

## ANEXOS

- Anexo 1.- Enrutamiento Internacional.
- Anexo 2.- Códigos Razón de Desconexión.
- Anexo 3.- Comandos de Consola.
- Anexo 4.- Descripción de las Instrucciones.
- Anexo 5.- Subrutinas del Programa REC.BIN.
- Anexo 6.- Información del utilitario "K".
- Anexo 7.- Información contenida en la memoria al "correr" el programa REC.BIN.
- Anexo 8.- Muestras.



ANEXO 1

## ANEXO

## ENRUTAMIENTO INTERNACIONAL

PAIS DE DESTINO	CODIGO DE ACCESO	PRIMERA RUTA	1. ALTERNATIVA	2. ALTERNATIVA
ABU DHABI	0893 0894	WUI - USA	ITT - USA	ITC - ITALIA
AFGANISTAN	079 <del>0892</del>	ITC - ITALIA	RCA - USA	ITT - USA
ALASKA	0200	RCA - USA	ITT - USA	ITC - ITALIA
ALEMANIA ORIENTAL	069	WUI - USA	ITT - USA	RCA - USA
ALBANIA	0604	ITC - ITALIA	RCA - USA	ITT - USA
ALEMANIA OCCIDENTAL	041	WUI - USA	ITT - USA	RCA - USA
ALTO VOLTA	0978	ITC - ITALIA	RCA - USA	ITT - USA
ANDORRA	0590	ITT - USA	RCA - USA	ITC - ITALIA
ANGOLA	0991	ITC - ITALIA	RCA - USA	ITT - USA
ANGJILA	0391	RCA - USA	ITT - USA	ITC - ITALIA
ANTIGUA	0393	ITT - USA	RCA - USA	ITC - ITALIA
ANTILLAS HOLANDESAS	0390	WUI - USA	ITT - USA	RCA - USA
ARABIA SAUDITA	0495	ITC - ITALIA	RCA - USA	ITT - USA
ALGERIA	0408	RCA - USA	ITT - USA	WUI - USA
ARGENTINA	033	ENTEL ARGENTINA	WUI - USA	ITT - USA
ASCENSION	0939	RCA - USA	ITT - USA	WUI - USA
AUSTRALIA	071	WUI - USA	ITT - USA	RCA - USA
AUSTRIA	047	WUI - USA	ITT - USA	RCA - USA
BAHAMAS	0297	TRT - USA	ITT - USA	ITC - ITALIA
BAHREIN	0490	WUI - USA	ITT - USA	RCA - USA
BANGLADESH	0780	RCA - USA	ITT - USA	WUI - USA
BARBADOS	0392	RCA - USA	ITT - USA	WUI - USA
BELGICA	046	WUI - USA	ITT - USA	ITC - ITALIA
BELIZE	0371	RCA - USA	ITT - USA	ITC - ITALIA
BENIN	0972	ITC - ITALIA	RCA - USA	ITT - USA
BERMUDAS	0290	WUI - USA	ITT - USA	RCA - USA
BIRMANIA	083	ITT - USA	RCA - USA	ITC - ITALIA
BOLIVIA	0309	WUI - USA	ITT - USA	RCA - USA
BOSTWANA	0962	RCA - USA	ITT - USA	ITC - ITALIA
BRASIL	038	EMBRATEL BRASIL	ITT - USA	RCA - USA
BRUNEI	0809	RCA - USA	ITT - USA	WUI - USA
BULGARIA	067	RCA - USA	ITT - USA	WUI - USA
BURUNDI	0903	RCA - USA	ITT - USA	WUI - USA
CABO VERDE	0993	RCA - USA	ITT - USA	WUI - USA
CAIMANES	0293	RCA - USA	ITT - USA	WUI - USA
CAMERUN	0970	RCA - USA	ITT - USA	WUI - USA
CANADA	021	- CANADA	WUI - USA	RCA - USA
CANADA (TWX)	026	TWX - CANADA	WUI - USA	RCA - USA
AFRICA CENTRAL	0971	RCA - USA	ITT - USA	ITC - ITALIA
CHAD	0976	RCA - USA	ITT - USA	ITC - ITALIA
CHECOSLOVAQUIA	066	WUI - USA	ITT - USA	RCA - USA
CHILE (TELEX). (TEXCON)	0342 0346	WUI - USA	ITT - USA	RCA - USA
CHILE (TRANS RADIO)	0343	RCA - USA	ITT - USA	WUI - USA
CHILE (ITT)	0344	ITT - USA	RCA - USA	ITC - ITALIA
CHINA	085	RCA - USA	ITT - USA	WUI - USA
	0890			
CHIPRE	0605	ITC - ITALIA	RCA - USA	ITT - USA

PAIS DE DESTINO	CODIGO DE ACCESO	PRIMERA RUTA	1. ALTERNATIVA	2. ALTERNATIVA
COLOMBIA	035	TELECOM COLOMBIA	ITT - USA	RCA - USA
COMORAS	0994	ITC - ITALIA	RCA - USA	ITT - USA
CONGO	0981	ITC - ITALIA	RCA - USA	ITT - USA
COOK	0772	RCA - USA	ITT - USA	WUI - USA
COREA	0801	RCA - USA	ITT - USA	WUI - USA
COSTA MARFIL	0983	ITC - ITALIA	RCA - USA	ITT - USA
COSTA RICA	0376	TRT - USA	WUI - USA	ITT - USA
CUBA	028	ITC - ITALIA	RCA - USA	ITT - USA
DINAMARCA	055	WUI - USA	ITT - USA	RCA - USA
DJIBOUTI	0979	ITC - ITALIA	RCA - USA	ITT - USA
DOMINICA	0394	ITT - USA	RCA - USA	ITC - ITALIA
DOMINICANA R.	0201	RCA - USA	ITT - USA	WUI - USA
DOMINICANA REP.	0202	ITT - USA	RCA - USA	ITC - ITALIA
DUBAI	0892	ITC - ITALIA	RCA - USA	ITT - USA
EGIPTO	091	WUI - USA	ITT - USA	RCA - USA
EL SALVADOR	0373	WUI - USA	ITT - USA	RCA - USA
ESPAÑA	052	CNT - ESPAÑA	WUI - USA	ITT - USA
EE. UU.	02301	TRT - USA 17%	15	
TWX	02302	RCA - USA 25%	28	
	02303	ITT - USA 36%	25	
	02304	WUI - USA 22%	2	
	02305			
	02306			
	02307			
	02308			
	02309			
EE. UU./TRT	0231	TRT - USA	ITC - ITALIA	
EE. UU./RCA	0232	RCA - USA	WUI - USA	
EE.UU./ITT	0234	ITT - USA	WUI - USA	0235
EE.UU./WUI	0236	WUI - USA	ITC - ITALIA	
EE.UU./TWX	025510	ITT - USA	WUI - USA	
	025710			
	025810			
	025910			
ETIOPIA	0980	ITC - ITALIA	RCA - USA	ITT - USA
FALKLAND	0306	ITC - ITALIA	RCA - USA	ITT - USA
FEROE	0502	RCA - USA	ITT - USA	ITC - ITALIA
FIJI	0701	ITT - USA	RCA - USA	WUI - USA
FILIF. C&W	0751	ITC - ITALIA	RCA - USA	ITT - USA
FILIP/RCA	0752	RCA - USA	ITT - USA	ITC - ITALIA
FILIP/GRF	0754	ITT - USA	RCA - USA	ITC - ITALIA
FILIP/ETP	0756	TRT - USA	WUI - USA	RCA - USA
FILIP/PTT	0758	TRT - USA	WUI - USA	RCA - USA
FINLANDIA	057	WUI - USA	ITT - USA	RCA - USA
FRANCIA	042	- FRANCIA	WUI - USA	RCA - USA
GABONESA	0973	RCA - USA	ITT - USA	ITC - ITALIA
GAMBIA	0996	RCA - USA	ITT - USA	ITC - ITALIA
GHANA	094	RCA - USA	ITT - USA	WUI - USA
GIBRALTAR	0405	RCA - USA	ITT - USA	ITC - ITALIA
GRANADA	0395	ITT - USA	RCA - USA	ITC - ITALIA
GRECIA	0601	WUI - USA	ITT - USA	RCA - USA

PAIS DE DESTINO	CODIGO DE ACCESO	PRIMERA RUTA	1. ALTERNATIVA	2. ALTERNATIVA
GROENLANDIA	0503	RCA - USA	ITT - USA	ITC - ITALIA
GUADALUPE	0299	ITC - ITALIA	RCA - USA	ITT - USA
GUAM	0700	RCA - USA	ITT - USA	ITC - ITALIA
GUATEMALA	0372	WUI - USA	ITT - USA	RCA - USA
GUINEA	0995	RCA - USA	ITT - USA	ITC - ITALIA
GUINEA BISSAU	0969	RCA - USA	ITT - USA	ITC - ITALIA
GUINEA ECUATORIAL	0999	RCA - USA	ITT - USA	ITC - ITALIA
GUAYANA	0295	ITT - USA	RCA - USA	WUI - USA
GUAYANA FRANCESA	0300	RCA - USA	ITT - USA	ITC - ITALIA
HAITI	0203	RCA - USA	ITT - USA	WUI - USA
HAWAI (RCA)	0704	RCA - USA	ITT - USA	WUI - USA
HAWAI (ITT)	0705	ITT - USA	RCA - USA	WUI - USA
HAWAI (WUI)	0708	WUI - USA	ITT - USA	RCA - USA
HAWAI (WUI)	0709	WUI - USA	ITT - USA	RCA - USA
HAWAI (HTC)	0773	ITT - USA	RCA - USA	WUI - USA
HONDURAS	0374	TRT - USA	WUI - USA	ITC - ITALIA
HONK KONG	0802	WUI - USA	ITT - USA	RCA - USA
HUNGRIA	061	WUI - USA	ITT - USA	RCA - USA
INDIA	081	WUI - USA	ITT - USA	RCA - USA
INDONESIA	073	RCA - USA	ITT - USA	WUI - USA
IRAN	088	RCA - USA	ITT - USA	WUI - USA
IRAQ	0491	WUI - USA	ITT - USA	RCA - USA
IRLANDA	0500	WUI - USA	ITT - USA	RCA - USA
ISLANDIA	0501	RCA - USA	ITT - USA	ITC - ITALIA
VIRGINIAS ISLAS	0208	ITT - USA	RCA - USA	ITC - ITALIA
ISRAEL	0606	WUI - USA	ITT - USA	RCA - USA
ITALIA	043	ITC - ITALIA	RCA - USA	ITT - USA
JAMAICA	0291	RCA - USA	ITT - USA	WUI - USA
JAPON	072	- JAPON	RCA - USA	ITT - USA
JORDANIA	0493	WUI - USA	ITT - USA	ITC - ITALIA
KENYA	0987	ITT - USA	RCA - USA	WUI - USA
KUWAIT	0496	ITC - ITALIA	RCA - USA	ITT - USA
LAO	0804	RCA - USA	ITT - USA	WUI - USA
LESOTHO	0963	RCA - USA	ITT - USA	WUI - USA
LIBANO	0494	RCA - USA	ITT - USA	WUI - USA
LIBERIA	0997	RCA - USA	ITT - USA	WUI - USA
LIBIA	0901	ITT - USA	RCA - USA	ITC - ITALIA
LUXEMBURGO	0402	WUI - USA	ITT - USA	RCA - USA
MACAOS	0808	RCA - USA	ITT - USA	WUI - USA
MADAGASCAR	0986	ITC - ITALIA	RCA - USA	ITT - USA
MALASYA	084	RCA - USA	ITT - USA	WUI - USA
MALAWI	0904	RCA - USA	ITT - USA	WUI - USA
MALDIVAS	0896	RCA - USA	ITT - USA	WUI - USA
MALI	0985	ITC - ITALIA	RCA - USA	ITT - USA
MALTA	0406	ITC - ITALIA	RCA - USA	ITT - USA
MARIANAS	0760	RCA - USA	ITT - USA	WUI - USA
MARRUECOS	0407	RCA - USA	ITT - USA	WUI - USA
MARTINICA	0298	ITC - ITALIA	RCA - USA	ITT - USA
MAURICIO	0966	RCA - USA	ITT - USA	WUI - USA
MAURITANIA	0974	RCA - USA	ITT - USA	WUI - USA
MEXICO	022	WUI - USA	ITT - USA	RCA - USA
MONGOLIA	0800	ITC - ITALIA	RCA - USA	ITT - USA
MONTSERRAT	0396	ITT - USA	RCA - USA	WUI - USA

PAIS DE DESTINO	CODIGO DE ACCESO	PRIMERA RUTA	1. ALTERNATIVA	2. ALTERNATIVA
MOZAMBIQUE	0992	ITC - ITALIA	RCA - USA	ITT - USA
NAMIBIA	0908	RCA - USA	ITT - USA	WUI - USA
NAURU	0775	RCA - USA	ITT - USA	WUI - USA
NEPAL	0891	RCA - USA	ITT - USA	WUI - USA
NICARAGUA	0375	WUI - USA	ITT - USA	RCA - USA
NIGER	0975	RCA - USA	ITT - USA	WUI - USA
NIGERIA	0905	RCA - USA	ITT - USA	WUI - USA
NORUEGA	056	ITT - USA	RCA - USA	WUI - USA
NUEVA CALEDONIA	0706	RCA - USA	ITT - USA	WUI - USA
NUEVAS HEBRIDAS	0771	ITC - ITALIA	RCA - USA	ITT - USA
NUEVA ZELANDIA	074	RCA - USA	ITT - USA	WUI - USA
OMAN	0498	TRT - USA	ITT - USA	ITC - ITALIA
PAISES BAJOS	044	TRT - USA	ITT - USA	ITC - ITALIA
PAKISTAN	082	TRT - USA	ITT - USA	ITC - ITALIA
PANAMA (INTEL)	0377	INTEL - PANAMA	TRT - USA	WUI - USA
PANAMA (TRT)	0378	TRT - USA	ITT - USA	WUI - USA
PANAMA (INTEL)	0379	INTEL - PANAMA	TRT - USA	WUI - USA
PADUA	0703	TRT - USA	ITT - USA	ITC - ITALIA
PARAGUAY	0305	WUI - USA	ITT - USA	RCA - USA
PERU	036	ENTEL - PERU	ITT - USA	RCA - USA
POLINESIA	0702	ITT - USA	RCA - USA	WUI - USA
POLONIA	063	WUI - USA	ITT - USA	RCA - USA
PORTUGAL	0404	WUI - USA	ITT - USA	RCA - USA
PUERTO RICO (RCA)	0205	RCA - USA	ITT - USA	WUI - USA
PUERTO RICO (AACR)	0206	ITT - USA	RCA - USA	WUI - USA
PUERTO RICO (CW/WUI)	0207	WUI - USA	ITT - USA	RCA - USA
PUERTO RICO (PRCA)	0209	RCA - USA	ITT - USA	WUI - USA
QATAR	0497	ITC - ITALIA	RCA - USA	ITT - USA
REINO UNIDO	051	WUI - USA	ITT - USA	RCA - USA
REUNION	0961	ITT - USA	RCA - USA	WUI - USA
RUANDA	0899	RCA - USA	ITT - USA	WUI - USA
	0909			
RUMANIA	065	RCA - USA	ITT - USA	WUI - USA
SANTA ELENA	0960	RCA - USA	ITT - USA	WUI - USA
SAN KITTS	0397	WUI - USA	ITT - USA	RCA - USA
SANTA LUCIA	0398	RCA - USA	ITT - USA	WUI - USA
SAN PEDRO	0204	ITC - ITALIA	RCA - USA	ITT - USA
SAN VICENTE	0399	RCA - USA	ITT - USA	WUI - USA
SALOMON	0778	RCA - USA	ITT - USA	WUI - USA
SAMOA NORTE	0770	RCA - USA	ITT - USA	WUI - USA
SAMOA OCCIDENTAL	0779	RCA - USA	ITT - USA	WUI - USA
SAN TOME	0967	ITT - USA	RCA - USA	WUI - USA
SENEGAL	0906	ITT - USA	RCA - USA	WUI - USA
SEYCHELLES	0965	RCA - USA	ITT - USA	WUI - USA
SIERRA LEONA	0998	RCA - USA	ITT - USA	WUI - USA
SINGAPUR	087	TRT - USA	ITT - USA	ITC - ITALIA
SIRIA	0492	ITC - ITALIA	RCA - USA	ITT - USA
SOMALIA	0900	ITC - ITALIA	RCA - USA	ITT - USA
SRI LANKA	0803	TRT - USA	ITT - USA	ITC - ITALIA
SUDAFRICA	095	WUI - USA	ITT - USA	RCA - USA

PAIS DE DESTINO	CODIGO DE ACCESO	PRIMERA RUTA	1. ALTERNATIVA	2. ALTERNATIVA
SUDAN	0984	RCA - USA	ITT - USA	WUI - USA
SUECIA	054	WUI - USA	ITT - USA	ITC - ITALIA
SUIZA	045	WUI - USA	ITT - USA	ITC - ITALIA
SURINAME	0304	RCA - USA	ITT - USA	WUI - USA
SWAZILANDIA	0964	RCA - USA	ITT - USA	WUI - USA
TAILANDIA	086	TRT - USA	ITT - USA	ITC - ITALIA
TAIWAN	0785	WUI - USA	ITT - USA	RCA - USA
TANZANIA	0989	RCA - USA	ITT - USA	WUI - USA
TOGO	0977	RCA - USA	ITT - USA	WUI - USA
TONGA	0777	RCA - USA	ITT - USA	WUI - USA
TRANSKEI	0968	ITC - ITALIA	RCA - USA	ITT - USA
	0990			
TRINIDAD TOBAGO	0294	RCA - USA	ITT - USA	WUI - USA
TUNEZ	0409	RCA - USA	ITT - USA	WUI - USA
TURQUESAS	0296	RCA - USA	ITT - USA	WUI - USA
TURQUIA	0607	WUI - USA	ITT - USA	RCA - USA
UGANDA	0988	ITT - USA	RCA - USA	ITC - ITALIA
U.R.S.S.	064	ITT - USA	RCA - USA	ITC - ITALIA
URUGUAY	032	WUI - USA	ITT - USA	RCA - USA
VATICANO	0504	ITC - ITALIA	RCA - USA	ITT - USA
VENEZUELA	031	CNTV - VENEZUE.	RCA - USA	WUI - USA
VIETNAM	0805	RCA - USA	ITT - USA	WUI - USA
VIRGENES	0292	RCA - USA	ITT - USA	WUI - USA
WALLIS Y FUTUNA	0707	RCA - USA	ITT - USA	WUI - USA
YEMEN R.A.	0895	ITC - ITALIA	RCA - USA	ITT - USA
YEMEN R.D.P.	0806	ITC - ITALIA	RCA - USA	ITT - USA
YUGOSLAVIA	062	WUI - USA	ITT - USA	RCA - USA
ZAMBIA	0902	RCA - USA	ITT - USA	WUI - USA
ZAIRE	0982	RCA - USA	ITT - USA	WUI - USA
ZIMBAWE	0907	ITT - USA	RCA - USA	WUI - USA

ANEXO 2



CODIGO	DESCRIPCION
00	Pedido de desconexión por parte del abonado llamante.
01	Abonado ausente (ABS).
02	Línea con avería (DER).
03	Abonado temporalmente no obtenible (INF).
04	Mensajes con este abonado no se admiten (NA).
05	Número del abonado ha sido cambiado (NCH).
06	Abonado está ocupado (OCC).
07	La máquina del abonado está con avería (DER).
08	Circuitos no disponibles (NC).
09	Número de marcación inválido (NP).
10	El abonado llamado no es un suscriptor (NP).
11	El abonado llamado no responde a un pedido de llamada.
12	Error en la clase de servicio.
13	No se usa.
14	Error en el sistema EDX-C, por ejemplo número de terminal de línea inválido (NC).
15	No se usa.
16	Caracteres extraños enviados desde el abonado llamante.
17	Caracteres extraños enviados desde el abonado llamado.
18	Caracteres extraños recibidos por el sistema EDX-C.
19	No hay disponible un bloque en el Registro de Datos de Llamada.
20	Error durante el pedido del Indicativo.
21	El Terminal de Línea no reacciona.
22	No se recibe el Indicativo.
23	Indicativo recibido inválido.
24	Número del concentrador inválido.
25	Entrada desde el concentrador no se permite.
26	Número de troncal inválido.
27	Error en el estado de la llamada.
28	Error en la recepción de la clase de servicio.
29	Error en la señal de transmisión del PTS 2.
30	Error en la transmisión de la señal del PTS 2.
31	Error de preselección.
32	Dígitos de selección inválidos.
33	Clase de servicio inválida del abonado llamante.
34	Clase de servicio inválido después de recibir los dígitos de selección.
35	Error en la selección del abonado llamante.
36	Error en la señal de transmisión del pedido de llamada.
37	Error en la transmisión de la señal de pedido de llamada.
38	Error en la señal de espera para confirmación de llamada.
39	Error en la señal de PTS recibida.
40	Error en la transmisión de la señal de selección.
41	Error en la transmisión de la clase de servicio.
42	Clase de servicio inválido del abonado llamado.
43	Prefijo de selección inválida.
44	Error en el envío de prefijo de selección.

