      IF (FRMHZ.EQ.43122) LIN='S10-v=1'
0587      IF (FRMHZ.EQ.42519) LIN='S10-v=3'
0588      IF (FRMHZ.EQ.45265) LIN='HC5N'
0589      IF (FRMHZ.EQ.45490) LIN='HC3N(5-4)'
0590      IF (FRMHZ.EQ.42880) LIN='29S10-v=0'
0591      IF (FRMHZ.EQ.115271) LIN='CO(1-0)'
0592      IF (FRMHZ.EQ.230542) LIN='CO(2-1)'
0593      IF (FRMHZ.EQ.110201) LIN='13CO(1-0)'
0594      END

```

```

0595 *****
0596     SUBROUTINE UPPER(LINEA)
0597 *     Pasamos a mayusculas todos los caracteres de la variable LINEA
0598     CHARACTER*(*) LINEA
0599     DO I=1,LEN(LINEA) $ K=ICHAR(LINEA(I:I))
0600         IF(IAND(K,140B).EQ.140B)LINEA(I:I)=CHAR(K-40B)
0601     ENDDO $ END
0602 *****
0603     SUBROUTINE ADIOS
0604 *     Despedimos el programa
0605     STOP
0606     END
0607 *****
0608     SUBROUTINE ERROR
0609 *     Indica que no reconoce un comando (SINTAXIS=FALSE)
0610     CHARACTER*20 INSTR(6)
0611     COMMON/INSTR/INSTR
0612     WRITE(1,10)INSTR(1)(1:7)
0613 10    FORMAT(' ERROR Comando   .A.   desconocido')
0614     END
0615 *****
0616     SUBROUTINE AYUDA
0617 *     Da acceso al fichero de informacio'n
0618     CHARACTER*20 INSTR(6)
0619     CHARACTER*80 TEXTO
0620     LOGICAL DENTRO,SINTAXIS
0621     COMMON/TODOS/SINTAXIS
0622     COMMON/INSTR/INSTR
0623     INDICE=0 $ DENTRO=.FALSE. $ SINTAXIS=.TRUE.
0624     IF(INSTR(2)(1:3).EQ.' ')INDICE=1
0625     IF(INSTR(2)(1:3).EQ.'AYU')INDICE=2
0626     IF(INSTR(2)(1:3).EQ.'HEL')INDICE=2
0627     IF(INSTR(2)(1:3).EQ.'DAT')INDICE=3
0628     IF(INSTR(2)(1:3).EQ.'COM')INDICE=4
0629     IF(INSTR(2)(1:3).EQ.'ESC')INDICE=5
0630     IF(INSTR(2)(1:3).EQ.'REB')INDICE=6
0631     IF(INSTR(2)(1:3).EQ.'ADI')INDICE=7
0632     IF(INSTR(2)(1:2).EQ.'EX') INDICE=7
0633     IF(INSTR(2)(1:2).EQ.'WD') INDICE=8
0634     IF(INDICE.EQ.0)WRITE(1,*)'ANAFITS) Comando desconocido'
0635     OPEN(15,FILE='/MT/PROG/ANAFITS.HELP')
0636     DENTRO=.FALSE.
0637     DO WHILE(.TRUE.)
0638         READ(15,'(A)')TEXTO
0639         READ(TEXTO(2:2),'(I1)')IKI
0640         IF(IKI.EQ.INDICE)DENTRO=.TRUE.
0641         IF(IKI.EQ.INDICE+1)GOTO 155
0642         IF(DENTRO)WRITE(1,'(A)')TEXTO(3:80)
0643     ENDDO
0644 155    CLOSE(15)
0645     RETURN
0646     END
0647 *****
0648     SUBROUTINE RREAD(LINEA,PAL,NMAX)

```