45 Error en el envío de la información de selección.  
46 Error en el envío del caracter de fin de selección.  
47 Error en el envío de la clase de servicio.  
48 Error en la transmisión del prefijo de selección.  
49 Error de selección del abonado llamado.  
50 Error en la recepción de la señal de conexión de llamada.  
51 Error en el envío del encabezamiento hacia el abonado llama-  
mado.  
52 Error en la transmisión de las señales de servicio.  
53 Terminal de línea bloqueado.  
54 Terminal de grupo desconectado.  
55 Conmutación del sistema EDX-C: del estado de conexión al es-  
tado de desconexión.  
56 No se usa.  
57 No se usa.  
58 No se usa.  
59 Desconexión forzada debido a la detección de lazo abierto.  
60 Desconexión forzada por el bloqueo del terminal de línea o  
por la desconexión del terminal de grupo efectuada por el o-  
perador.  
61 La llamada ha sido desconectada debido a que la duración de  
la misma excedió el máximo de tiempo permitido de 13.65 ho-  
ras.  
62 El terminal de línea del abonado llamado o llamante ha sido  
desconectado en el establecimiento de la comunicación.  
63 El abonado llamante ha solicitado al sistema la desconexión  
pulsando en su teleimpresor cuatro letras M.  
64 El abonado llamante ha solicitado al sistema la desconexión  
pulsando en su teleimpresor cuatro letras N.  
65 El abonado llamante ha enviado la señal de fin de comunica-  
ción.  
66 No está disponible el teleimpresor de monitoreo del sistema.  
67 Teleimpresor de monitoreo no envió la señal de confirmación  
de llamada.  
68 No se usa.  
Códigos de error del servicio de multimensaje:  
69 No hay disponible un bloque del Registro de Datos de Llama-  
da.  
70 El Test para terminales de línea presenta fallas.  
71 Desconexión del abonado llamante durante el pedido de servicio.  
72 La llamada se extendió fuera del tiempo permitido.  
73 Desconexión del abonado llamado durante la señal de conexión  
y la señal de pedido de Indicativo efectuada por el sistema.  
74 Error del abonado llamado en el envío del encabezamiento  
del multimensaje.  
75 Número de terminales de línea no disponibles para el servi-  
cio de multimensaje.  
76 Número de abonados llamados mayor al permitido por el siste-  
ma.

77 Abonado llamado desconectado durante la conmutación del sistema del estado de conexión a desconexión.

78 Inválido número del terminal de grupo.

79 No se usa.

80 No se usa.

81 No se usa.

82 No se usa.

83 No se usa.

84 No se usa.

85 No se usa.

Códigos de error del servicio de entrega diferida:

86 Desconexión prematura del abonado llamado o llamante.

87 No hay disponible un bloque en el Registro de Datos de Llamada.

88 El abonado llamante se extendió del tiempo permitido por el sistema.

89 Mensaje almacenado por más del tiempo máximo permitido por el sistema.

90 No se usa.

91 Indicativo inválido.

92 Selección de información del abonado llamante inobtenible.

93 Exceso en el máximo de caracteres permitidos por el sistema.

94 Conmutación del sistema EDX-C del estado de conexión al estado de desconexión.

95 La llamada no puede ser extendida debido a que el abonado llamado fue redireccionado por el sistema.

96 No se usa.

97 No se usa.

98 No se usa.

99 No se usa.

A0 Desconexión forzada por el operador del Puesto de Conmutación Manual.

ANEXO 3

## COMANDOS-DE-CONSOLA

Una vez que se estudió el conjunto de instrucciones del procesador PDP 11/44, es necesario conocer la forma en que estas instrucciones y datos son almacenadas en la memoria principal. Una de las maneras es a través del terminal de consola, el cual por medio de los denominados Comandos de Consola nos permiten leer o escribir desde o hacia las diferentes localidades (o registros). El terminal de consola opera como un dispositivo de entrada y salida, en el que los datos desde o hacia el terminal son controlados por el programa que está siendo ejecutado por el Procesador, a esta forma de operación se le denomina **Modo I/O**; pero también se lo puede utilizar para ingresar los denominados comandos y que el Procesador los ejecutará únicamente cuando el terminal es te en el **Modo de Consola**. Este Modo se puede acceder cuando el usuario pulsa el carácter control P (ASCCI |P); en la Fig.II. se muestra la disposición de las teclas del terminal.

Los siguientes párrafos proveen una breve descripción de los principales Comandos de Consola y un ejemplo de su utilización. A continuación se detallan los mismos:

CONTINUE	C
DEPOSITE	D
EXAMINE	E
HALT	H
INICIALIZE	I
STEP TO STEP	N
REPIT	R
START	S

SINTAXIS DE LOS COMANDOS.- En la descripción de cada comando se usarán, ciertas expresiones, caracteres especiales y calificadores que a continuación detallaremos:

**EXPRESIONES:**

< >            Una expresión entre estos signos se usa para denotar categoría de nombre. Por ejemplo el nombre de categoría <dirección> representa cualquier dirección válida; así en un comando una dirección (por ejemplo 177564) debe ser pulsada en lugar del nombre de categoría. Entre las principales expresiones utilizadas entre estos signos tenemos:

<SP>	representa un espacio.
<COUNT>	representa contador numérico en octal.
<DIRECCION>	representa una dirección en octa.
<DATA>	representa un valor numérico en octal.
<CALIFICADOR>	modificador del comando
<INPUT-PROMPT>	la consola arroja la línea >>>
<CR>	retorno del carro
<LF>	avance de línea.

| |            Un expresión entre corchetes indica, en la descripción de un comando, que la expresión es opcional.

CALIFICADORES.- Los comandos pueden ser modificados añadiendo los denominados calificadores, los cuales expanden la capacidad de los comandos. Todos los calificadores son opcionales y siempre empiezan con un slash (/). La siguiente es una lista de los calificadores con una breve descripción de los mismos:

- /G Provee un método para leer o grabar en uno de los registros de propósito general. Es utilizado con los comandos E o D (Examine o Deposit), seguido por el calificador /G y por el número que representa a uno de los registros generales (0,1,2,3,4,5,6,7). Por ejemplo al ejecutar: E/G <SP> 7 <CR>, el contenido del registro 7 aparecerá impreso en el terminal de consola.
- /N Permite efectuar varias veces los comandos E o D en direcciones de memoria secuenciales, sin necesidad de usar un nuevo comando para cada dirección. Generalmente el calificador /N se usa en conjunción con el valor del término que representa <COUNT>, el cual especifica el número de operaciones secuenciales a ser realizadas. La sintaxis es: /N | : <COUNT> |  
 Por ejemplo al ingresar el comando: E/N:6 <SP> 1000, se imprimirá en el terminal de consola el contenido de las localidades de memoria 1000 a 1006.

DESCRIPCION-DE-LOS-PRINCIPALES-COMANDOS.-

CONTINUE        C <CR>

Al ejecutar este comando el Procesador inicia la ejecución de la instrucción desde la localidad de memoria cuya dirección está contenida en el registro contador de programa.

DEPOSITE        D |<CALIFICADOR>| <SP> <DIRECCION> <SP> <DATA> <CR>

Este comando carga o deposita datos en localidades de memoria.

<DIRECCION> representa un número octal de 1 a 8 dígitos, pero cuando se usa el calificador /G este valor no puede exceder de 17. Alternativamente se puede utili

lizar direcciones mnemónicas que las describiremos más adelante.

<DATA> Representa un número octal de 1 a 6 dígitos.

La respuesta del Procesador al ejecutar un comando Deposite es:

<CR> <LF> <INPUT PROMPT>

lo que nos indica que el terminal de consola se ubicará en la primera posición de la siguiente línea de la hoja en la cual se imprimirá >>>.

Ejemplos:

>>>D <SP> 1000 <SP> 5 Deposita 5 en la localidad 1000.

>>>D/G/N:5 <SP> 0 <SP> 35 Deposita 35 en los 5 primeros registros de propósito general, iniciando el proceso desde R0.

EXAMINE E | |<CALIFICADOR> | <SP> <DIRECCION> | <CR> |

Examina el contenido de la localidad de memoria (o registro) especificado en el campo dirección. <DIRECCION> y <DATA>, tienen el mismo significado que en el caso del comando Deposit.

La respuesta a este comando es un Prompt (>>>), seguido de:

<CR> <LF> <DIRECCION> <SP> <DATA>

lo que nos indica que el terminal de consola, al ejecutar este comando, se ubicará en la primera posición de la siguiente línea de la hoja en la cual se imprimirá la dirección de la localidad de memoria (o registro) seguido por el contenido de dicha localidad.

Ejemplos:

>>>E <SP> 1000 <CR>

00001000 002625 Examina el contenido de la localidad 1000, el mismo que es de 002625.

>>>E/N:3 2000

Examina secuencialmente el contenido de las localidades 2000, 2001 y 2002.



00002000 000001  
00002001 000002  
00002002 000003

En los dos comandos anteriores E y D, cuando no se especifica <DIRECCION> o <DATA> se asume el valor de cero. Además, se pueden utilizar direcciones mnemónicas que son la siguientes:

- + Deposita (examina) la localidad de memoria o registro que sigue a la última localidad o registro referenciado.
- Deposita (examina) la localidad de memoria o registro inmediatamente anterior a la última localidad o registro referenciado.
- \* Deposita (examina) la última localidad o registro referenciada.
- @ Deposita (examina) una dirección indicada por el último dato depositado (o examinado).

Ejemplos:

>>>D <SP> 1000 <SP> 5      Deposita 5 en la localidad 1000.  
>>>D <SP> + <SP> 776      Deposita 776 en la localidad siguiente a la última localidad referenciada, en este caso en la localidad 1002.  
>>>D <SP> \* <SP> 400      Deposita 400 en la última localidad referenciada, en este caso en la localidad 1002.  
>>>E <SP> @ <CR>

HALT            <CR>

Si el procesador estuvo ejecutando un programa para la ejecución del mismo no sin antes de finalizar la instrucción en curso. Como resultado de este comando se imprimirá en el terminal de consola el contenido

del contador del programa (PC).

Ejemplo:

>>> H <CR>

17777707 001000 El Procesador imprime el contenido del PC.

NEXT N | <SP> <COUNT> |<CR>

El Procesador ejecutará, desde la dirección almacenada en el contador de programa, el número de instrucciones especificadas por el valor que representa <COUNT>. Si no se especifica <COUNT> una sola instrucción se ejecutará; el contador se decrementa después que cada instrucción es ejecutada, hasta cuando toma el valor de cero y entonces el Terminal de Consola imprimirá la última dirección contenida en el PC.

Ejemplo:

>>> N <SP> 3 <CR>

Se ejecutan secuencialmente 3 instrucciones y se imprimen las direcciones de las mismas.

17777707 001000

17777707 001002

17777707 001004

REPEAT R <SP> <COMANDOS DE CONSOLA> <CR>

Este comando ejecuta repetidamente el comando Examine o Deposit hasta cuando se pulsa el carácter control-C (ASCCI: ).

START S |<SP> <DIRECCIONES> |<CR>

El valor que representa <DIRECCION> se almacena en el registro PC y si no se especifica no se carga ningún valor. El Procesador iniciará la ejecución de las instrucciones a partir de la dirección contenida en PC.

Ejemplo:

>>> S <SP> 1000 <CR>

Deposita 1000 en el PC e inicia la ejecución desde dicha localidad.

ANEXO 4

### INSTRUCCIONES DE SIMPLE OPERANDO

<u>Instrucción</u>	Mnemotécnico	Octal
<u>CLEAR</u>	CLR CLR (B)	0050DD 1050DD

Bits de condición

N = 0

Z = 1

V = 0

C = 0

Descripción: El contenido de la localidad de memoria indicada (o de un registro general) por el campo de destino es reemplazado por ceros.

<u>Instrucción</u>	Mnemotécnico	Octal
<u>COMPLEMENT</u>	COM COM (B)	0051DD 1051DD

Bits de condición

N = 1

Z = 0

V = 0

C = 1

Descripción: El contenido de una localidad de memoria (o de un registro general) indicada por el campo de destino es reemplazada por su complemento a uno.

<u>Instrucción</u>	Mnemotécnico	Octal
<u>INCREMENT</u>	INC INC(B)	0052DD 1052DD

Bits de condición

N = 1

Z = 1

V = 1

C = \*

Descripción: Añade 1 al contenido de la localidad de memoria (o de un registro general) indicada por el campo de destino.

<u>Instrucción</u>	Mnemotécnico	Octal
<u>DECREMENT</u>	DEC DEC(B)	0053DD 1053DD

Bits de condición

N = 1

Z = 1

V = 1

C = \*

Descripción: Subtrae 1 del contenido de la localidad de memoria (o de un registro general) indicada por el campo de destino.

<u>Instrucción</u>	Mnemotécnico	Octal
NEGATE	NEG	0054DD
	NEG(B)	1054DD

Bits de condición

N = 1

Z = 1

V = 1

C = \*

Descripción: Reemplaza el contenido de la localidad de memoria (o de un registro general) especificado por el campo de destino, por su complemento a dos.

<u>Instrucción</u>	Mnemotécnico	Octal
ARITHMETIC	ASL	0063DD
SHIFT LEFT	ASL(B)	1063DD

Bits de condición

N = 1

Z = 1

V = N ⊗ C

C = se carga con el MSB del contenido de la dirección del campo de destino.

Descripción: Todos los bits especificados por la dirección del campo de destino se desplazan un lugar hacia la izquierda. El bit LSB se carga con cero; y el bit MSB se transfiere al bit de condición C.

<u>Instrucción</u>	Mnemotécnico	Octal
ARITHMETIC	ASR	0062DD
SHIFT RIGHT	ASR(B)	1062DD

Bits de condición

N = 1

Z = 1

V = N ⊗ C

C = se carga con el LSB indicado en la dirección del campo destino.

Descripción: Todos los bits especificados por la dirección del campo de destino se desplazan un lugar hacia la derecha. El bit LSB se transfiere al bit de condición C.

<u>Instrucción</u>	Mnemotécnico	Octal
ROTATE RIGHT	ROR ROR(B)	0060DD 1060DD

Bits de condición

N = 1

Z = 1

V = N ⊕ C

C = 1, si el LSB del campo de destino fue 1 antes de ejecutarse la instrucción.

Descripción: Todos los bits especificados por la dirección del campo de destino rotan un lugar a la derecha. El bit LSB se transfiere al bit de condición C y el contenido anterior se transfiere al MSB del campo de destino.

<u>Instrucción</u>	Mnemotécnico	Octal
ROTATE LEFT	ROL ROL(B)	0061DD 1061DD

Bits de condición:

N = 1

Z = 0

V = N ⊕ C

C = 1, Si el MSB del campo de destino fue 1 antes de ejecutarse la instrucción.

Descripción: Todos los bits del campo de destino rotan un lugar hacia la izquierda. El MSB pasa al bit C y el previo contenido de este último bit pasa al LSB del campo de destino.

<u>Instrucción</u>	Mnemotécnico	Octal
TEST	TST TST(B)	0057DD 1057DD

Bits de condición

N = 1

Z = 1

V = 0

C = 0

Descripción: Coloca los bits de condición N y Z en el registro PSW, de acuerdo al contenido de la localidad de memoria (o de un registro general) indicado por el campo de destino.

<u>Instrucción</u>	Mnemotécnico	Octal
SWAP BYTES	SWAB	0003DD

Bits de condición:

N = 1

Z = 1

V = 0

C = 0

Descripción: Intercambia el high byte con el low byte contenidos en la localidad de memoria (o en un registro general) especificada por el campo de destino.

<u>Instrucción</u>	Mnemotécnico	Octal
ADD CARRY	ADC	0055DD
	ADC(B)	1055DD

Bits de condición

N = 1

Z = 1

V = 1

C = \*

Descripción: Añade el bit C con el contenido de la localidad de memoria (o al de un registro general) indicada por el campo de destino.

<u>Instrucción</u>	Mnemotécnico	Octal
SUBTRACT CARRY	SBC	0056DD
	SBC(B)	1056DD

Bits de condición

N = 1

Z = 1

V = 1

C = 1

Descripción: Substrae el bit C del contenido de la localidad de memoria (o de un registro general) indicada por el campo de destino.

<u>Instrucción</u>	Mnemotécnico	Octal
SIGN EXTEND	SXT	0067DD

Bits de condición

N = \*

Z = 1

V = 0

C = \*

Descripción: Si el bit de condición N es 1, entonces -1 se carga en la localidad de memoria (o en un registro general) especificada por el campo de destino; si N es 0 entonces se cargan ceros.

### INSTRUCCIONES DE DOBLE OPERANDO

<u>Instrucción</u>	Mnemotécnico	Octal
MOVE	MOV MOV(B)	01SSDD 11SSDD

Bits de condición:

N = 1  
Z = 1  
V = 0  
C = \*

Descripción: El contenido de la localidad de memoria (o de un registro general) especificada por el campo fuente se transfiere hacia la localidad de memoria (o hacia un registro general) indicada por el campo de destino. El operando fuente no se altera.

<u>Instrucción</u>	Mnemotécnico	Octal
ADD	ADD	06SSDD

Bits de condición

N = 1  
Z = 1  
V = 1  
C = 1

Descripción: Añade el contenido de la localidad de memoria (o de un registro general) especificado por el campo fuente con el contenido de la localidad de memoria (o de un registro general) especificada por el campo de destino. El resultado se almacena en la localidad indicada por el campo de destino, y el contenido de la localidad del campo fuente no cambia.

<u>Instrucción</u>	Mnemotécnico	Octal
SUBSTRACT	SUB	16SSDD

Bits de condición:

N = 1  
Z = 1  
V = 1  
C = 1



Descripción: Subtrae el contenido de la localidad de memoria ( o de un registro general) especificado por el campo fuente con el contenido de la localidad de memoria (o de un registro general) especificado por el campo de destino. El resultado se almacena en la localidad indicada por el campo de destino, y el contenido de la localidad del campo fuente no cambia.

<u>Instrucción</u>	Mnemotécnico	Octal
COMPARE	CMP CMP(B)	02SSDD 12SSDD

Bits de condición

N = 1

Z = 1

V = 1

C = 1

Descripción: Campara el contenido de la localidad de memoria (o de un registro general) especificado por el campo fuente, con el contenido de la localidad de memoria (o de un registro general) especificado por el campo de destino. Y coloca los bits de condición de acuerdo al resultado de la comparación.

<u>Instrucción</u>	Mnemotécnico	Octal
ARITHMETIC SHIFT	ASH	072RSS

Bits de condición

N = 1

Z = 1

V = 1

C = 0, si R es 0. 0 se carga con el bit que fue desplazado fuera del registro R.

Descripción: los bits del registro R son desplazados hacia la derecha o a la izquierda, el número de veces especificados por el contenido de la localidad de memoria (o de un registro general) especificada por el campo fuente; constituyendo un contador de desplazamiento de 6 bits (desde -32 a +31). Un número negativo indica un desplazamiento a la derecha y un positivo un desplazamiento a la izquierda

<u>Instrucción</u>	Mnemotécnico	Octal
ARITHMETIC SHIFT COMBINED	ASHC	073RSS
Bits de condición		
N = 1		
Z = 1		
V = 1		
C = se carga con el bit MSB si el desplazamiento es hacia la izquierda o se carga con el bit LSB si el desplazamiento es hacia la derecha.		
Descripción: El contenido del registro Rn y del registro Rn+1 (n = 0,1, 2,3,4,5,6) son tratados como un solo registro y los 32 bits que lo conforman son desplazados el número de veces que in- dica el contenido de la localidad de memoria (o registro) especificados en el campo fuente. El rango de desplazamien- to es de -32 a +31, un número negativo significa un despla- miento hacia la derecha y un positivo hacia la izquierda; un cero significa que no hay desplazamiento.		

<u>Instrucción</u>	Mnemotécnico	Octal
MULTIPLY	MULT	070RSS
Bits de condición		
N = 1		
Z = 1		
V = 0		
C = 1, si el resultado es menor que $-2^{15}$ o mayor o igual que $+2^{15}$ .		
Descripción: El contenido del registro R y de la localidad de memoria (o registro) especificada en el campo fuente se represen- tan como enteros en complemento a 2, son multiplicados en tre si y el resultado se almacena en los registros Rn y Rn+1 si n es impar; si n es par solamente el low-byte del producto se almacena.		

<u>Instrucción</u>	Mnemotécnico	Octal
DIVIDE	DIV	071RSS
Bits de condición		
N = 1		
Z = 1		
V = 1, si el operando fuente es 0 o sino puede ser representado como un entero en complemento a 2.		
C = 1, si se divide por 0.		
Descripción: El contenido del registro R se divide por el operando fuente y el resultado se almacena en R.		

<u>Instrucción</u>	Mnemotécnico	Octal
BIT TEST	BIT BIT(B)	03SSDD 13SSDD

Bits de condición

N = 1

Z = 1

V = 0

C = \*

Descripción: Efectúa la comparación lógica AND entre el contenido de la localidad de memoria (o registro) especificada por el campo fuente, y el contenido de la localidad de memoria (o registro) especificado por el campo destino. Además modifica los bits de condición de acuerdo al resultado de la operación, los operandos del campo fuente y del campo destino son afectados.

<u>Instrucción</u>	Mnemotécnico	Octal
BIT CLEAR	BIC BIC(B)	04SSDD 14SSDD

Bits de condición

N = 1

Z = 1

V = 0

C = \*

Descripción: Coloca un 0 en cada bit en la localidad de memoria (o registro) especificado por el campo de destino y correspondiente al bit que es puesto a 1 en la localidad de memoria (o registro) especificado por el fuente. El resultado se almacena en esta última localidad (o registro); el operando fuente mantiene su valor después de ejecutada la instrucción.

<u>Instrucción</u>	Mnemotécnico	Octal
BIT SET	BIS BIS(B)	05SSDD 15SSDD

Bits de condición

N = 1

Z = 0

V = 0

C = \*

Descripción: Realiza la operación lógica OR entre los operando fuente y destino y el resultado se almacena en la localidad de memoria (o registro) especificado por el campo destino. El operando fuente mantiene su valor después de ejecutada la instrucción.

Instrucción

Mnemotécnico

Octal

EXCLUSIVE OR

XOR

074RDD

Bits de condición

N = 1

Z = 1

V = 0

C = \*

Descripción: Ejecuta la operación lógica OR EXCLUSIVE entre el contenido del registro R y el operando de destino. El resultado se almacena en la localidad de memoria (o registro) especificada en el campo de destino. El contenido de R no se altera después de ejecutada la instrucción.

### INSTRUCCIONES DE BIFURCACION

Los bits de condición mantiene su valor después de ejecutada la instrucción.

Instrucción

Mnemotécnico

Octal

BRANCH UNCONDITIONAL

BR

000400

más 8 bits de desplazamiento

Descripción: Provee una manera de transferir el control del programa dentro de un rango de -128 a +127 palabras.

Instrucción

Mnemotécnico

Octal

BRANCH IF NOT

BNE

001000

EQUAL TO ZERO

más 8 bits de desplazamiento

Descripción: Prueba el estado del bit Z y causa la bifurcación si el bit Z es cero.

Instrucción	Mnemotécnico	Octal
BRANCH IF EQUAL TO ZERO	BEQ	001400 más 8 bits de desplazamiento
Descripción: Prueba el estado del bit Z y causa la bifurcación si el bit Z es 1.		

Instrucción	Mnemotécnico	Octal
BRANCH IF PLUS	BPL	100000 más 8 bits de desplazamiento
Descripción: Prueba el estado del bit N y causa la bifurcación si N es cero.		

Instrucción	Mnemotécnico	Octal
BRANCH IF MINUS	BMI	100400 más 8 bits de desplazamiento
Descripción: Prueba el estado del bit N y causa la bifurcación si N es 1.		

Instrucción	Mnemotécnico	Octal
BRANCH IF OVERFLOW IS CLEAR	BVC	102000 más 8 bits de desplazamiento
Descripción: Prueba el estado del bit V y causa la bifurcación si V es cero.		

Instrucción	Mnemotécnico	Octal
BRANCH IF OVERFLOW IS SET	BVS	102400 más 8 bits de desplazamiento
Descripción: Prueba el estado del bit V y causa la bifurcación si V es uno.		

Instrucción	Mnemotécnico	Octal
BRANCH IF CARRY IS CLEAR	BCC	103000 más 8 bits de desplazamiento
Descripción: Prueba el estado del bit C y causa la bifurcación si C es cero.		

Instrucción	Mnemotécnico	Octal
BRANCH IF CARRY IS SET	BCS	103400 más 8 bits de desplazamiento

Descripción: Prueba el estado del bit C y causa la bifurcación si C es uno.

Instrucción	Mnemotécnico	Octal
BRANCH IS GREATER THAN OR EQUAL TO ZERO	BGE	002000 más 8 bits de desplazamiento

Descripción: Causa una bifurcación si los bits  $N = 1$  y  $V = 1$  o si  $N = 0$  y  $V = 0$ .

Instrucción	Mnemotécnico	Octal
BRANCH IF LESS THAN ZERO	BLT	002400 más 8 bits de desplazamiento

Descripción: Causa una bifurcación si el resultado de la operación lógica OR, entre los bits N y V es 1.

Instrucción	Mnemotécnico	Octal
BRANCH IF GREATER THAN ZERO	BGT	003000 Más 8 bits de desplazamiento

Descripción: Causa una bifurcación si el bit Z es cero y el bit N es igual a el bit V.

Instrucción	Mnemotécnico	Octal
BRANCH IF LESS THAN OR EQUAL TO ZERO	BLE	003400 más 8 bits de desplazamiento

Descripción: Causa una bifurcación si el bit Z es cero y si el bit N es diferente al bit V.

Instrucción	Mnemotécnico	Octal
SUBTRACT ONE AND BRANCH IF NOT EQUAL TO ZERO	SOB	077R00 más 6 bits de desplazamiento

Descripción: El contenido del registro R se decrementa en 1. Si no es igual a cero, se transfiere el control del programa a la localidad de memoria indicada por el desplazamiento, en caso contrario se ejecuta la siguiente instrucción.

<u>Instrucción</u>	Mnemotécnico	Octal
BRANCH IF HIGHER	BHI	101000
		más 8 bits de desplazamiento
Descripción: Causa la bifurcación si la operación anterior no produce un carry o un resultado 0.		

<u>Instrucción</u>	Mnemotécnico	Octal
BRANCH IF LOWER OR SAME	BLOS	101400
		más 8 bits de desplazamiento
Descripción: Causa la bifurcación si la operación anterior causa un carry o un resultado cero.		

<u>Instrucción</u>	Mnemotécnico	Octal
BRANCH IF HIGHER OR SAME	BHIS	103000
		más 8 bits de desplazamiento
Descripción: Prueba el estado del bit C y causa la bifurcación si C es cero (0).		

<u>Instrucción</u>	Mnemotécnico	Octal
BRANCH IF LOWER	BLO	103400
Descripción: Prueba el estado del bit C y causa la bifurcación si C es uno.		

### INSTRUCCIONES DE SUBROUTINA Y SALTO

No afectan los bits de condición.

<u>Instrucción</u>	Mnemotécnico	Octal
JUMP	JMP	0001DD
Descripción: El control del programa se transfiere a la localidad de memoria indicada en la dirección del campo destino.		

<u>Instrucción</u>	Mnemotécnico	Octal
JUMP TO SUBROUTINE	JSR	004RDD
Descripción: El control del programa se transfiere a la localidad de memoria, cuya dirección está especificada por el campo de destino. Este último valor se transfiere al PC y desde dicha localidad se inicia la ejecución de la subrutina; además el contenido de R se almacena en la localidad de memoria (pila) especificada por R6.		

<u>Instrucción</u>	Mnemotécnico	Octal
RETURN FROM SUBROUTINE	RTS	00020R

Descripción: Transfiere el contenido del registro R hacia el registro PC y extrae el elemento de la pila apuntado por R6 transfiriéndole hacia el registro R.

INSTRUCCIONES DEL CODIGO DE CONDICION

No afectan los bits de condición.

<u>Instrucción</u>	Mnemotécnico	Octal
CLEAR CARRY	CLC	00024I

Descripción: Pone un cero en el bit C.

<u>Instrucción</u>	Mnemotécnico	Octal
CLEAR OVERFLOW	CLV	00024J

Descripción: Coloca un cero en el bit V.

<u>Instrucción</u>	Mnemotécnico	Octal
CLEAR NEGATIVE	CLN	00025O

Descripción: Coloca un cero en el bit N.

<u>Instrucción</u>	Mnemotécnico	Octal
CLEAR ZERO	CLZ	00024K

Descripción: Coloca un cero en el bit Z.

<u>Instrucción</u>	Mnemotécnico	Octal
CLEAR ALL CONDITION	CCC	000257

Descripción: Coloca un cero en cada uno de los bits de códigos de condición (C = V = N = Z = 0)

<u>Instrucción</u>	Mnemotécnico	Octal
SET CARRY	SEC	00026I

Descripción: Coloca un uno en el bit C.

<u>Instrucción</u>	Mnemotécnico	Octal
SET OVERFLOW	SEV	00026J

Descripción: Coloca un uno en el bit V.



<u>Instrucción</u>	Mnemotécnico	Octal
SET ZERO	SEZ	000264
Descripción: Coloca un uno en el Bit Z.		

<u>Instrucción</u>	Mnemotécnico	Octal
SET NEGATIVE	SEN	000270
Descripción: Coloca un uno en el bit N.		

<u>Instrucción</u>	Mnemotécnico	Octal
SET ALL CONDITIONS	SCC	000277
Descripción: Coloca un uno en cada uno de los bits de condición de estado (C = V = Z = N = 1).		

#### INSTRUCCIONES MISCELANEAS

<u>Instrucción</u>	Mnemotécnico	Octal
HALT	HALT	000000
Descripción: Causa que el procesador cese en la operación que está realizando.		

<u>Instrucción</u>	Mnemotécnico	Octal
NO OPERATION	NOP	000240
Descripción: No se ejecuta ninguna operación.		

Además de las instrucciones descritas, el PDP 11/44 dispone de instrucciones de entrada salida, enmascaramiento; pero por , cuanto no las utilizaremos no las hemos descrito. siembargo se anexa el conjunto completo de instrucciones.

ANEXO 5

; SUBROUTINA: INGRESO ITERATIVO DEL PERIODO DE OBSERVACION

; Notación/rangos de validación:

; 01 < mes < 12            d2 d1 < mes < M2 M1  
; 01 < día < 31            d2 d1 < día < D2 D1  
; 00 < año < 99            d2 d2 < año < A2 A1  
; 00 < hora < 23           d2 d2 < hora < H2 H1  
; 00 < minu < 59           d2 d2 < minu < N2 N1

; Argumentos de entrada:

- ; 1.- Dígitos menos significativos del rango superior: DX1  
; En localidades 7730/1 cargar con el carácter ASCII M1  
; En localidades 7732/3 cargar con el carácter ASCII D1  
; En localidades 7734/5 cargar con el carácter ASCII A1  
; En localidades 7736/7 cargar con el carácter ASCII H1  
; En localidades 7740/1 cargar con el carácter ASCII N1  
; 2.- Dígitos más significativos del rango superior: DX2  
; En localidades 7750/1 cargar con el carácter ASCII M2  
; En localidades 7752/3 cargar con el carácter ASCII D2  
; En localidades 7754/5 cargar con el carácter ASCII A2  
; En localidades 7756/7 cargar con el carácter ASCII H2  
; En localidades 7760/1 cargar con el carácter ASCII N2  
; 3.- En localidades 7704/5 cargar con la dirección de la localidad a partir de  
; la cual se almacenan los caracteres pulsados a través del teclado y que  
; constituyen los argumentos (mes, día, etc.).Cada uno almacenado en dos  
; localidades (cada una de un byte).  
; En localidades 7706/7 cargar con la dirección de la localidad restada 5,  
; desde la que se encuentran los caracteres que conforman las frases a ser  
; impresas: mes: ,día: ,año: ,hora: ,min: (cada frase cinco caracteres).  
; En localidades 7710/1 cargar con el número de caracteres que conforman ca-  
; da frase a ser impresa (cinco).  
; En localidades 7712/3 cargar el número de dígitos que conforman cada dato  
; de la fecha (2).  
; En localidades 7714/5 cargar con carácter ASCII d2.  
; En localidades 7716/7 cargar con carácter ASCII d1.  
; En localidades 7056/7 cargar dirección de retorno.



; Argumentos de salida:  
 ; 1.- Se imprime: "Ingreso Fecha Inicial (MM,DD,AA,hh,mm):"  
 ; 2.- Se imprimen cada una de las frases (Mes: ,Día ,etc) y se espera que el operador digite una secuencia de caracteres, finalizando con retorno de carro  
 ; 3.- Si los dígitos pulsados cumplen con los requisitos de validación impuestos para cada una de las variables (mes, día, etc.), se almacenan desde la localidad 500 a 510 (área FII).  
 ; 4.- Se imprime:"Ingreso Fecha Final (MM,DD,AA,hh,mm): "  
 ; Se repite lo anotado en 2.- y 3.- con la excepción de que cada una de las variables se almacenan en las localidades 512 a 522 (área FII)

17420	4737	JSR PC,CRLF	; A subrutina retorno de carro, avance de línea.
	30030		
17424	4737	JSR PC,CRLF	; A subrutina retorno de carro, avance de línea.
	30030		
17430	4737	JSR PC,CRLF	; A subrutina retorno de carro, avance de línea.
	30030		
			; Impresión de frase:
17434	012700	MOV #126,R0	; "Ingreso de Fecha Inicial (MM,DD,AA,hh,mm):".
	126		; En R0 se carga número de caracteres a imprimirse
17440	012701	MOV #300,R1	; (126).
	300		; En R1 dirección de la localidad (300) en que se
17444	4737	JSR PC,IMP	; encuentra el ler. carácter.
	30000		; A subrutina de impresión.
17450	240	NOP	
17452	240	NOP	
17454	012704	MOV #12,R4	; Inicialización de contador en doce (se finaliza con las variables de la fecha inicial cuando el contador alcanza el valor de 5
	12		; CT ——— 12 (Número de variables: 5 de fecha inicial y 5 de final)

```

; Se obtienen dígitos de rango superior de validación de la primera
; variable (mes de fecha inicial).
MOV #7730,R0 ; R0 se carga con la dirección de localidad en que
; está almacenado M1.
MOV #7750,R1 ; R1 se carga con la dirección de la localidad en
; que está almacenado M2.
NOP
; Se guardan dígitos DX2, DX1.
MOV (R0)+,@#7700 ; El contenido de la localidad apuntado por R0 se
; transfiere a localidad 7700 (DX1).
MOV (R1)+,@#7702 ; El contenido de la localidad apuntada por R1 se
; transfiere a localidad 7702 (DX2).
; Se obtiene dirección de la localidad desde la cual se almacenan
; las variables digitadas (2 localidades de un byte por variable)
; D ——— D + 2.
ADD #2,@#7704 ; El contenido de la localidad 7704 se incrementa
; en 2 y el resultado se almacena en dicha locali-
; dad.
; Se obtiene dirección de la localidad en que están almacenados los
; primeros caracteres de la frase de cada variable a imprimirse
; I ——— I + 5.
ADD #5,@#7706 ; Se incrementa en 5 el contenido de la localidad
; 7706.
NOP
NOP
NOP
CMP #10,R4 ; Es el contenido de R4 igual a 10?
BNE A1 ; No bifurca a A1
; Si continúa:
; Se hacen ceros (ASCCI) los dígitos del rango inferior de valida-
; ción de las variables (mes, día, etc.).
MOV @#7714,@#7716 ; El contenido de la localidad 7714 se transfiere a
; la localidad 7716.

```

17540	240	NOP				
17542	240	NOP				
17544	012706	AI	MOV	#10000,SP		
17550	10000		MOV	R4,-(SP)		
17552	010046		MOV	R0,-(SP)		
17554	010146		MOV	R1,-(SP)		
17556	240	NOP				
17560	012737		MOV	#17572,@#7000		
	17572					
	7000					
17566	4737		JSR	PC,ITVN		
	30140					
17572	240	NOP				
17574	240	NOP				
17576	012706		MOV	#7772,SP		
	7772					
17602	012601		MOV	(SP)+,R1		
17604	012600		MOV	(SP)+,R0		
17606	012604		MOV	(SP)+,R4		
17610	240	NOP				
17612	022704		CMP	#6,R4		
17616	1023		BNE	A2		

; Almacenamiento del contenido de los registros.  
; Se inicializa puntero de pila en 10000.  
; El contenido de R4 se transfiere a localidades  
; 7776/7.  
; El contenido de R0 se transfiere a localidades  
; 7774/5.  
; El contenido de R1 se transfiere a localidades  
; 7772/3.  
; A variable X se le somete a proceso iterativo: impresión, válida  
; ción y almacenamiento una vez que ha sido correctamente digitada  
; En localidad 7000 se almacena dirección de retor  
; no 17572.  
; A subrutina de ingreso iterativo de variables nu-  
; méricas.  
; Restitución del contenido de los registros.  
; Inicialización del puntero de pila en 7772.  
; El contenido de las localidades 7772/3 se trans-  
; fiere a R1.  
; El contenido de las localidades 7774/5 se trans-  
; fiere a R0.  
; El contenido de las localidades 7776/7 se trans-  
; fiere a R4.  
; ¿Se finaliza con las variables de la fecha inicial?  
; ¿CT = 6?  
; Es el contenido de R4 igual a 6?  
; No bifurca a A2  
; Si continúa:

```

17620      4737      JSR PC,CRLF      ; A subrutina retorno de carro, avance de línea.
           30030
           ; Impresión de frase:
           ; "Ingrese Fecha Final (MM,DD,AA,hh,mm): "
MOV        012700  #52,R0      ; En R0 se carga número de caracteres a imprimirse
           52      ; (52).
17630      012701  #426,R1     ; En R1 se carga dirección (426) de la localidad
           426    ; en que está almacenado el 1er. carácter.
17634      4737      JSR PC,IMP     ; A subrutina de impresión.
           30000
17640      012737  #130261,@#7716 ; Actualización de los argumentos para variables de fecha final.
           130261 ; Se carga en localidades 7716/77 el carácter ASCII
           7716  ; 1 (130261); dígito menos significativo del rango
17646      012700  #7730,R0     ; RO se carga con la dirección (7730) de la locali-
           7730  ; dad en que esta almacenado el dígito M1.
17652      012701  #7750,R1     ; R1 se carga con la dirección (7750) de la locali-
           7750  ; dad en que esta almacenado el dígito M2.
17656      012737  #773,@#7706 ; Localidad 7706 se carga con la dirección 773, des
           773   ; de la cual están los caracteres que conforman los
           7706 ; nombres de las variables a imprimirse.
17664      240      NOP
           ; ¿CT = 4?
17666      022704  A2      CMP #4,R4      ; ¿Es el contenido de R4 igual a 4?
17672      1003    BNE A3      ; No bifurca a A3.
           ; Si continúa: se hacen ceros (ASCII) los dígitos
           ; del rango inferior de las variables año, hora y
           ; minuto de la fecha final.
17674      013737  @#7714,@#7716 ; El contenido de las localidades 7714/5 se trans-
           7714  ; fiere a localidades 7716/7.
           7716

```

```

17702      5304      A3      DEC R4      ; CT --- CT - 1
; El contenido de R4 se decrementa en 1
; Última variable?).
17704      1272      BNE LAZO  ; ¿CT = 0?
; No bifurca a LAZO.
; Si continúa:
17706      240      NOP
17710      012706   MOV #7056,SP ; En puntero de pila se carga dirección de la loca-
7056      ; lidad en la cual está almacenada la dirección de
; retorno.
17714      207      RTS PC      ; Retorno.