```

0649 * Identifica las palabras que constituyen un comando
0650 * Buscamos los blancos. El comando entra en la variable LINEA, y
0651 * las palabras que lo constituyen se devuelven en el vector PAL
0652 CHARACTER*60 LINEA
0653 CHARACTER*20 PAL(NMAX)
0654 DO II=1,NMAX $ DO JJ=1,20 $ PAL (II)(JJ:JJ)=' ' $ ENDDO $ ENDDO
0655 M=1
0656 DO I=1,60
0657 IF(I.EQ.60)RETURN
0658 IF(LINEA(I:I).EQ.' ')GOTO 10
0659 J=I
0660 DO K=J+1,60
0661 IF(LINEA(K:K).NE.' ')GOTO 9
0662 PAL(M)=LINEA(J:K-1)
0663 M=M+1
0664 IF(M.GT.NMAX)RETURN
0665 I=K-1
0666 GOTO 10
0667 9 ENDDO
0668 10 ENDDO
0669 END

```

## Apéndice V. Programas auxiliares

Presentamos aquí dos programas auxiliares para el manejo de ficheros FITS que funcionan en el HP1000: el programa CREAMFITS, que construye ficheros en este formato a partir de un fichero del disco que sirve como modelo, y LEEFITS, que lee de la cinta magnética un fichero en formato FITS y lo escribe en el disco separando la cabecera en líneas.

### CREAFITS

Este programa parte de un fichero de nombre FICHERO.FITS, donde se encuentra la cabecera y el espectro, la primera troceada en cadenas de 80 caracteres, y el segundo en enteros \*2. Este fichero se puede editar y modificar a voluntad, siempre con la precaución de no cambiar el formato. Se pueden añadir líneas de comentario, eliminar otras, etc. Una vez que ya tenemos el fichero con los datos deseados el programa CREAMFITS lo leerá y escribirá los registros de 2880 caracteres en la cinta magnética. Este programa puede actuar como una subrutina de otro que genere espectros sintéticos y de esta manera los espectros pueden ser analizados con los programas estándar.

Existe una versión análoga que funciona en el MicroVAX, con la particularidad de que hace uso de una subrutina para la trasposición de bytes.

### LEEFITS

Este programa lee ficheros FITS en la cinta magnética y crea un fichero de nombre ESPECTRO.FITS en el disco, que contiene la cabecera separada en líneas, y el espectro en enteros. El formato de escritura es igual que el de lectura de CREAMFITS, de manera que podemos partir de un fichero FITS en la cinta, escribirlo en el disco (con LEEFITS), modificarlo, y escribir la nueva versión en la cinta, otra vez en formato adecuado (con CREAMFITS).

/MT/PROG/CREAFITS.FTN:1:4:10:3

```
0001 *
0002 * Programa para crear ficheros FITS. Partimos de un fichero de
0003 * nombre FICHERO.FITS donde se encuentra la cabecera en forma
0004 * de lineas, y el espectro en enteros. Leemos las lineas y creamos
0005 * los registros que constituyen el fichero FITS. El resultado se
0006 * graba en cinta magnetica.
0007 *
0008 $FILES 0,1
0009 PROGRAM CREAMFITS
0010 CHARACTER*2880 BUF
0011 DIMENSION IBUF(1440),IAUX(1440)
0012 EQUIVALENCE(BUF,IBUF)
0013 CALL LGBUF(IAUX,1440)
0014 OPEN(22,FILE='FICHERO.FITS',STATUS='OLD',ERR=9999,IOSTAT=IOS)
0015 I=1 $ IND=0
0016 DO WHILE (.TRUE.)
0017 IF(IND.EQ.0)THEN ! IND=0 si se trata de la cabecera del espectro
0018 READ(22,'(A)',END=9998)BUF((I-1)*80+1:80*I)
0019 IF(I.EQ.36.OR.BUF((I-1)*80+1:(I-1)*80+3).EQ.'END')THEN
0020 WRITE(8,'(A)')BUF
0021 IF(BUF(80*(I-1)+1:80*(I-1)+3).EQ.'END')IND=1
0022 CALL VMOV(0.,0,IBUF,1,720)
0023 I=1
0024 ELSE
0025 I=I+1
0026 ENDIF
0027 ELSE
0028 READ(22,'(160(9I8,/))',END=9998)(IBUF(I:1),I:1,1440)
0029 ENDIF
0030 ENDDO
0031 9998 WRITE(8,'(A)')BUF
0032 ENDFILE 8
0033 STOP
0034 9999 WRITE(1,*)' Error',IOS,' al abrir el fichero'
0035 END
```

/MT/PROG/LEEFITS.FTN:1:4:4:28

```
0001 *
0002 * Programa para leer de la cinta un fichero tipo FITS
0003 *
0004 $files 0,1
0005 program leefits
0006 character*2880 buf
0007 dimension iaux(1440),ic(128)
0008 equivalence(ic,buf)
0009 call lgbuf(iaux,1440)
0010 open(22,file='espectro.fits',status='un')
0011 do while(.true.)
0012 10 read(8,'(A)',end=33)buf
0013 if(indic.eq.0)then
0014 do j=1,36
0015 c write(22,'(A)')buf(80*(j-1)+1:80*j-3)
0016 * Eliminaamos caracteres por razones de espacio
0017 write(6,'(1x,A)')buf(80*(j-1)+1:80*j)
0018 * Eliminaamos caracteres por razones de espacio
0019 if(buf(80*(j-1)+1:80*(j-1)+3).eq.'END')then
0020 indic=1
0021 goto 10
0022 endif
0023 enddo
0024 else
0025 write(6,'(9i8)')ic
0026 c write(22,'(9i8)')ic
0027 indic=0
0028 endif
0029 enddo
0030 33 stop
0031 end
```