```

```

*****
*****

```

**; SUBROUTINA: INGRESO ITERATIVO DE VARIABLES NUMERICAS**

```

; Notación: d2d1<X2X1<D2D1
; d,rango inferior D, rango superior X, variable(caracteres pulsados)
; Subíndice 2 dígito más significativo
; Subíndice 1 dígito menos significativo.
; Argumentos de entrada:
; En localidades 7000/1 se carga dirección de retorno
; En localidades 7700/1 se carga D1.
; En localidades 7702/3 se carga D2
; En localidades 7704/5 se carga dirección de la localidad desde la cual se
; almacenan los caracteres pulsados.
; En localidades 7706/7 se carga dirección de la localidad en que está el ler.
; carácter del nombre de la variable a imprimirse.
; En localidades 7710/1 se carga el número de caracteres que conforman el nom
; bre de la variable.
; En localidades 7712/3 se carga el número (N) de dígitos de la variable.
; En localidades 7714/5 se carga d2.
; En localidades 7716/7 se carga d1.
; Argumentos de salida:
; Se imprime en la unidad de I/O el nombre de la variable (consulta) y los ca-
; racteres pulsados por el operador (respuesta). Si estos son dos caracteres

```



; numéricos igual al valor contenido en la localidad 7712/3 y estén dentro del  
 ; rango previsto, se almacenan desde la localidad indicada por el contenido de  
 ; la localidad 7704. Variable ocupa el número de localidades (de un byte)  
 ; correspondientes al número de dígitos a validar.

```

30140      240      NOP      ; Obtención de los argumentos de entrada.
30142      012706   MOV      #7700,SP      ; Se inicializa puntero de pila en 7700.
          7700
30146      012604   MOV      (SP)+,R4      ; Contenido de la localidad 7700/1 se transfiere a
          ; R4 (D1).
30150      012603   MOV      (SP)+,R3      ; Contenido de la localidad 7702/3 se transfiere a
          ; R3 (D2).
30152      012602   MOV      (SP)+,R2      ; Contenido de la localidad 7704/5 se transfiere a
          ; R2.
30154      012601   MOV      (SP)+,R1      ; Contenido de la localidad 7706/7 se transfiere a
          ; R1.
30156      012600   MOV      (SP)+,R0      ; Contenido de la localidad 7710/1 se transfiere a
          ; R0.
30160      012637   MOV      (SP)+,@#7722  ; Contenido de la localidad 7712/3 se transfiere a
          7722          ; localidades 7722/3 (N).
30164      012637   MOV      (SP)+,@#7726  ; Contenido de las localidades 7714/5 se transfiere
          7726          ; a localidades 7726/7.
30170      012637   MOV      (SP)+,@#7724  ; Contenido de las localidades 7716/7 se transfiere
          7724          ; a localidades 7724/5.
30174      012706   MOV      #7000,SP      ; Se reinicializa puntero de pila en 7000.
          7000
30200      4737     JSR      PC,CRLF      ; A subrutina retorno de carro, avance de línea.
          30030
30204      4737     JSR      PC,IMP      ; Consulta: Impresión del nombre de la variable en la unidad I/O.
          30000          ; A subrutina de impresión.
; Respuesta: Operador pulsa secuencia de caracteres, que son impre-
; ; sos y finaliza con carácter retorno de carro. Estos ca
; ; racteres se almacenan en localidades apuntadas por R2.

```



30254	142705	BICB #360,R5	; Se encera los 4 bits más significativos de R5.
	360		; Carácter pulsado es alfabético?
30260	122705	CMPB #13,R5	; Es el contenido de R5 igual que 13?
	13		
30264	1326	BNE ITVN	; Si regresa a ITVN.
			; No continúa: Tecla pulsada corresponde a un cará <u>er</u>
			; ter especial ASCII (51, etc.)
30266	112205	MOVB (R2)+,R5	; Obtención del carácter pulsado.
			; El contenido de la localidad apuntada por R2 se
			; transfiere a R5.
30270	142705	BICB #360,R5	; Se encera los bits más significativos de R5.
	360		; Carácter pulsado es especial (alfabético, numéri-
			; co)?
30274	120527	CMPB R5,#12	; Es el contenido del registro R5 mayor que 12?
	12		
30300	100320	BPL ITVN	; Si, bifurca a ITVN
			; No continúa: Se decrementa en 1 el número de ca-
			; racteres N — N - 1.
30302	5337	DEC @#7720	; El contenido de la localidad 7720 se resta 1
	7720		; Es último carácter digitado?
30306	240	NOP	
30310	240	NOP	
30312	240	NOP	
30314	1352	BNE LAZO	; No regresa a Lazo
			; Si continúa
			; 3.- Se verifica que dígitos ingresados (variable) estén dentro
			; del rango especificado.
			; Obtención de X2.
30316	5302	DEC R2	; El contenido del registro R2 se decrementa en 1.
30320	5302	DEC R2	; El contenido del registro R2 se decrementa en 1.

30322	121203		CMPB (R2),R3	;	Es X2 < D2?
				;	Es el contenido de la localidad apuntada por R2.
				;	menor o igual que el contenido de R3?
30324	101402		BLOS A1	;	Si bifurca a A1.
30326	704		BR ITVN	;	No regresa a ITVN.
30330	240		NOP	;	
30332	121203	A1	CMPB (R2),R3	;	Es X2 = D2?
				;	Es el contenido de la localidad apuntado por R2
				;	igual al contenido de R3?
30334	1005		BNE A2	;	No bifurca a A2.
				;	Si continúa:
				;	Obtención de X1.
30336	5202		INC R2	;	El contenido del registro R2 se incrementa en 1.
				;	Es X1 < D1?
30340	121204		CMPB (R2),R4	;	Es el contenido de la localidad apuntada por R2
				;	menor o igual que el contenido de R4?
30342	101420		BLOS FIN	;	Si bifurca a FIN.
30344	675		BR ITVN	;	No regresa a ITVN.
30346	240		NOP	;	
30350	123712	A2	CMPB @#7726,(R2)	;	Es d2 < X2?
	7726			;	Es el contenido de la localidad 7726 menor o
				;	igual que el contenido de la localidad apuntada
				;	por R2?
30354	101402		BLOS A3	;	Si bifurca a A3.
30356	670		BR ITVN	;	No regresa a ITVN.
30360	240		NOP	;	
30362	123712	A3	CMPB @#7726,(R2)	;	Es d2 = X2?
	7726			;	Es el contenido de la localidad 7726 igual al con_
				;	tenido de la localidad apuntada por R2?
30366	1006		BNE FIN	;	No bifurca a FIN
				;	Si continúa:
				;	Obtención de X1.
30370	5202		INC R2	;	El contenido de R2 se incrementa en 1.
				;	Es d1 < X1?
30372	123712		CMPB @#7724,(R2)	;	Es el contenido de la localidad 7724 menor o
	7724			;	igual que el contenido de la localidad apuntada
				;	por R2.

```

30376 101402      BLOS FIN      ; Si bifurca a FIN.
30400 657         BR ITVN       ; No bifurca a ITVN
30402 240         NOP          ;
30404 207         FIN RTS PC    ; Retorno.

```

```

*****
*****

```

**; SUBROUTINA: VALIDACION DE TRONCALES**

```

; Notación: 0d2d1 < TR0 ≤ 0D2D1
; Argumentos de entrada:
; Registro R2 se carga la dirección de la localidad desde la cual
; se almacenan los dígitos que conforman las troncales.
; En localidades 7701/1 se carga con el carácter ASCII D1.
; En localidades 7702/3 se carga con el carácter ASCII D2
; En localidades 7704/5 dirección desde la cual se almacenan los
; caracteres pulsados.
; En localidades 7706/7 se carga dirección de la localidad en
; la que está el primer carácter de la frase de consulta (TR0:).
; En localidades 7710/1 número de caracteres que conforman la fra
; se.
; En localidades 7712/3 número de dígitos a validar.
; En localidades 7714/5 se carga el carácter ASCII d2.
; En localidades 7716/7 se carga el carácter ASCII d1.
; En localidades 7056 se carga dirección de retorno.
; Argumentos de salida:
; Esta subrutina llama a la subrutina ITVN por lo que los argumen-
; tos de salida son los mismos y además el número de troncales in-
; gresadas se almacena en la localidad 1362.
; Con los argumentos de entrada se verifica que los
; dígitos que pulsó el operador estén dentro del
; rango especificado.

```



30554	012705		MOV	#570,R5					; Se carga en R5 la dirección (570) de la localidad
	570								; en que está almacenado el 1er. dígito de la prime
									; ra troncal digitada contenidas en la tabla TR0.
30560	162702		SUB	#3,R2					; Obtención de la última troncal digitada.
	3								; Se obtiene en R2 la dirección de la localidad en
									; que está almacenado el 1er. dígito de la última
									; troncal de la tabla TR0.
30564	122265		CMPB	(R2)+,0(R5)					; Es Última troncal igual a la troncal digitada?
	0								; Es el 1er. dígito de la última troncal igual al
30570	1010		BNE	A1					; 1er. dígito de la troncal digitada?
									; No, bifurca a A1.
									; Si continúa:
30572	122265		CMPB	(R2)+,1(R5)					; Es el 2do. dígito de la última troncal igual al
	1								; 2do. dígito de la troncal digitada?
30576	1005		BNE	A1					; No, bifurca a A1.
									; Si continúa:
30600	122265		CMPB	(R2)+,2(R5)					; Es el 3er. dígito de la última troncal igual al
	2								; 3er. dígito de la troncal digitada?
30604	1002		BNE	A1					; No bifurca a A1.
30606	735		BR	VALTRO					; Si bifurca a VALTRO.
30610	240		NOP						
30612	062705		ADD	#3,R5					; Obtención de la siguiente troncal digitada.
	3								; El contenido de R5 se incrementa en 3. Se obtiene
									; dirección de la localidad del 1er. dígito de la
									; siguiente troncal.
30616	010302		MOV	R3,R2					; Obtención de la última troncal.
									; El contenido de R3 se transfiere a R2.
30620	077421		SOB	R4,LAZO					; N — N - 1 y ¿N = 0?
									; Se decrementa en 1 el contenido de R4. Si es dife
									; rente a cero bifurca a LAZO; si es igual a cero
									; continúa en la próxima instrucción.
30622	240		NOP						
30624	012706		MOV	#7056,SP					; Puntero de pila se carga con dirección de la loca
	7056								; lidad en que está almacenada dirección de retorno
30630	207		RTS	PC					; Retorno.

**;SUBROUTINA:IMPRESION**

; Argumentos:  
; Entrada : En R0 se carga el número (cantidad) de caracteres a imprimirse,  
; los mismos que deben estar en código ASCII y cada uno almacenado  
; en una localidad de un byte .  
; En R1 se carga la dirección de la localidad en que está almacenado el primer carácter a imprimirse.  
; Salida : En la unidad de I/O se imprimen el contenido de la localidad apuntada por R1; finalizando con el número de caracteres indicados en R0.

30000	240	IMP	NOP		
30002	105737	LIST	TSTB @#TPS		: Listo impresor?
30006	100375	BPL	LIST		; Se comprueba que registro TPS (177564) esté listo
					; para que imprima el contenido del registro TPK.
					; No, regresa a list
					; Si, continúa :
					; Impresión del contenido de la localidad apuntada
					; por R1.
30010	112137	MOVB	(R1)+,@#TPB		; EL contenido de la localidad apuntada por R1 se
	177566				; transfriere al registro TPB (177566).
30014	077006	SOB	R0,LIST		; Ultimo carácter a ser impreso?
30016	207	RTS	PC		; EL contenido de R0 se decrementa en 1. Si es diferente de 0 bifurca a LIST, caso de ser 0 continúa
					; Retorno.

\*\*\*\*\*  
\*\*\*\*\*

**; SUBROUTINA: RETORNO DE CARRO, AVANCE DE LINEA**

; Argumentos:  
; Salida : El carro del impresor de la unidad I/O se coloca en la primera columna y avanza a la siguiente línea .



```

30030 105737 CRLF TSTB @#TPS
      177564
      100375 BPL CRLF
; Listo Impresor?
; Se comprueba que registro TPS (177564) esté listo
; para que se imprima el contenido del registro TPK
; No, regresa a CRLF.
; Si, continúa:
; Se ejecuta, en I/O, retorno de carro.
30036 112737 MOVB #15,@#177566
      15
      177566
; Se transfiere a registro TPB (177566) carácter
; retorno de carro (15).

30044 105737 LF TSTB @#TPS
      177564
      100375 BPL LF
      45000
; Listo Impresor?
; Se comprueba que registro TPS (177564) esté listo
; para que se imprima el contenido del registro TPK
; No, regresa a LF.
; Si, continúa :
; Se ejecuta en la unidad de I/O avance de línea.
30052 112737 MOVB #12,@#177566
      12
      177566
; Se transfiere al registro TPB (177566), carácter
; avance de línea (12).

30060 207 RTS PC
; Retorno.

```

```

*****
*****

```

; SUBROUTINA : CONSULTA

```

; Argumentos:
; Entrada : En R2 se carga dirección de la localidad, desde la cual se almace-
;           nan los caracteres ingresados por teclado. Se finaliza ingreso al
;           pulsar tecla retorno de carro.
; Salida  : Los caracteres pulsados en el teclado se almacenan en las localida-
;           des apuntadas por R2 y son impresos en la forma como se los va
;           digitando.
; El contenido de R2 indica la dirección de la localidad en la que
; está almacenado el último carácter ingresado.

```

```

30070 105737 RESP TSTB @#TKS
177560
; Listo teclado?
; Se comprueba que registro TKS (177560) esté listo
; para que el registro TKB reciba el carácter pulsa
; do.
30074 100375 BPL RESP
; No, regresa a RESP
; Si, continúa:
; Digitador pulsa tecla que corresponde a carácter
; retorno de carro (<)?
30076 123727 CMPB @#TKB,#2L5
177562
; Es el contenido del registro TKB (177562) igual
; al valor 2L5?.
30104 1412 BEQ SALIDA
; Si, bifurca a SALIDA.
; No, continúa:
; Se almacena en memoria carácter pulsado.
30106 113722 MOVB @#TKB,(R2)+
177562
; El contenido del registro TKB(177562) se transfiere a la
; localidad de memoria apuntada por R2.
; Impresión:
; Listo Impresor?
30112 105737 LIST TSTB @#FPS
177564
; Se comprueba que registro FPS(177564) esté listo
; para que se imprima contenido del registro TPK.
30116 100375 BPL LIST
34220
; No, regresa a LIST.
; Si, continúa:
; Se imprime el carácter contenido en registro TKB
30120 113737 MOVB @#177562,
177562 @#177566
177566
; El contenido del registro TKB (177562) se transfiere al
; registro TPK(177566).
30126 137 JSR PC,RESP
30070
; Regreso a RESP.
30132 207 RTS PC
; Retorno.

```

```

*****
*****

```

**; SUBROUTINA: DETECCION DE FECHA.**

; Notación: DMS(F) Dígito más significativo del dato minutos del área F

**; Argumentos:**

Entrada : R3 (Puntero P1) se carga con la dirección de la primera localidad del área de memoria F que contiene la fecha a ser comparada.  
R2 (Puntero P2) se carga con la dirección de la primera localidad del área de memoria CAMPO PEDIDO DE LLAMADA (CPLL).  
Las fechas contenidas en las dos áreas están en el orden: año, mes día, hora, y minutos; y los caracteres en código EBCDIC y cada área ocupa cinco localidades de 2 bytes.

R6 (Puntero de pila) se carga con el valor 10000.  
Localidades 7776/7 se carga con la dirección "si"  
Localidades 7774/5 se carga con dirección "no"  
Salida: Si la fecha contenida en el Area CPLL es igual a la fecha contenida en el Area F, la subrutina bifurca a la dirección contenida en localidad 7776/7; caso contrario a la dirección contenida en localidad 7774/5.

31210	240	DEFEC	NOP				
31212	240		NOP				
31214	010200		MOV	R2,R0			; Obtención de P1 y P2.
31216	010301		MOV	R3,R1			; El contenido del registro R2 se transfiere a R0
							; El contenido del registro R3 se transfiere a R1
							; Inicialización del contador utilizado en los pri
							; meros 4 datos de la fecha:
							; CON _____ 10
31220	012737		MOV	#10,@#212			; Localidad 212 se carga con el valor 10.
	10						
	212						

; Se compara si cada dígito del año, mes, día, hora y minutos del campo pedido de llamada es igual a cada uno de los dígitos correspondientes al Area F .  
 ; Es el contenido de la localidad apuntada por R0 igual al de la apuntada por R1 ?  
 ; No bifurca a N.  
 ; Si continúa:  
 ; CON ——— CON - 1  
 ; Se decrementa en 1 el contenido de la localidad 212.  
 ; Es CON = 0?  
 ; No, regresa a LAZO.  
 ; Si, continúa:  
 ; ¿DMS(F) > DMS(C)?  
 ; Es el contenido de la localidad apuntada por R1 mayor que la de la apuntada por R0?  
 ; Si, bifurca a N.  
 ; No, continúa :  
 ; ¿DMS(F) = DMS(C)?  
 ; Es el contenido de la localidad apuntada por R1 igual que a la apuntada por R0.  
 ; Si, bifurca a A  
 ; No, continúa :  
 ; Obtención de DMS(F) y DMS(C).  
 ; Se decrementa en 1 el contenido de R0 .  
 ; Se decrementa en 1 el contenido de R1  
 ; Se incrementa en 1 el contenido de la localidad apuntada por R1.  
 ; ¿DMS(F) = DMS(C)?  
 ; Es el contenido de la localidad apuntada por R1 igual al de la apuntada por R0.  
 ; No, regresa a LAZ1  
 ; Si, continúa:

31226	122021	LAZO	CMPB (R0)+,(R1)+
31230	1036	BNE N	
31232	5337 212	DEC @#212	
31236	1373	BNE LAZO	
31240	121110	CMPB (R1), (R0)	
31242	101031	BLOS N	
31244	122120	CMPB (R1)+,(R0)+	
31246	1415	BEQ A	
31250	5300	DEC R0	
31252	5301	DEC R1	
31254	105211	LAZ1 INCB (R1)	
31256	121110	CMPB (R1), (R0)	
31260	1375	BNE LAZ1	

31262	5200	INC R0	; Obtención de dms(F) y dms(C)
31264	5201	INC R1	; Contenido de R0 se incrementa en 1
31266	240	NDP	; Contenido de R1 se incrementa en 1
31270	240	NDP	
31272	112711	MOVB #360, (R1)	; dms(F) — 0(EBCDIC)
	360		; Se carga la localidad apuntada por R1 con el va-
31276	240	NDP	; lor 360.
31300	240	NDP	
31302	121110	A CMPB (R1), (R0)	; ¿ dms(F) > dms(C)?
31304	101010	BLOS N	; Es el contenido de la localidad apuntada por R1
	1362		; mayor que la apuntada por R0
31306	121110	CMPB (R1), (R0)	; Si, bifurca a N.
31310	1403	BEQ S	; No, continúa :
			; ¿ dms(F) = dms(C)?
			; Es el contenido de la localidad apuntada por R1
			; igual que la apuntada por R0.
			; Si, bifurca a S.
			; No, continúa :
31312	105211	LAZ2 INCB (R1)	; dms(F) — dms(F) + 1
			; Se incrementa en 1 el contenido de la localidad
			; apuntada por R1.
31314	121110	CMPB (R1), (R0)	; ¿dms(F) = dms(C)?
31316	1375	BNE LAZ2	; Es el contenido de la localidad apuntada por R1
			; igual que la apuntada por R0
			; No, bifurca a LAZ2
			; Si, continúa :
31320	5206	S DEC SP	; Obtención de dirección "si"
31322	5206	DEC SP	; Se decrementa en 1 el contenido del puntero de
			; pila.
			; Se decrementa en 1 el contenido del puntero de
			; pila.
31324	402	BR SALIDA	; Salto a salida.
31326	062706	N ADD #4, SP	; El contenido del registro R6 se incrementa en 4.
			4
31332	207	SALIDA RTS PC	; Retorno.

; SUBROUTINA: COMPARACION DE FECHA (CPLL > F)

; Notación: CPLL y F denotan áreas de memoria en las que están contenidas, en código EBCDIC, los caracteres que representan las fechas a ser comparadas. Los datos de cada fecha son y están en el orden: año, mes, día, hora, minutos; en que cada dato está formado por dos dígitos y cada uno contenido en una localidad de un byte; el dígito más significativo ocupa el Low Byte y el más significativo el High Byte

; Argumentos:  
 ; Entrada : R3 (Puntero P1) se carga con la dirección de la primera localidad del área de memoria F.  
 R2 (Puntero P2) se carga con la dirección de la primera localidad del área de memoria CPLL.  
 R6 (Puntero de pila) se carga con el valor 10000.  
 Localidades 7776/7 se carga con dirección "No".  
 Localidades 7774/5 se carga con dirección "Si"  
 Salida : Si la fecha contenida en el área CPLL es mayor o igual que la fecha contenida en el área F, la subrutina bifurca a la dirección contenida en localidades 7774/5, caso contrario a la dirección contenida en la localidad 7776/7.

; Inicialización del contador de dígitos.

; I = 5

; Localidad 214 se carga con el valor 5

32276 012737 CMPFE MOV #12,@#214

5

214

32304 240 NOP

32306 010205 MOV R2,R5

32310 010301 MOV R3,R1

; Se obtienen punteros P1 y P2.

; El contenido de R2 se transfriere a R5.

; El contenido de R3 se transfriere a R1

; Se obtienen DAT(F) y DAT(C).

; El contenido de la localidad apuntada por R5 se

; transfriere a localidades 2200/1

32312 012537 LAZO MOV (R5)+,@#2200

2200

32316	012737 2202	MOV (R1)+, @#2202	; El contenido de la localidad apuntada por R1 se ; transfiere a localidades 2200/1.
32322	337 2200	SWAP @#2200	; Se ordenan los dígitos de cada dato.
32326	337 2202	SWAP @#2202	; Se intercambia el Low byte y High byte de la loca ; lidad 2200/1
32332	023737 2200 2202	CMP @#2200, @#2202	; Se intercambia el Low byte y High byte de la loca ; lidad 2202/3
32340	101007	BHI SALL	; DATF(I) > DATC(I)?
32342	023737 2200 2202	CMP @#2200, @#2202	; Es el contenido de la localidad 2200 mayor que el ; de la localidad 2202 ?
32350	1006	BNE SAL2	; Si, salta a SALL .
32352	5337 214	DEC @#214	; No, continúa: ; ¿DATF(I) = DATC(I)?
32356	1355 40	BNE LAZO	; Es el contenido de la localidad 2200 igual al de ; la localidad 2202.
32360	5206	DEC SALL	; No, bifurca a SAL2.
32362	5206	DEC SP	; Si, continúa: ; I = I - 1
32364	402	BR SALIDA	; Se decrementa en 1 el contenido de la localidad ; 214 .
32366	062706 4	SAL2 ADD #4, SP	; ¿ I = 0?
32372	207	SALIDA RTS PC	; No, regresa a LAZO ; Si, continúa : ; Se obtiene dirección "no" ; Se decrementa en 1 el contenido del registro R6 ; Se decrementa en 1 el contenido del registro R6 ; Salto a SALIDA. ; Se obtiene dirección "si" ; Se incrementa en 4 el contenido de R6

**; SUBROUTINA: COMPARACION DE TRONCAL(CTRO=TR0(I)**

**; Notación:** TR0 representa el área de memoria en la que están almacenados los caracteres EBCDIC que forman las troncales; cada troncal de tres dígitos y cada uno ocupa una localidad de un byte.  
CPTR0 es el conjunto de tres localidades que contienen los dígitos de cada troncal.  
CTR contador de troncales .  
I = 1,...N = número de troncales  
TR0 (I) troncales almacenadas en tablas TR0  
J contador de dígitos = 1,2,3

**; Argumentos:**

**Entrada :** R4(Puntero P1) se carga con la dirección de la primera localidad del área de memoria TR0

R2 (Puntero P2) se carga con la dirección de la primera localidad del área de memoria CPLL

Localidad 1362 se carga con el valor 0

Se carga puntero de pila con valor 10000

Localidades 20774/5 se carga con dirección de retorno

**Salida :** Si, CTRO = TR0(I) la subrutina bifurca a efectuar el proceso de Acumulación, luego de barrer con todas las troncales en la tabla TR0 se retorna al programa principal.



32174	5237 2300	CMTRO	INC	@#2300	;	I = I + 1
					;	Se incrementa en 1 el contenido de la localidad 2300
32200	010401	MOV		R4, R1	;	Se obtienen P1 y P2 .
32202	010205	MOV		R2, R5	;	El contenido del registro R4 se transfiere a R1
					;	El contenido del registro R2 se transfiere a R5
32204	062705 102	ADD		#102, R5	;	Se obtiene campo troncal saliente CTR0
					;	El contenido de R5 se incrementa en 102.
32210	012737 3 216	MOV		#3, @#216	;	J — 3
					;	Se carga localidad 216 con el valor 3
32216	122521	LAZO	CMPB	(R5)+, (R1)+	;	¿DT(J) = DC(J)?
					;	Es el contenido de la localidad apuntada por R5 igual al de la apuntada por R1?
32220	1005	BNE		A1	;	No, bifurca a A1
					;	Si, continúa :
32222	5337 216	DEC		@#216	;	J — J - 1
					;	Se decrementa en 1 el contenido de localidad 216
32226	1373	BNE		LAZO	;	¿ J = 0?
					;	No, regresa a LAZO
32230	4737 47600	JSR		PC, ACU	;	Si, continúa :
					;	A proceso de Acumulación
32234	240	NOP			;	A subrutina de Acumulación
32236	240	NOP			;	
32240	023737 2300 1362	CMP		@#2300, @#1362	;	¿ I = N?
					;	Es el contenido de la localidad 2300 igual al de la 1362?
32246	1403	BEQ		A1	;	Si, bifurca a A1
					;	No, continúa:



```

; SUBROUTINA: PROCESO ITERATIVO DE DECISION (S/N)
; Argumentos de entrada:
; En registro R5 se carga el número de caracteres a ser impresos.
; En registro R3 se carga la dirección de la localidad del primer carácter.
; En registro SP se carga el valor 10000.
; En localidad 7776/7 cargar dirección a que bifurcará el programa al pulsar
; tecla S.
; En localidad 7774/5 cargar dirección a que bifurcará el programa al pulsar
; tecla N.
; Argumentos de salida:
; Se imprime en la unidad de I/O el número de caracteres indicados en R5, ini-
; ciando la impresión con el contenido de la localidad direccionado por R3.
; El operador al pulsar la tecla S bifurca a la dirección contenida en la locali-
; dad 7776/7.
; El operador al pulsar la tecla N bifurca a la dirección contenida en la locali-
; dad 7774/5.
; Al pulsar otras teclas se repite el proceso.

30410 4737 ITDSN JRS PC,CRLF ; A subrutina retorno de carro, avance de línea.
      30030
30414 4737 JSR PC,CRLF ; A subrutina retorno de carro, avance de línea.
      30030
30420 010500 MOV R5,R0 ; Impresión de consulta (caracteres).
30422 240 NOP ; Se obtiene número de caracteres. El contenido
30424 010301 MOV R3,R1 ; de R5 se transfriere a R0.
30426 240 NOP ; Se obtiene dirección de la localidad del ler. ca-
30430 4737 JSR PC,IMP ; rácter. El contenido de R3 se transfriere a R1.
      30000 ; A subrutina de impresión
; Respuesta
; Se guarda carácter pulsado en localidad 1200
; Registro R2 se carga con la dirección 1200
30434 012702 MOV #1200,R2
      1200

```

```

30440 4737 JSR PC,RESP ; A subrutina de respuesta.
      30070
30444 123727 CMPB @#1200,#323 ; Operador pulsa tecla S?
      1200 ; El contenido de la localidad 1200 es igual al ca-
      323 ; rácter ASCII S(323)?
30452 1407 BEQ S1 ; Si bi furca a S1
      ; No continúa:
      ; Operador pulsó tecla N?
30454 123727 CMPB @#1200,#316 ; Es el contenido de la localidad 1200 igual al ca-
      1200 ; rácter ASCII N (316)?
      316
30462 1352 BNE ITDSN ; No regresa a ITDSN
      ; Si continúa.
30464 012706 MOV #7774,SP ; En SP se carga dirección de la localidad en la
      7774 ; que está almacenada dirección representada por N.
30470 402 BR FIN ; A FIN.
30472 012706 S1 MOV #7776,SP ; En SP se carga dirección de la localidad en la
      7776 ; que está almacenada dirección representada por S.
30474 207 FIN RTS PC ; Retorno.

```

```

*****
*****

```

; SUBROUTINA: DETECCION EN CINTA MAGNETICA DE LA FECHA INICIAL INGRESADA

```

; Argumentos de entrada:
; En el área de memoria FII (localidades 500 a 512), se almacenan en el si-
; guiente orden los datos que representan la fecha inicial: Mes, Día, Año, Ho-
; ra y Minutos. Cada dato formado por dos caracteres EBCDIC, en que cada dígi-
; to almacenado en una localidad de un byte.
; Cargar en localidad 206 el valor 0.
; En localidad 20776 cargar dirección de retorno.

```

```

; Argumentos de salida:
; En localidad 206 se almacena el número de bloques leídos.
; En caso de estar grabada la fecha contenida el el área FII en la cinta magnética la subrutina regresa al programa principal; caso contrario se imprime en la unidad I/O la leyenda: "Favor ingresar otra cinta"; y el operador deberá pulsar la tecla retorno de carro (<).
; Salto de bloque cero.
30640 013700 DTSEC MOV @#2610,R0
2610
; R0 se carga con dirección (2610) de la localidad en la que está almacenada la dirección del registro de estado de la unidad de Cinta Magnética (UCM).
; R1 se carga con dirección (2612) de la localidad en la que está almacenada la dirección del registro de Comando de UCM.
; R2 se carga con dirección (2614) de la localidad en la que está almacenada la dirección del registro Contador de Bytes de UCM.
; R3 se carga con dirección (2616) de la localidad en la que está almacenada la dirección del registro Direcciones de la UCM.
; A subrutina de Salto de Bloque 0.
30644 013701 MOV @#2612,R1
2612
30650 013702 MOV @#2614,R2
2614
30654 013703 MOV @#2616,R3
2616
30660 4737 JSR PC,SB0
31072
30664 240 NOP
30666 240 NOP
30670 240 NOP
30672 240 NOP
30674 240 NOP
30676 240 NOP

30700 013701 MOV @#2612,R1
2612
; Lectura de bloque: El contenido de un bloque de Cinta Magnética se transfiere a las localidades 3000 a 5000 de la Memoria Principal (1 k byte).
; R1 se carga con dirección (2612) de la localidad en la que está almacenada la dirección del registro de Comando de la UCM.

```

30704	013702 2614	MOV	@#2614,R2	; R2 se carga con dirección (2614) de la locali- ; dad en la que está almacenada la dirección del re- ; gistro Contador de Bytes de UCM.
30710	013703 2616	MOV	@#2616,R3	; R3 se carga con dirección (2616) de la locali- ; dad en la que está almacenada la dirección del re- ; gistro de Direcciones de la UCM.
30714	4737 31400	JSR	PC,LECBL	; A subrutina de lectura de bloque.
30720	240	NOP		
30722	240	NOP		
30724	240	NOP		
30726	012701 177777	MOV	#177777,R1	; Retraso ; R1 se carga con valor 177777.
30732	077101	SOB	R1,H	; El contenido de R1 se decrementa en 1. Si R1 es ; diferente a cero se bifurca a H, si es cero con- ; tinúa en la próxima instrucción.
30734	012702 3100	MOV	#3100,R2	; Inicialización del puntero que señala las locali- ; dades que conforman el campo pedido de llamada: ; P2 — 3100.
30740	012737 14 210	MOV	#14,@#210	; R2 se carga con el valor 3100. Dirección de la ; lra. localidad en que se localiza el ler. campo. ; Localidad 210 se carga con valor 14.
30746	012705 500	MOV	#500,R5	; Carga de argumentos.
30752	010503	MOV	R5,R3	; En R5 se almacena dirección (500) de la lra. loca- ; lidad del área de memoria en la que se encuentra ; la fecha inicial ingresada (FII). ; El contenido del registro R5 se transfiere al re- ; gistro R3.
30754	012706 10000	MOV	LAZ01 #10000,SP	; ¿Campo Pedido de Llamada es igual a la fecha con- ; tenida en FII? ; Inicialización del puntero de pila en 10000.

30760	012746	MOV	#30774, (-SP)				; No: bifurca a A (En puntero de pila se carga dirección 30774 que representa A).
30764	013746	MOV	@#20774, (-SP)				; Si: retorno a programa principal (puntero de pila se carga con dirección (20774) de la localidad en la que está almacenada dirección de retorno.
30770	4737	JSR	PC, DEFEC				; A subrutina de Detección de Fecha.
	31210						
30774	062702	MOV	#120, R2	A			; Obtención de la dirección local de la localidad del siguiente registro correspondiente al Campo Pedido de Llamada.
	120						; El contenido del registro R2 se añade 120, el resultado se almacena en dicho registro.
31000	5337	DEC	@#210				; ¿ Último registro ? (CR =0)
	210						; Se decrementa el contador de registros en 1.
							; (CR ——— CR - 1).
31004	1363	BNE	LAZ01				; Es CR = 0 ?
31006	5237	INC	@#206				; No a LAZ01
	206						; Si es cero continúa en la próxima instrucción.
31012	013700	MOV	@#2610, R0				; Contador de bloques se incrementa en 1
	2610						; (CB ——— CB + 1).
31016	032710	BIT	#6200, (R0)				; Es fin de cinta (EOF)?
	6200						; El registro R0 se carga con dirección (2610) de la localidad en la que está almacenada la dirección del registro de estado de la UCM.
31022	1720	BEQ	LAZ0				; ¿ El registro de estado (direccionado por R0) contiene el código correspondiente a EOF?
							; No a LAZ0;
							; Si continúa:
31024	013701	MOV	@#2610, R1				; Retroceder Cinta a punto de carga e impresión de frase: "Favor colocar otra cinta" (<).
	2610						; En R1 se carga dirección (2610) de la localidad en la que está almacenada dirección del Registro de Estado de la UCM.
31030	4737	JSP	PC, CPCIM				; A subrutina de Punto de Carga.
	31450						

31034 713 BR DTFC ; A reiniciar proceso de búsqueda

\*\*\*\*\*  
\*\*\*\*\*

; SUBROUTINA INCREMENTO EN 1 HORA DE FECHA

; Argumentos de entrada:  
; En la localidad 7006/7 cargar dirección de retorno.  
; En el área de memoria FCHC (localidades 31340 a 31350) deben estar  
; almacenados los datos que representan la fecha en el orden mes, día,  
; año, hora y minuto. Cada dato está constituido de dos caracteres nu-  
; méricos EBCDIC y cada dígito en una localidad de memoria de 1 byte.  
; Argumentos de salida:  
; En las localidades que conforman FCHC se obtiene la fecha anterior  
; incrementado en 1 hora y los datos en el mismo orden y secuencia ori-  
; ginales.

; Los datos del área FCHC se ponen en el orden Año,  
; Mes, Día, Hora y Minuto.  
; En R0 se carga dirección del 3er. dato (Año) a  
; ser colocado como primer dato en el área FCHC.  
; A subrutina de ordenamiento.

35200	012700	INCFE	MOV	#31344,R0
	31344			
35204	4737		JSR	PC,ORD
	31530			
35210	240		NOP	
35212	240		NOP	
35214	012701		MOV	#31340,R1
	31340			
35220	4737		JSR	PC,INTER
	32400			
35224	240		NOP	
35226	240		NOP	

; Se intercambia el contenido de dos localidades ad-  
; yacentes de 1 byte.  
; En R1 se carga dirección (31340) de la 1ra.locali-  
; dad del área de memoria FCHC.  
; A subrutina de intercambio.





47600	012706	MOV	#7000,R6	; Almacenamiento en las localidades 6770 a 6776 de
	7000			; los punteros que direccionan los campos utiliza-
				; dos en el proceso.
				; Se inicializa puntero de pila en 7000
47604	010146	MOV	R1,-(SP)	; Contenido de R1 se transfiere a localidad 6776/7
47606	010246	MOV	R2,-(SP)	; Contenido de R2 se transfiere a localidad 6774/5
47610	010446	MOV	R4,-(SP)	; Contenido de R4 se transfiere a localidad 6772/3
47612	010546	MOV	R5,-(SP)	; Contenido de R5 se transfiere a localidad 6770/1
47614	240	NOP		
47616	012700	MOV	#76,R0	; Encerado de Buffer Principal (Localidades 2400 a
	76			; 2554).
				; En R0 se carga número (76) de palabras de 2 bytes
				; a ser encerados.
47622	012701	MOV	#2400,R1	; En R1 se carga dirección de la primera localidad
	2400			; del Buffer.
47626	4737	JSR	PC,ENC	; A subrutina de encerado.
	32520			
47632	240	NOP		
				; Transferencia de los campos 17, 18 y 19 (Tiempos de Establecimiento y
				; Tasación) a las áreas T1, T2, TT de Buffer Principal (Localidades T1:
				; 2400 a 2404, T2: 2406 a 2412, TT: 2414 a 2424).
47634	4737	JSR	PC,TRTBP	; A subrutina de transferencia de Campos a Buffer
	32660			; Principal (localidades 2400 a 2424).Area temporal
				; C16: 2400 a 2404, C17: 2406 a 2412, C18: 2414 a
				; 2424 en el orden original los dígitos de cada cam
				; po, cada dígito ocupa una palabra de 2 bytes.
47640	240	NOP		
				; El campo 17 (T1) contenido en Buffer Principal se
				; transfiere a Buffer Auxiliar (localidades 40150 a
				; 40160).
				; Los dígitos del tiempo se ordenan en el Buffer BA
				; desde el Dms al DMS en que cada dígito es una pa-
				; labra de 2 bytes.
47642	012700	MOV	#3,R0	; En R0 se carga número (3) de palabras a ser orde-
	3			; nados.

47646	012701	MOV	#2406, R1	; En R1 se carga dirección (2406) de la localidad
	2406			; que permite localizar el dígito menos significati
47652	4737	JSR	PC, ORTBA	; vo del campo 17 (T1).
47656	32710			; A subrutina de ordenamiento y transferencia a
	240	NOP		; Buffer Auxiliar.
47660	012701	MOV	#40150, R1	; El contenido del Buffer Auxiliar se transfiere a
	40150			; Buffer Principal(Campo 17 distribuido en el campo
47664	012703	MOV	#2400, R3	; de memoria T1 del Buffer Principal, T1: localida-
	2400			; des 2400 a 2404.
47670	012704	MOV	#3, R4	; En R1 se carga dirección (40150) de la primera lo
	3			; calidad del Buffer Auxiliar.
47674	4737	JSR	PC, TRANS	; En R3 se carga dirección (2400) de la lra. locali
	32474			; dad del área de memoria T1 del Buffer Principal.
47700	240	NOP		; En R4 se carga número (3) de palabras de 2 bytes
				; a ser transferidos.
				; A subrutina de transferencia.
47702	012700	MOV	#3, R0	; El campo 18 (T2) contenido en Buffer Principal se
	3			; transfiere a Buffer Auxiliar. Los dígitos del
47706	012701	MOV	#2414, R1	; tiempo se ordenan desde Dms al DMS en que cada dí
	2414			; gito es una palabra de 2 bytes (2 localidades de
				; memoria).
47712	4737	JSR	PC, ORTBA	; En R0 se carga número (3) de palabras (2 bytes)
	32710			; a ser ordenados.
47716	240	NOP		; En R1 se carga dirección (2414) de la localidad
				; que permite localizar el dígito menos significati
				; vo del campo 18 (T2).
				; A subrutina de ordenamiento y transferencia.

47720	012701	MOV	#40150, R1	; El contenido del Buffer Auxiliar se transfiere a
	40150			; Buffer Principal (Campo 18 a área de memoria T2,
47724	012703	MOV	#2406, R3	; localidades 2406 a 2412 del Buffer Principal.
	2406			; En R1 se carga dirección (40150) de la primera lo
47730	012704	MOV	#3, R4	; calidad del Buffer Auxiliar.
	3			; En R3 se carga dirección (2406) de la lra. locali
47734	4737	JSR	PC, TRANS	; dad del área de memoria T2 del Buffer Principal.
	32474			; En R4 se carga número (3) de palabras (2 bytes) a
47740	240	NOP		; ser transferidos
				; A subrutina de transferencia.
47742	012700	MOV	#5, R0	; El campo 19 (TT) contenido en el Buffer Principal
	5			; se transfiere a Buffer Auxiliar. Los dígitos del
47746	012701	MOV	#2426, R1	; tiempo se ordenan en el Buffer Auxiliar desde el
	2426			; DMS al DMS; en que cada dígito es una palabra de
				; dos bytes.
47752	4737	JSR	PC, ORTBA	; En R0 se carga número (5) de palabras (2 bytes) a
	32710			; ser ordenados.
47756	240	NOP		; En R1 se carga dirección (2426) de la localidad
				; que permite localizar el dígito menos significati
				; vo del campo 19 (TT).
				; A subrutina de ordenamiento y transferencia.
47760	012701	MOV	#40150, R1	; El contenido del Buffer Auxiliar se transfiere al
	40150			; Buffer Principal.
47764	012703	MOV	#2414, R3	; Campo 19 (TT) a área de memoria TT, localidades
	2414			; 2416 a 2424 del Buffer Principal.
47770	012704	MOV	#5, R4	; En R1 se carga dirección (40150) de la lra. loca-
	5			; lidad del Buffer Auxiliar.
				; En R3 se carga dirección (2424) de la lra. locali
				; dad del área de memoria TT de Buffer Auxiliar.
				; En R4 número (5) de palabras (2 bytes) a ser
				; transferidos.

47774	4737	JSR	PC,TRANS	; A subrutina de transferencia.
	32474			
50000	240	NOP		
50002	4737	JSR	PC,ACBTL	; Activación de Bandera destinada al tipo de llama
	32540			; da.
50006	240	NOP		; Si el campo 09, tipo de llamada, contiene el código
				; go EBCDIC "B" (llamada efectiva) coloca un 1 en
				; localidad 2426; caso contrario un cero.
				; A subrutina de activación de Bandera Tipo de llama
				; mada
				; Activación de Banderas para los códigos Razón de
				; Desconexión.
				; Si el campo 11, Razón de Desconexión de Llamada,
				; es igual a uno de los códigos almacenados en la
				; tabla CRD (localidad 100000 a 100242 se coloca en
				; set (1) la bandera correspondiente para dicho código
				; del buffer Principal; caso contrario se pone
				; en set la bandera bnll (localidad 2427).
50010	4737	JSR	PC,ACBCR	; A subrutina de Activación de Banderas para Código
	32600			; gos Razón de Desconexión
				; Se convierten los caracteres numéricos que repre
				; sentan los tiempos T1, T2 y T3 a código BCD.
50014	240	NOP		
50016	4737	JSR	PC,VEBCD	; A subrutina de conversión de EBCDIC a BCD.
	33000			
50022	012704	MOV	#40000,R4	; Carga de direcciones base en área de memoria "Di-
	40000			; recciones" (localidades 40000 a 40022). Se almace
50026	012724	MOV	#100360,(R4)+	; nan las direcciones base de los siguientes datos
	100360			; T1, T2, T3, T4, NLL, NLLI, CRD, TSUTE,
				; TSUTI.
				; Registro R4 se carga con dirección (40000) de
				; la 1ra. localidad del área Direcciones.
50032	012724	MOV	#100400,(R4)+	; Se almacena en localidades 40000/1 la dirección
	100400			; 100360 (T1).
				; Se almacena en localidades 40002/3 la dirección
				; 100400 (T2).

50036	012724	MOV #100420, (R4)+	; Se almacena en localidades 40004/5 la dirección
	100420		; 100420 (TT).
50042	012724	MOV #100444, (R4)+	; Se almacena en localidades 40006/7 la dirección
	100444		; 100444 (Tj).
50046	012724	MOV #100472, (R4)+	; Se almacena en localidades 40010/1 la dirección
	100472		; 100472 (NTLL).
50052	012724	MOV #100474, (R4)+	; Se almacena en localidades 40012/3 la dirección
	100474		; 100474 (NTLE).
50056	012724	MOV #100476, (R4)+	; Se almacena en localidades 40014/5 la dirección
	100476		; 100476 (NLLI).
50062	012724	MOV #100540, (R4)+	; Se almacena en localidades 40016/7 la dirección
	100540		; 100540 ((CRD).
50066	012724	MOV #100500, (R4)+	; Se almacena en localidades 40020/1 la dirección
	100500		; 100500 (TSUTE).
50072	012724	MOV # 100520, (R4)+	; Se almacena en localidades 40022/3 la dirección
	100520		; 100520 (TSUTI).
50076	012706	MOV #10000, SP	; Se suman los valores almacenados en las áreas de
	10000		memoria: T1, T2, Tj, TSUTE, TSUTI, NLLT, NLLI,
50102	4737	JSR PC, INCTA	; NLLI, CRD localizados en las tablas de totales
	50172		; con los correspondientes del Buffer Principal. El
50106	240	NOP	; resultado se almacena en las tablas.
			; Se inicializa puntero de pila en 10000.
			; A subrutina de Incremento de Tablas.
50110	013700	MOV @#2300, R0	; En área de memoria "Direcciones" se almacenan
	2300		; direcciones bases de la tabla de troncal TR0 (I)
50114	012706	MOV #10000, SP	; en proceso. Se determina el subíndice I=1..N
	10000		; En R0 se carga el número I de la troncal conteni
50120	012746	MOV #12, -(SP)	; da en la localidad 2300.
	12		; Se inicializa puntero de pila en 10000.
50124	012746	MOV #40000, -(SP)	; En localidad 7776/7 se carga el valor 12 (Número
	40000		; de localidades de 2 bytes).
50130	012746	MOV #500, -(SP)	; En localidad 7774/5 se carga el valor 40000 (di-
	500		; rección de la lra. localidad del área Direcciones
			; En localidad 7772/3 se carga el valor 500 (número
			; de localidades que están separadas cada tabla de
			; troncal).

```

50134 4737 JSR PC, ACTDIR ; A subrutina de actualización de Tablas de direc-
      33210 ; ciones.
50140 240 NOP

50142 012706 MOV #10000, SP ; Se inicializa puntero de pila en 10000.
      10000
50146 4737 JSR PC, INCTA ; A subrutina de Incremento de Tablas.
      50172

50152 012706 MOV #6770, SP ; Restitución de los punteros al programa principal
      6770 ; Se inicializa puntero de pila en 6770.
50156 012605 MOV (SP)+, R5 ; Contenido de 6770/1 se transfiere a registro R5.
50160 012604 MOV (SP)+, R4 ; Contenido de 6772/3 se transfiere a registro R4.
50162 012602 MOV (SP)+, R2 ; Contenido de 6774/5 se transfiere a registro R3.
50164 012601 MOV (SP)+, R1 ; Contenido de 6776/7 se transfiere a registro R1.
50166 137 JMP PC, RET ; Retorno
      32234
*****
*****

```

**; SUBROUTINA: INCREMENTO**

```

; Argumentos:
; Entrada : Cargar puntero de pila con el valor 10000
;           Area de memoria "Direcciones" (localidades 40000 a 40022)
;           debe contener los siguientes valores:
;           Localidad 40000/1 con la dirección de la lra. localidad del
;           área de memoria T1.
;           Localidad 40002/3 con la dirección de la lra. localidad del
;           área de memoria T2.
;           Localidad 40004/5 con la dirección de la lra. localidad del
;           área de memoria T1.
;           Localidad 40006/7 con la dirección de la lra. localidad del
;           área de memoria Tj. (Tj = T1 + T2 + T1).

```

50172	012700	MOV	#13, R0	; Encerado de Buffer Auxiliar (localidades 40150 a 40174).
	13			
50176	012701	MOV	#40150, R1	; En R0 se carga número de palabras (13) de 2 bytes a ser encerados.
	40150			
50202	4737	JSR	PC, ENC	; En R1 se carga dirección (40150) de la primera localidad del Buffer Auxiliar.
	32520			; A subrutina de Encerado.
50206	240	NOP		
50210	012703	MOV	#3, R3	; Suma BCD del contenido del área de memoria T1 del Buffer Principal (BP) con el área de memoria T1 (i). El resultado se almacena en la última área: T1(BP) + T1(i) — T1(i).
	3			
50214	012704	MOV	#10, R4	; En R3 se carga número (3) de palabras de 2 bytes del 1er. sumando (T1 de Buffer Principal).
	10			
50220	012701	MOV	#2400, R1	; En R4 se carga número máximo (10) de palabras de 2 bytes del 2do. sumando (T1(i)).
	2400			
50224	013700	MOV	@#40000, R0	; En R1 se carga dirección (2400) de la primera localidad del área de memoria T1 Buffer Principal.
	40000			
50230	4737	JSR	PC, SBCD	; En R0 se carga dirección (40000) de la localidad en la que está almacenada la dirección base del área de memoria T1(i).
	33020			; A subrutina de suma en BCD
50234	240	NOP		
50236	240	NOP		
50240	240	NOP		
50242	012703	MOV	#3, R3	; Suma BCD del contenido del área de memoria T2 del Buffer Principal con el área de memoria T2(i). El resultado se almacena en la última área: T2(BP) + T2(i) — T2(i).
	3			
50246	012704	MOV	#10, R4	; En R3 se carga número (3) de palabras de 2 bytes del 1er. sumando (T2) de Buffer Principal.
	10			



50252	012701 2406	MOV #2406,R1	; En R1 se carga dirección (2406) de la primera lo- ; calidad del área de memoria T2 de Buffer Princi- ; pal.
50256	013700 40002	MOV @#40002,R0	; En R0 se carga dirección (40002) de la localidad ; en la que está almacenada la dirección base del ; área de memoria T2(i).
50262	4737 33020	JSR PC,SBCD	; A subrutina de suma en BCD.
50266	240	NOP	
50270	240	NOP	
50272	240	NOP	
50274	240	NOP	
50276	012703 5	MOV #5,R3	; Suma BCD: TT(BP) + TT(i) TT(i).
50302	012704 12	MOV #12,R4	; En R3 se carga con número (5) de palabras de 2 ; bytes del ler. sumando (TT de B.P.).
50306	012701 2414	MOV #2414,R1	; En R4 se carga con número (12) máximo de palabras ; de 2 bytes del 2do. sumando (TT(i)).
50312	013700 40004	MOV @#40004,R0	; En R1 se carga dirección (2414) de la primera lo- ; calidad del área de memoria TT de B.P.
50316	4737 33020	JSR PC,SBCD	; En R0 se carga dirección (40004) de la localidad ; en la que está almacena la dirección base del ; área de memoria TT(i).
50322	240	NOP	; A subrutina de suma en BCD.
50324	240	NOP	
50326	240	NOP	
50330	012704 11	MOV #11,R4	; Suma BCD: T1(i) + Buffer Auxiliar -- Buffer auxi- ; liar.
50334	012703 10	MOV @#10,R3	; En R4 se carga número de palabras (11) de 2 bytes ; de Buffer Auxiliar (2do. sumando). ; En R3 se carga número de palabras (10) de 2 bytes ; del área de memoria T1(i) (ler. sumando).

50340	013701 40000	MOV @#40000,R1	; En R1 se carga dirección (40000) de la localidad ; en la que está almacenada la dirección base del ; área de memoria T1(i).
50344	012700 40150	MOV #40150,R0	; En R0 se carga dirección (40150) de la primera lo- ; calidad del Buffer Auxiliar.
50350	4737 33020	JSR PC,SBCD	; A subrutina de Suma en BCD.
50354	240	NOP	
50356	240	NOP	
50360	240	NOP	
50362	012703 10	MOV #10,R3	; Suma BCD T2(i) + Buffer Auxiliar → Buffer Auxili- ; liar.
50366	012704 11	MOV #11,R4	; En R3 se carga número de palabras (10) de 2 bytes ; del área de memoria T2(i) (ler. sumando).
50372	013701 40002	MOV @#40002,R1	; En R4 se carga número de palabras (11) de 2 bytes ; del Buffer Auxiliar (2do. sumando).
50376	012700 40150	MOV #40150,R0	; En R1 se carga dirección (40002) de la localidad ; en la que está almacenada la dirección base del ; área de memoria T2(i).
50402	4737 33020	JSR PC,SBCD	; En R0 se carga dirección (40150) de la lra. loca- ; lidad del Buffer Auxiliar.
50406	240	NOP	; A subrutina de suma en BCD.
50410	240	NOP	
50412	240	NOP	
50414	012703 12	MOV #12,R3	; Suma BCD:TT(i) + Buffer Auxiliar → Buffer Auxili- ; liar.
50420	012704 13	MOV #13,R4	; R3 se carga con número (12) de palabras de 2 ; bytes del ler. sumando (TT(i)).
50424	013701 40004	MOV @#40004,R1	; R4 se carga con número máximo (13) de palabras ; de 2 bytes del 2do. sumando (Buffer Auxiliar).
			; En R1 se carga dirección (40004) de la localidad ; en la que está almacenada la dirección base del ; área de memoria TT(i).

50430	012700	MOV	#40150, R0	; En R0 se carga dirección (40150) de la lra.locali
	40150			; dad del Buffer Auxiliar.
50434	4737	JSR	PC, SBCD	; A subrutina de suma en BCD.
	33020			
50440	012701	MOV	#40150, R1	; Transferencia del contenido del Buffer Auxiliar
	40150			; al área de memoria Tj(1).
50444	013703	MOV	@#40006, R3	; R1 se carga con dirección (40150) de la primera
	40006			; localidad del Buffer Auxiliar.
50450	012704	MOV	#13, R4	; En R3 se carga dirección (40006) de la localidad
	13			; en la que está almacenada la dirección base del
50454	4737	JSR	PC, TRANS	; área de memoria Tj(i).
	32474			; R4 se carga con el número (13) de palabras de 2
50460	240	NOP		; bytes a ser transferidas.
50462	240	NOP		; A subrutina de transferencia.
50464	013700	MOV	@#40010, R0	
	40010			
50470	013701	MOV	@#40012, R1	; R0 se carga con dirección (40010) de la localidad
	40012			; en la que está almacenada la dirección de la loca
50474	013704	MOV	@#40014, R4	; lidad destinada a acumular NLL.
	40014			; R1 se carga con dirección (40012) de la localidad
50500	012703	MOV	#2426, R3	; en la que se encuentra la dirección de la locali-
	2426			; dad destinada a acumular NLL.
				; En R4 se carga con dirección (40014) en la que es
				; tá almacenada la dirección de la localidad desti-
				; nada a acumular NLLI.
				; R3 se carga con dirección (2426) en la que se
				; encuentra la bandera "tipo de llamada" (btll).

50504	4737	JSR	PC,ALLA	; A subrutina de acumulación de llamadas.
	33120			
50510	240	NOP		
50512	240	NOP		
50514	240	NOP		
<p>; Suma binaria: Dependiendo de que bandera del código Razón de Desconexión esté en set (1), se incrementa en 1 la localidad correspondiente a acumular los datos del código razón de desconexión tan de las tablas de totales y de troncal.</p> <p>; Akk(i) — Akk(i) + 1 kk = Código razón de desconexión.</p> <p>    i = 0 para tablas totales</p> <p>    i = 1... n = para tablas TR0(i).</p>				
50516	012703	MOV	@#40016,R3	; En R3 se carga con dirección (40016) de la localidad en la que está almacenada la dirección base del área de memoria CRD.
	40016			
50522	012701	MOV	#2431,R1	; En R1 se carga con dirección (2431) de la localidad destinada a la bandera que indica si existen en el campo 11 los códigos de la tabla CRD.
	2431			
50526	4737	JSR	PC,ACDR	; A subrutina que acumula los Códigos Razón de desconexión
	33150			
50532	240	NOP		
50534	240	NOP		
50536	240	NOP		
<p>; Se suman los valores almacenados en las áreas de memoria TSUTE y TSUTI contenidos en las tablas de troncal TR0(I), con los correspondientes del Buffer Principal.</p>				
50540	4737	JSR	PC,ATSUT	; A subrutina de acumulación de tiempos.
	33370			
50544	240	NOP		
50546	207	RTS	PC	; Retorno.

; SUBROUTINA: ACUMULACION DE LOS TIEMPOS DE ESTABLECIMIENTO DE LAS LLAMADAS EFEE  
 ; TIVAS E INEFECTIVAS.

; Se utiliza Buffer Auxiliar, localidades 40150 a 40174.

; Argumentos:

; Entrada : En localidad 40020/1 se carga con la dirección de la lra. localid-  
 ; dad del área de memoria TSUTE.

; En localidad 40022/3 se carga con la dirección de la lra. localid-  
 ; dad del área de memoria TSUTI.

; Localidad 2426, bandera tipo de llamad.

; Salida : De acuerdo a dirección contenida en las anteriores localidades se  
 ; suman los valo almacenados en las áreas TSUTE y TSUTI, conteni-  
 ; dos en la tabla de troncal TR0 (I) en proceso, con los correspon-  
 ; dientes del Buffer Principal. Si localidad 2426 es l(llamada efec-  
 ; tiva) TSUT de Buffer se transfriere a TSUTE de tabla TR0(I), si es  
 ; 0 a TSUTI de tabla TR0(I).

33370 012700 MOV #13,R0 ; Encerado de Buffer Auxiliar.  
 ; En R0 se carga número (13) de palabras de 2 bytes  
 ; a ser enceradas.  
 33374 012701 MOV #40150,R1 ; En R1 se carga dirección (40150) de la primera lo-  
 ; calidad del Buffer.  
 33400 4737 JSR PC,ENC ; A subrutina de encerado.

33404 240 NOP  
 33406 240 NOP  
 33410 240 NOP  
 33412 240 NOP

33414 012701 MOV #2400,R1 ; Transferencia del contenido del área de memoria  
 ; TL de Buffer Principal al Buffer Auxiliar  
 ; TL(BP) ——— BA.  
 ; En R1 se carga dirección (2400) de la lra. localid-  
 ; dad del área TL.

33420	012703	MOV	#40150,R3	; En R3 se carga dirección (40150) de la lra. localidad del buffer auxiliar.
	40150			
33424	012704	MOV	#3,R4	; En R4, se carga número (3) de palabras de 2 bytes a ser transferidas.
	3			
33430	4737	JSR	PC,TRANS	; A subrutina de transferencia.
	32474			
33434	240	NOP		
33436	240	NOP		
33440	240	NOP		
33442	012703	MOV	#3,R3	; Suma BCD: T2 (Buffer Principal) + Buffer Auxiliar
	3			; — Buffer Auxiliar.
33446	012704	MOV	#7,R4	; Se obtiene el tiempo de establecimiento (TSUT) almacenado en Buffer Auxiliar.
	7			; En R3 se carga número de palabras (3) de 2 bytes del área de memoria T2.
33452	012701	MOV	#2406,R1	; En R4 se carga número (7) de palabras de 2 bytes del Buffer Auxiliar.
	2406			; En R1 se carga dirección (2406) de la lra. localidad del área de memoria T2.
33456	012700	MOV	#40150,R0	; En R0 se carga dirección (40150) de la lra. localidad del buffer auxiliar.
	40150			; A subrutina de suma BCD.
33462	4737	JSR	PC,SBCD	
	33020			
33466	240	NOP		
33470	240	NOP		
33472	240	NOP		
33474	022737	CMP	#0,@#2426	; ¿Es llamada efectiva? btl1 = 1 — llamada efectiva
	0			; 0 — llamada inefectiva
	2426			
33502	1015	BNE	NO	; Es contenido de localidad 2426 (bandera tipo llamada /btl1) es cero?
				; No bifurca a NO (Es llamada efectiva),
				; Si continúa:
				; Se almacena tiempo de establecimiento de llamada
				; inefectiva en área TSUTI(i).

; Suma BCD: Buffer Auxiliar + TSUTI(i) — TSUTI(i)  
 ; En R3, se carga número de palabras (7) de 2 bytes  
 ; del Buffer Auxiliar.  
 ; En R4 se carga número de palabras (10) de 2 bytes  
 ; del área de memoria TSUTI.  
 ; En R1 se carga dirección (40150) de la lra. local-  
 ; lidad del Buffer.  
 ; En R0 se carga dirección (40022) de la localidad  
 ; en la que está almacenada la dirección base del  
 ; área de memoria TSUTI.  
 ; A subrutina de suma BCD.

; Salto a SALIDA.  
 ; Se almacena tiempo de establecimiento de llamada  
 ; efectiva en el área de memoria TSUTE.

; Suma BCD:  
 ; Buffer Auxiliar + TSUTE(i) — TSUTE(i).  
 ; En R3 se carga el número de palabras (7) de 2 by-  
 ; tes del Buffer.  
 ; En R4 se carga número de palabras (10) de 2 bytes  
 ; del área de memoria TSUTE.  
 ; En R1 se carga dirección (40150) de la lra.locali-  
 ; dad del Buffer.  
 ; En R0 se carga dirección (40020) de la localidad  
 ; en la que está almacenada la dirección base del área  
 ; de memoria TSUTE.  
 ; A subrutina de suma BCD.

; Retorno.

33504	012703	MOV	#7,R3			
33510	012704	MOV	#10,R4			
33514	012701	MOV	#40150,R1			
33520	013700	MOV	@#40022,R0			
33524	4737	JSR	PC,SBCD			
	33020					
33530	240	NOP				
33532	240	NOP				
33534	412	BR	SALIDA			
33536	012703	MOV	#7,R3	NO		
33542	012704	MOV	#10,R4			
33546	012701	MOV	#40150,R1			
33552	013700	MOV	@#40020,R0			
33556	4737	JSR	PC,SBCD			
	33020					
33562	240	NOP	SALIDA			
33564	240	NOP				
33566	207	RTS	PC			

;SUBROUTINA:HORA CARGADA

```

; Notación:
; BA, Buffer Auxiliar localidades 40036 a 40050.
; A(K), B(K), HHT(K), HHFE(K), donde K = 1,2,..N, número de troncales almacena
; das en la tabla TR0. Las localidades de las primeras áreas (K = 1) son las
; siguientes (cada área formada por un número igual de localidades):
; A(1) 43000 a 43024; B(1) 44000 a 44024; HHT(1) 45000 a 45024 y HHFE(1)
; 45026 a 45036.
; A,B son Buffer Auxiliares, HHT contiene el volumen de tráfico pico y HHFE
; la hora a que se produce dicho tráfico.
; Argumentos:
; Entrada : En localidad 20774 se carga dirección de retorno.
;           El área de memoria Tj(K) contiene en código BCD el tiempo total
;           empleado en las comunicaciones y el área FCHC la fecha.
; Salida  : En las K áreas de memoria HHT y HHFE se almacena el volumen de trá
;           fico pico y la fecha a la que se produce dicho tráfico.

```

```

34136 012700 MOV #40050,R0
      40050
34142 010340 MOV R3,-(R0)
34144 010140 MOV R1,-(R0)
34146 010240 MOV R2,-(R0)
34150 010440 MOV R4,-(R0)
34152 010540 MOV R5,-(R0)
; Almacenamiento en memoria (localidades 40050-
; 40036 de los punteros que direccionan los campos
; utilizados en el Proceso).
; En registro R0 se carga dirección (40050) de la
; lra. localidad del área de memoria.
; El contenido del registro R3 se almacena en loca-
; lidades 40046/7.
; El contenido del registro R1 se almacena en loca-
; lidades 40044/5.
; El contenido del registro R2 se almacena en loca-
; lidades 40042/3.
; El contenido del registro R4 se almacena en loca-
; lidades 40040/1.
; El contenido del registro R5 se almacena en loca-
; lidades 40036/7.

```



34154	240	NOP					
34156	240	NOP					
34160	022737	CMP	#0, @#2302				; Bandera HC es cero? ; Contenido de la localidad 2302 es cero?
	0						
	2302						
34166	1055	BNE	2H				; No bifurca a 2H. ; Si continúa: ; Proceso para primera hora. ; Se transfiera al contenido de las N áreas de memo- ; ria Tj(K) de las tablas de troncal, a las N áreas ; de memoria HHT(K). K = 1...N ; En cada área Tj se encuentra el tiempo total em- ; pleado en la comunicación: ; Tj(K) — HHT(K). ; R5 se carga el número de troncales N almacenados ; en tabla TRO. ; En R2 se carga dirección base (45000 de las ta- ; blas en las que se almacena el volumen de tráfico ; (HHT(1)). ; En localidad 7774 se carga dirección retorno ; (34220) de la próxima subrutina.
34170	013705	MOV	@#1362, R5				
	1362						
34174	012702	MOV	#45000, R2				
	45000						
34200	012737	MOV	#34220, @#7774				
	34220						
	7774						
34206	012737	MOV	#40, @#7770				
	40						
	7770						
34214	4737	JSR	PC, TRTA				
	35000						
34220	240	NOP					
34222	013705	MOV	@#1362, R5				
	1362						

; El contenido de las N áreas de memoria Tj(K) se  
; transfiera a las N áreas de memoria A(K). K = 1..N  
; Tj(K) — A(K)  
; R5 se carga con dirección (1362) de la localidad  
; en la que está almacenado el número de troncales  
; "N".

34226	012702	MOV	#43000, R2	; R2 se carga con dirección (43000) de la primera
	43000			; localidad del área de memoria A(1).
34232	012737	MOV	#34252, @#7774	; En localidad 7774 se carga dirección de retorno
	34252			; (34252) de la próxima subrutina.
	7774			
34240	012737	MOV	#26, @#7770	; En localidad 7770 se carga número de palabras de
	26			; 2 bytes que conforman cada área de memoria A(K).
	7770			
34246	4737	JSR	PC, TRTA	; A subrutina de transferencia de áreas.
34252	35000	NOP		
	240			
34254	013705	MOV	@#1362, R5	; El contenido del área de memoria FCHC (Incrementa-
	1362			; dor de fecha) se almacena en los N áreas de memo-
				; ria HHFE (K).
				; FCHC — HHFE (K)
				; R5 se carga con dirección (1362) de la localidad
				; en la que está almacenada el número de troncales
				; "N".
34260	012703	MOV	#45000, R3	; R3 se carga con dirección (45000) de la lra loca-
	45000			; lidad del área de memoria HHFE(i).
34264	4737	JSR	PC, TRFEC	; A subrutina transferencia de fecha.
	34054			
34270	240	NOP		
34272	077504	SOB	R5, A	; Se decrementa en 1 R5. Si el valor contenido en
				; este registro es diferente de cero, bifurca a A
				; si es igual a cero continúa:
34274	240	NOP		
34276	240	NOP		
34300	240	NOP		
34302	137	JMP	PC, 3H	; Salto a 3H.
	34550			
34306	240	NOP		
34310	240	NOP		
34312	240	NOP		



34370	013737	MOV	@#1362, @#2306	; al área HHFE(K). Donde K = 1...N. El proceso se repite "N" veces.
	1362			
	2306			
34376	012700	MOV	#44000, R0	; El contenido de la localidad de memoria 1362 en la que está almacenada el número de troncales N se transfiere a la localidad 2306.
	44000			
34402	012702	MOV	#45000, R2	; En R0 se carga dirección (44000) de la primera localidad del área B(1) (K = 44000).
	45000			
34406	012705	MOV	#45000, R5	; En R2 se carga dirección (45000) de la 1ra.localidad del área HHCC(1)
	45000			; El contenido de R2 se transfiere a R5.
34412	010001	MOV	R0, R1	; El contenido de R0 se transfiere a R1.
34414	010203	MOV	R2, R3	; El contenido de R2 se transfiere a R3.
34416	4737	JSR	PC, CPFEC	; A subrutina de Comparación de fecha.
	34000			
34422	240	NOP		
34424	240	NOP		
34426	062700	ADD	#26, R0	; Se obtiene siguiente área B(K).
	26			
34432	062702	ADD	#40, R2	; Se obtiene siguiente área HHCC(M).
	40			
34436	062705	ADD	#40, R5	; Se carga R5 con el valor 40.
	40			
34442	240	NOP		
34444	240	NOP		
34446	5337	DEC	@#2306	; ¿ Es última troncal ? (N=0)
	2306			; N = N - 1.
				; Se decrementa en 1 el contenido de la localidad 2306.
34452	1357	BNE	LAZO	; ¿ Es N = 0 ?
34454	240	NOP		; No, regresa a LAZO.
34456	240	NOP		; Si continúa:
34460	240	NOP		
34462	240	NOP		
34464	240	NOP		
34466	240	NOP		
34470	240	NOP		

34472	240	NOP			
34474	240	NOP			
34476	240	NOP			
34500	240	NOP			
34502	240	NOP			
34504	240	NOP			
34506	240	NOP			
34510	240	NOP			
34512	012737	MOV	#26, @#7770		
	26				
	7770				
34520	013705	MOV	@#1362, R5		
	1362				
34524	012702	MOV	#43000, R2		
	43000				
34530	012737	MOV	#34542, @#7774		
	34542				
	7774				
34536	4737	JSR	PC, TRTA		
	35000				
34542	240	NOP			
34544	240	NOP			
34546	240	NOP			
34550	240	NOP			3H
34552	012737	MOV	#34570, @#7006		
	34570				
	7006				
34560	012706	MOV	#10000, SP		
	10000				

; El contenido de las N áreas de memoria Tj(K) se  
 ; transfiere a las N áreas de memoria A(K). Donde  
 ; Tj(K) — A(K).  
 ; En localidad 7770 se carga número (26) de pala-  
 ; bras de 2 bytes que conforman cada área A(K).  
 ; En R5 se carga dirección (1362) de la localidad  
 ; en que está almacenada el número de troncales N.  
 ; En R2 se carga dirección (43000) de la lra.locali-  
 ; dad del área de memoria A(1).  
 ; En localidad 7774 se carga dirección de retorno  
 ; 34542.  
 ; A subrutina de transferencia de áreas.  
 ; Los datos contenidos en el área de memoria FCHC  
 ; y que reperesentan una fecha se incrementan en  
 ; 1 hora: FCHC — FCHC + 1.  
 ; En localidad 7006 se carga dirección de retorno  
 ; 34570.  
 ; Se inicializa el puntero de pila en 10000.

```

34564 4737 JSR PC,INCFEC ; A subrutina de incremento en 1 hora a la fecha
35200 35200 ; contenida en FCHC.
34570 240 NOP
34572 240 NOP
34574 012737 MOV #1,@#2302 ; Bandera HC en 1.
1 ; Localidad 2302 se carga con valor 1.
2302

34602 4737 JSR PC,POP ; Restitución de los punteros
35100 35100 ; A subrutina que restituye al programa principal,
34606 240 NOP ; los punteros de los diferentes campos.
34610 012706 MOV #20774,SP ; Puntero de pila se carga con dirección(20774) de
20774 ; localidad en que esta dirección de retorno.
207 ; Retorno.
*****
*****

```

; SUBROUTINA CARGA DE DIRECCIONES

```

; Argumentos:
; Entrada : Se carga el puntero de pila con el valor 10000.
; Salida : El área de memoria "Direcciones" se almacena los siguientes valo-
; res:
; Localidades 40000/1 con 100500
; Localidades 40002/3 con 100400
; Localidades 40004/5 con 100420
; Localidades 40006/7 con 100444
; Localidades 40010/1 con 100472
; Localidades 40012/3 con 100474
; Localidades 40014/5 con 100476
; Localidades 40016/7 con 100520
; Localidades 40020/1 con 100540
; Localidades 40022/3 con 44766
; Localidades 40024/5 con 100500
; Localidades 40026/7 con 100474
; Localidades 40030/1 con 100420
; Localidades 40032/3 con 100472
; Localidades 40034/5 con 100476
; Localidades 40036/7 con 100520
; Localidades 40040/1 con 100540
;

```

13000	012704 40000	TDRI	MOV	#40000,R4	;	R4 se carga con la dirección de la lra. localidad del área de memoria "Direcciones"
					;	Localidades 40000 a 40022 se cargan con las direcciones necesarias en el cálculo de los datos de la columna (a) del CCSTI.
13004	012724		MOV	#100500,(R4)+	;	En localidades 40000/1 se almacena dirección
13010	012724		MOV	#100400,(R4)+	;	100500.(lra. localidad del área de memoria TSUTE)
13014	012724		MOV	#100420,(R4)+	;	En localidades 40002/3 se almacena dirección
13020	012724		MOV	#100444,(R4)+	;	100400. (lra. localidad del área T2).
13024	012724		MOV	#100472,(R4)+	;	En localidades 40004/5 se almacena dirección
13030	012724		MOV	#100474,(R4)+	;	100420. (lra. localidad del área TT).
13034	012724		MOV	#100476,(R4)+	;	En localidades 40006/7 se almacena dirección
13040	012724		MOV	#100520,(R4)+	;	100444. (lra. localidad del área Tj).
13044	012724		MOV	#100540,(R4)+	;	En localidades 40010/1 se almacena dirección
13050	012724 44766		MOV	#44766,(R4)+	;	100472. (localidad de NLLT).
					;	En localidades 40012/3 se almacena dirección
					;	100474. (localidad de NLLI).
					;	En localidades 40014/5 se almacena dirección
					;	100476. (localidad de NLLI).
					;	En localidades 40016/7 se almacena dirección
					;	100520. (lra. localidad del área TSUTI)
					;	En localidades 40020/1 se almacena dirección
					;	100540. (lra. localidad del área CRD).
					;	En localidades 40022/3 se almacena dirección
					;	44766. (lra. localidad del área HHCC).
					;	Localidades 40024 a 40040 se cargan con direcciones requeridas para los cálculos de los datos de la columna (b) del cuadro CCSTI.
13054	012724		MOV	#100500,(R4)+	;	En localidades 40024/5 se almacena dirección
13060	012724		MOV	#100474,(R4)+	;	100500. (lra. localidad del área TSUTE).
13064	012724 100420		MOV	#100420,(R4)+	;	En localidades 40026/7 se almacena dirección
					;	100474. (localidad NLLI).
					;	En localidades 40030/1 se almacena dirección
					;	100420. (lra.localidad del área TT).

```

13070 012724 MOV #100472,(R4)+ ; En localidades 40032/3 se almacena dirección
100472 ; 100472. (localidad de NLLT).
13074 012724 MOV #100476,(R4)+ ; En localidades 40034/5 se almacena dirección
100476 ; 100476. (localidad de NLLI)
13100 012724 MOV #100520,(R4)+ ; En localidades 40036/7 se almacena dirección
100520 ; 100520. (lra. localidad del área TSUTI).
13104 012724 MOV #100540,(R4)+ ; En localidades 40040/1 se almacena dirección
100540 ; 100540. (lra. localidad del área CRD)
13110 240 NOP
13112 240 NOP
13114 240 NOP
13116 240 RST PC ; Retorno.

```

\*\*\*\*\*  
\*\*\*\*\*

**; SUBROUTINA: ITERATIVO/IMPRESION**

```

; Argumentos:
; Entrada: Localidad 20774 se carga con la dirección de retor-
; no.
; Salida: El contenido del registro R0 indica el número (I) de
; la troncal digitada.
; En la Unidad de I/O se imprime frase: "Impresión de re
; sultados de TRO". Entonces el operador debe pulsar los
; dígitos (3) de la troncal que desea editar; si estos
; dígitos no corresponden a una de las troncales almace-
; nadas en la tabla TRO se repite la impresión de la fra
; se; caso contrario retorna al programa principal.
; A subrutina retorno de carro, avance de línea.
14600 4737 ITEIMP JSR PC,CRLF
30030
14604 4737 N JSR PC,CRLF
30030
; Impresión de frase: "Impresión de resultados de
; TRO:".
; En R0 se almacena número (40) de caracteres que
; constituyen la frase a imprimirse.
14610 012700 MOV #40,R0
40

```



14614	012701	MOV	#36070,R1	; En R1 se almacena dirección (36070) de la locali-
	36070			; dad en que se encuentra ler. carácter.
14620	4737	JSR	PC,IMP	; A subrutina de impresión.
	30000			
14624	240	NOP		
14626	240	NOP		
14630	012702	MOV	#1402,R2	; Consulta
	1402			; R2 se carga con la dirección (1402) de la locali-
14634	4737	JSR	PC,CONSUT	; dad desde la cual se almacenan los caracteres di-
	30070			; gitados por el operador desde I/O.
14640	4737	JSR	PC,CRLF	; A subrutina consulta.
	30030			; A subrutina de retorno de carro, avance de línea.
14644	012700	MOV	#1402,R0	; Conversión de ASCII a EBCDIC de los anteriores ca
	1402			; racteres ingresados.
14650	012703	MOV	#1405,R3	; En R0 y en R3 se cargan las direcciones (1402 y
	1405			; 1405) de localidades en que están almacenados res
14654	4737	JSR	PC,CONAE	; pectivamente,el primero y último caracteres .
	31054			; A subrutina de conversión de ASCII a EBCDIC
14660	240	NOP		
14662	240	NOP		
14664	240	NOP		
14666	240	NOP		
14670	012706	MOV	#10000,SP	; Se verifica que los 3 caracteres digitados corres
	10000			; pondan a 3 caracteres almacenados en Tabla TR0.
14674	012737	MOV	#14604,@#7774	; En el caso de que el conjunto de 3 caracteres di-
	14604			; gitados sea igual a uno de los conjuntos de 3 ca-
	7774			; racteres de la tabla TR0, el programa bifurca a
				; la dirección 14716; caso contrario a 14604(N).
				; Inicialización de puntero de pila en 10000.
14702	012702	MOV	#14604,R2	; En localidad 7774 se almacena dirección "14604(N)
	1402			; ; En R2 se carga dirección (1402) en que está el
				; ler. dígito (del ler conjunto de 3) a ser com -
				; ; parados.

```

14706 012703 MOV #524,R3 ; En R3 se carga dirección 524 en que está primer
524 ; dígito (del conjunto de 3) de la tabla TR0.
14712 4737 JSR PC,VERIF ; A subrutina de verificación
36150
14716 240 NOP
14720 012706 MOV #20774,SP ; Se carga en puntero de pila la dirección de la lo
20774 ; calidad (20774) en la cual está almacenada direc-
; ción de retorno.

14724 240 NOP
14726 240 NOP
14730 207 RTS PC ; Retorno.

```

```

*****
*****

```

```

; SUBROUTINA: LINEA 3.PERIODO DE OBSERVACION (FECHAS INICIAL Y FINAL
; INGRESADAS A TRAVES DE I/O).

```

```

; Argumentos:
; Entrada : En localidad 20774 almacenar dirección de retorno.
; Salida : Se imprime en la unidad de I/O frase: "Período de ob-
; servación del:".
; Se calcula e imprimen los valores correspondientes al
; período de observación del cuadro CCSTI: fecha ini-
; cial y final en GMT.
; En R0 se almacena cantidad (34) de caracteres
; que forman la frase a imprimirse.
; En R1 dirección (24620) de la localidad en que
; está ler. carácter de frase.
; A subrutina de impresión.

21000 012700 LIN3 MOV #34,R0
34
21004 012701 MOV #24620,R1
24620
21010 4737 JSR PC,IMP
30000
21014 240 NOP
21016 240 NOP
21020 240 NOP
21022 240 NOP

```

```

21024 012737      MOV #21042,@#20776
      21042      ; Cálculo de la Hora Internacional (GMT) de la fe-
      20776      ; cha inicial ingresada y su almacenamiento en el
      ; área de memoria HGMT.
21032 012701      MOV #500,R1
      500       ; La dirección (21042) a la cual se retorna luego
      4737      ; de efectuarse la próxima subrutina, se almacena
      24770     ; en localidad 20776.
      240      ; En R1 se carga dirección (500) de la lra.locali-
21042 240        NOP
      240      ; dad del área FII (Fecha Inicial Ingresada).
21044 240        NOP
      240      ; A subrutina que calcula Hora Internacional
21046 240        NOP
21050 240        NOP
21052 240        NOP

21054 012700      MOV #34,R0
      34       ; Impresión de los caracteres contenidos en área
21060 012701      MOV #24712,R1
      24712    ; HGMT que constituyen la fecha inicial en GMT.
21064 4737       JSR PC,IMP
      30000   ; En R0 se almacena cantidad (34) de caracteres a
      240     ; imprimirse.
      240     ; En R1 la dirección (24712) de la localidad en que
21070 240        NOP
21072 240        NOP

21074 012737      MOV #21164,@#20776
      21164    ; Cálculo de al Hora Internacional (GMT) de la fe-
      20776    ; cha inicial ingresada y su almacenamiento
      ; en el área de memoria HGMT.
21102 012701      MOV #512,R1
      512     ; La dirección (21164) a la cual se retorna luego
21106 4737       JSR PC,GMT
      24770   ; de efectuarse la subrutina se almacena en locali-
      ; dad 20776.
      ; En R1 se carga dirección (512) de la lra.locali-
      ; dad del área FFI (Fecha Final Ingresada).
      ; A subrutina que calcula Hora Internacional

```



21200	013701 40022	MOV @#40022,R1	; Transferencia del contenido del área HHCC a área ; HHPP.
21204	012703 25120	MOV #25120,R3	; En R1 se carga dirección base (40022) localidad ; en la que está almacenada la dirección 25120 (lra ; localidad del área de memoria HHCC de la troncal ; en edición).
21210	012704 5	MOV #5, R4	; En R3 se carga dirección de la la. localidad del ; área HHPP.
21214	4737 32474	JSR PC, TRANSF	; En R4 se carga cantidad (5) de palabras a ser ; transferidas (cada palabra formado por 2 bytes) ; A subrutina de transferencia.
21220	240	NOP	
21222	240	NOP	
21224	240	NOP	
21226	4737 25150	JSR PC, DECLH	; Obtención de Hora Inicial. ; A subrutina que resta 1 hora a la fecha contenida ; en HHPP.
21232	240	NOP	
21234	240	NOP	
21236	240	NOP	
21240	012737 21256 20776	MOV #21256,@#20776	; Cálculo de la Hora Internacional (GMT) de la Hora ; Inicial contenida en área HHPP y su almacenamien- ; to en el área HGMT. ; En localidad 20776 se almacena dirección de ; retorno 21256.
21246	012701 25120	MOV #25120,R1	; En R1 se carga dirección (25120) de la lra locali- ; dad del área HHPP.
21252	4737 24770	JSR PC, GMT	; A subrutina que calcula Hora Internacional
21256	240	NOP	
21260	240	NOP	
21262	240	NOP	
21264	012700 24	MOV #24, R0	; Impresión de 24 caracteres "blancos". ; En R0 se carga cantidad (24) de caracteres a ; imprimirse.

21270	012701	MOV	#36750,R1	; En R1 se carga dirección (36750) de la localidad
	36750			; en que está ler. carácter.
21274	4737	JSR	PC,IMP	; A subrutina de Impresión.
21300	30000	NOP		
	240			
21302	012700	MOV	#15,R0	; Impresión de 15 caracteres blancos.
	15			; En R0 se carga cantidad (15) de caracteres a
21306	012701	MOV	#36750,R1	; imprimirse.
	36750			; En R1 dirección (36750) de la localidad en que es
21312	4737	JSR	PC,IMP	; tá ler. carácter.
	30000			; A subrutina de Impresión.
21316	240	NOP		
21320	240	NOP		
21322	012700	MOV	#13,R0	; Impresión de los caracteres contenidos en el área
	13			; HGMT que representan la Hora Inicial en GMT.
21326	012701	MOV	#24732,R1	; En R0 se carga cantidad (13) de caracteres a
	24732			; imprimirse.
21332	4737	JSR	PC,IMP	; En R1 dirección (24732) de la localidad en que es
	30000			; tá ler. carácter.
21336	240	NOP		; A subrutina de Impresión.
21340	240	NOP		
21342	240	NOP		
21344	012737	MOV	#21362,@#20776	; Cálculo de la Hora Internacional (GMT) de la Hora
	21362			; Final y su almacenamiento en el área HGMT.
	20776			; La dirección (21362) a la cual se retorna luego
21352	013701	MOV	@#40022,R1	; de efectuarse la próxima subrutina se almacena en
	40022			; localidad 20776.
				; En R1 se carga dirección (40022) de la localidad
21356	4737	JSR	PC,GMT	; en la que está almacenada la dirección de la lra.
	24770			; localidad del área de memoria HHCC.
				; A subrutina que calcula la Hora Internacional.

21362	240	NOP			
21364	240	NOP			
21366	012700	MOV	#21,RO		
	21				
21372	012701	MOV	#36750,R1		
	36750				
21376	4737	JSR	PC,IMP		
	30000				
21402	240	NOP			
21404	012700	MOV	#13,RO		
	13				
21410	012701	MOV	#24732,R1		
	24732				
21414	4737	JSR	PC,IMP		
	30000				
21420	4737	JSR	PC,CRLF		
	30030				
21424	012706	MOV	#20774,SP		
	20774				
21430	207	RTS	PC		

```

; Impresión de 21 caracteres blancos.
; En R0 se carga cantidad (21) de caracteres a
; ser impresos.
; En R1 dirección (36750) de la localidad del ler.
; carácter.
; A subrutina de Impresión.

; Impresión de los caracteres que representan la
; Hora Final en GMT.
; En R0 se carga cantidad (13) de caracteres a
; ser impresos.
; En R1 dirección (24732) de la localidad del ler.
; carácter.
; A subrutina de Impresión.

; A subrutina retorno de carro, avance de línea.

; En puntero de pila se almacena la dirección
; (20774) de la localidad en la cual está la direc-
; ción de retorno.
; Retorno.

```

\*\*\*\*\*

; SUBROUTINA: LINEA 5.- NUMERO DE LLAMADAS

```

; Argumentos:
; Entrada : En localidad 20774 almacenar dirección de retorno.
; Salida  : Se imprime frase "Número total de llamadas".
;          Se calcula e imprime el valor de dicho número del cuadro CCSTI, de
;          la troncal en edición.

```

21432	012706	LINS	MOV	#10000,SP	; Se inicializa puntero de pila en 10000.
	10000				
					; Impresión de frase: Número total de llamadas.
21436	012700		MOV	#22,R0	; Impresión de frase: "Número total".
	22				; En R0 se carga cantidad (22) de caracteres a
21442	012701		MOV	#25260,R1	; imprimirse.
	25260				; En R1 dirección (25260) de la localidad del ler.
21446	4737		JSR	PC,IMP	; carácter.
	30000				; A subrutina de Impresión.
21452	240		NOP		
21454	240		NOP		
					; Impresión de frase "Llamadas".
21456	012700		MOV	#12,R0	; En R0 se carga cantidad (12) de caracteres a
	12				; imprimirse.
21462	012701		MOV	#24046,R1	; En R1 dirección (24046) de la localidad del ler.
	24046				; carácter.
21466	4737		JSR	PC,IMP	; A subrutina de Impresión.
	30000				
21472	240		NOP		
					; Impresión de 24 caracteres "blancos".
21474	012700		MOV	#24,R0	; En R0 se carga cantidad (24) de caracteres a
	24				; imprimirse.
21500	012701		MOV	#36750,R1	; En R1 dirección (36750) de la localidad del ler.
	36750				; carácter.
21504	4737		JSR	PC,IMP	; A subrutina de Impresión.
	30000				
21510	240		NOP		
					; Conversión de Binario a ASCII del número de Llama
					; das de la troncal en edición.
21512	012705		MOV	#40010,R5	; En R5 se carga dirección (40010) de la localidad
	40010				; en la que está almacenada la dirección de la loca
					; lidad en la que se encuentra el valor binario co-
					; rrespondiente al Nº de Llamadas.





```

21602 4737 JSR PC,IMP ; A subrutina de impresión.
      30000
21606 240 NOP
21610 207 RTS PC, ; Retorno.
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
; SUBROUTINA: LINEA 7.
; Argumentos:
; Entrada : En localidades 36750 a 36774, 25450 a 25455 están almacenados los
; caracteres ASCII a imprimirse.
; Salida : Se imprimen caracteres: blancos (a) blancos (b) del cuadro CCSTI.
;          ;Impresión de 61 caracteres " blancos".
;          ; Impresión de 24 "blancos"
;          ; En R0 se carga número (24) de caracteres a impresi-
;          ; mirse.
;          ; En R1 dirección (36750) de la localidad del ler.
;          ; carácter.
;          ; A subrutina de impresión.
;          ; Impresión de 24"blancos".
;          ;
;          ;
;          ; Misma explicación anotada en las 3 instrucciones
;          ; anteriores.
;          ;
;          ; Impresión de 11 caracteres " blancos".
;          ; En R0 se carga número (11) de caracteres a impresi-
;          ; mirse.
;          ; En R1 dirección (36750) de la localidad del ler.
;          ; carácter.
;          ; A subrutina de impresión.
;          ; Impresión de caracteres: " (a) " .
;          ; En R0 se carga número (5) de caracteres a impresi-
;          ; mirse.
;          ; En R1 dirección (25450) de la localidad del ler.
;          ; carácter.

```



```

; Cálculo del valor de la columna (a) del cuadro CCSTI:
; (Nº LLAMADAS EFECTIVAS/Nº LLAMADAS) X 100.
22036 013700 MOV #@40012,R0
      40012
; En R0 se carga dirección (40012) de la localidad
; en la que está almacenada la dirección de la loca-
; lidad en la que se encuentra el valor binario co-
; rrespondiente al N° de Llamadas Efectivas.
22042 013702 MOV #@40010,R2
      40010
; En forma análoga en R2 se carga dirección (40010)
; de la localidad que nos permite encontrar el va-
; lor binario del N° de Llamadas.
22046 4737 JSR PC,PORC
      25560
22052 240 NOP

```

```

; Cálculo del valor de la columna (b) de cuadro CCSTI:
; (Nº LLAMADAS EFECTIVAS TOTALES/Nº LLAMADAS TOTALES) X 100
22054 013700 MOV #@40026,R0
      40026
; En R0 se carga dirección (40026) de la localidad
; en la que está almacenada la dirección de la lo-
; calidad en la que se encuentra el valor binario
; correspondiente al N° de Llamadas Efectivas-Tota-
; les.
22060 013702 MOV #@40032,R2
      40032
; En R2 se carga dirección (40032) de la localidad
; en la que está almacenada la dirección de la loca-
; lidad en la que se encuentra el valor binario co-
; rrespondiente al N° de Llamadas-Totales.
22064 4737 JSR PC,PORC
      25560
22070 240 NOP
22072 240 NOP
22074 207 RTS PC
; Retorno

```

```

; SUBROUTINA: LINEA 9
; Argumentos:
; Entrada : En localidad 20774 se almacena dirección de retorno.
;           En localidades 24076 a 24111, 36750 a 36765, 24112 a 24136 están
;           almacenados los caracteres ASCCI a imprimirse.
; Salida  : Impresión de: "Efectivas" "blancos" "Duración media de" "blancos".
;           Se calcula e imprime el valor correspondiente a la DURACION MEDIA
;           DE ESTABLECIMIENTO PARA LLAMADAS EFECTIVAS, de la columna (a) y
;           (b) del cuadro CCSTII.
22100 4737 4737 JSR PC,CRLF ; A subrutina de retorno de carro, avance de línea.
      30030
22104 012700 MOV #13,RO ; Impresión de frase: "Efectivas".
      13 ; En RO se carga número (13) de caracteres a impri-
22110 012701 MOV #24076,R1 ; marse.
      24076 ; En R1 se carga dirección (24076) de la localidad
22114 4737 JSR PC,IMP ; del ler. carácter.
      30000 ; A subrutina de Impresión.
22120 4737 JSR PC,CRLF ; A subrutina retorno de carro, avance de línea.
      30030
22124 012700 MOV #15,RO ; Impresión de 15 caracteres "blancos".
      15 ; En RO se carga número (15) de caracteres a impri-
22130 012701 MOV #36750,R1 ; marse.
      36750 ; En R1 dirección (36750) de la localidad del pri-
22134 4737 JSR PC,IMP ; mer carácter.
      30000 ; A subrutina de Impresión.
22140 012700 MOV #24,RO ; Impresión de frase: "Duración Media de"
      24 ; En RO se carga número (24) de caracteres a impri-
22144 012701 MOV #24112,R1 ; marse.
      24112 ; En R1 se carga dirección (24112) de la localidad
22150 4737 JSR PC,IMP ; del ler. carácter.
      30000 ; A subrutina de impresión.
22154 012700 MOV #6,RO ; Impresión de 6 caracteres "blancos".
      6 ; En RO se carga número (6) de caracteres a impri-
22160 012701 MOV #36752,R1 ; marse.
      36752 ; En R1 dirección (36752) de la localidad del ler.
           ; carácter.

```

22164	4737 30000	JSR PC,IMP	; A subrutina de impresión.
22170	012700 1	MOV #1,RO	; Impresión de carácter "2" .
22174	012701 25472	MOV #25472,R1	; En RO se carga número (1) de caracteres a imprimirse.
22200	4737 30000	JSR PC,IMP	; En R1 se carga dirección (25472) de la localidad en que esta carácter. ; A subrutina de impresión.
22204	012706 10000	MOV #10000,SP	; Cálculo e Impresión del valor de la Duración Media de Establecimiento correspondiente a la columna (a) del cuadro CCSTI: (Llamadas Efectivas)
22210	012701 1	MOV #1,R1	; Se efectua el producto del Tiempo de Estabecimiento(BCD) por 6.El resultado se almacena en área BD: TSUT X 6 ——— BD(BCD)
22214	012702 40012	MOV #40012,R2	; Se efectúa el producto del N° de Llamadas Efectivas(binario) por 10. El resultado se almacena en área Bd en código BCD: NLLÉ(TRO) X 10 ——— Bd (BCD). ; Se divide el contenido del área BD para Bd y se imprime el resultado
22220	012705 40000	MOV #40000,R5	; Se inicializa puntero de pila en 10000. ; En R1 se almacena potencia (entera) a la cual se eleva la base 10, en este caso 1 y cuyo resultado se multiplica con el valor contenido de la localidad en que esta el N° de Llamadas Efectivas. ; En R2 se carga dirección (40012) de la localidad en la que está almacenada la dirección de la localidad en que se encuentra el valor (binario) correspondiente al N° Llamadas Efectivas.
22224	4737 22340	JSR PC,ATSNLE	; En R5 se carga dirección (40000) de la localidad en la que está almacenada la dirección de la localidad del área de memoria en la que se encuentra el Tiempo Establecimiento para Llamadas Efectivas(BCD). ; A subrutina que efectua el producto del contenido del área de memoria ( direccionada por R5) por 6 y el resultado (BCD) se almacena en el área BD. ; Además del producto contenido en la localidad (direccionado por R2) por 10 (R1) y el resultado (BCD) se almacena en el área Bd.

22230 240 ; División:(Tiempo de establecimiento x 6) / Nº Llamadas Efectivas por 10  
 ; NOP  
 22232 240 NOP  
 22234 012737 MOV #22246,@#20776 ; En localidad 20776 se almacena dirección (22246)  
 ; 22246 ; de retorno de próxima subrutina.  
 ; 20776  
 22242 4737 JSR PC,DIVBCD ; A subrutina que divide en BCD el contenido de las  
 ; 37570 ; áreas de memoria BD para Bd e imprime el resulta-  
 ; ; do.  
 22246 240 NOP  
 ; Cálculo e Impresión del valor de la Duración Media de Establecimiento corres -  
 ; pondiente a la columna (b) del cuadro CCSTII: (Llamadas Efectivas Totales).  
 ; Se efectúa el producto del Tiempo de Establecimiento(BCD) por 6.El resultado se  
 ; almacena en área BD: TSUT X 6 ——— BD(BCD)  
 ; Se efectúa el producto del Nº de Llamadas Efectivas Totales(binario).por 10.El  
 ; El resultado se almacena en área Bd en código BCD:NLE(Tot) X 10 ——— Bd(BCD)  
 ; Se divide el contenido del área BD para Bd y se imprime el resultado  
 ; MOV #10000, SP ; Se inicializa puntero de pila.  
 22250 012706 MOV #1,R1 ; En R1 se carga potencia (entera) a la cual se de-  
 ; 10000 ; be elevar la base 10 (en este caso 1) y cuyo re-  
 22254 012701 MOV #1,R1 ; sultado (10) se multiplica con el Nº de Llamadas  
 ; 1 ; Totales.  
 22260 012702 MOV #40026,R2 ; En R2 se carga dirección (40026) de la localidad  
 ; 40026 ; en la que ésta almacenada la dirección de la loca-  
 ; ; lidad en la que se encuentra el valor binario co-  
 22264 012705 MOV #40024,R5 ; rrespondiente al Nº de Llamadas Efectivas Totales  
 ; 40024 ; En R5 se carga dirección (40024) de la localidad  
 ; ; en la que está almacenada la dirección de la lra.  
 ; ; localidad del área de memoria en la que se encuen-  
 ; ; tra el valor BCD del Tiempo Establecimiento de  
 ; ; Llamadas Efectivas (total).  
 22270 4737 JSR PC,ATSNLE ; A subrutina que efectua el producto del contenido  
 ; 22340 ; del área de memoria ( direccionada por R5) por 6  
 ; ; y el resultado (BCD) se almacena en el área BD.  
 ; ; Además del producto contenido en la localidad (di-  
 ; ; reccionado por R2) por 10 (R1) y el resultado (BCD)  
 ; ; se almacena en el área Bd.

```

; División:
; (Tiempo de establecimiento(tot) x 6) / Nº Llamadas Efectivas(tot) por 10
22274 240 NOP
22276 240 NOP
22300 012737 MOV #22246,@#20776 ; En localidad 20776 se almacena dirección (22246)
22246 ; de retorno de próxima subrutina.
20776
22306 4737 JSR PC,DIVBCD ; A subrutina que divide en BCD el contenido de las
37570 ; áreas de memoria BD para Bd e imprime el resulta-
; do
22312 012706 MOV #20774,SP ; En puntero de pila se almacena dirección (20774)
20774 ; de localidad en que esta almacenada dirección de
22320 207 RTS PC ; Retorno.
*****
*****
*****
; SUBROUTINA: LINEA 9.1
; Argumentos:
; Entrada : En localidades 36750 a 36765, 24136 a 24157, 22472 a 22477 están
; almacenados los caracteres a ser impresos.
; Salida : Se imprime: "blancos", "Establecimiento", y "(seg)".
22500 4737 LINEST JSR PC,CRLF ; A subrutina de retorno de carro, avance de línea.
30030
22504 012701 MOV #36750,R1 ; Impresión de 15 caracteres blancos.
36750 ; En R1 se carga dirección (36750) de la localidad
22510 012700 MOV #15,R0 ; en la que está almacenado el 1er carácter a impri-
15 ; mirse.
4737 JSR PC,IMP ; En R0 número (15) de caracteres a ser impresos.
30000 ; A subrutina de impresión.
22520 012700 MOV #21,R0 ; Impresión de frase (caracteres verticales) "Esta
21 ; blecimiento".
22524 012701 MOV #24136,R1 ; En R0 se carga número (21) de caracteres a ser im-
24136 ; presos.
; En R1 dirección (24136) de la localidad del 1er.
; carácter a imprimirse.

```



```

22530      4737      JSR  PC,IMP      ; A subrutina de Impresión.
          30000

22534      012700    MOV  #5,R0      ; Impresión de frase "(seg)"
          5        ; En R0 se carga número (5) de caracteres a impri-
22540      012701    MOV  #22472,R1  ; mise.
          22472   ; En R1 dirección (22472) de la localidad del ler.
22544      4737      JSR  PC,IMP      ; carácter.
          30000   ; A subrutina de Impresión.

22550      207      RTS  PC      ; Retorno.
*****
*****
; SUBROUTINA: LINEA 10
; Argumentos:
; Entrada : En localidad 20774 se almacena dirección de retorno.
;          En localidades 36750 a 36765, 24112 a 24124, 24164 a 24175 y en
;          25474 se almacenan caracteres ASCCI a imprimirse.
; Salida  : Se imprime: "blancos", "Duración", "Tasable", "blancos", "3"
;          Se calcula e imprimen el valor de la DURACION TASABLE MEDIA DE LAS
;          LLAMADAS EFECTIVAS, de la columna (a) y (b) del cuadro CCSTI.
22560      4737      JSR  PC,CRLF   ; A subrutina retorno de carro, avance de línea.
          30030

22564      4737      JSR  PC,CRLF   ; A subrutina retorno de carro, avance de línea.
          30030

22570      012700    MOV  #15,R0     ; Impresión de 15 caracteres "blancos".
          15      ; En R0 se carga número (15) de caracteres a impri-
22574      012701    MOV  #36750,R1  ; mise.
          36750   ; En R1 dirección (36750) del ler. carácter.
22600      4737      JSR  PC,IMP      ; A subrutina de Impresión.
          30000

22604      012700    MOV  #13,R0     ; Impresión de frase: "Duración"
          13      ; En R0 se carga número (13) de caracteres a impri-
22610      012701    MOV  #24112,R1  ; mise.
          24112   ; En R1 dirección (24112) de la localidad del ler.
; carácter.

```

22614	4737	JSR	PC,IMP		; A subrutina de impresión.
	30000				
22620	012700	MOV	#11,R0		; Impresión de frase: "Tasable"
	11				; En R0 se carga número (11) de caracteres a imprimirse.
22624	012701	MOV	#24164,R1		; En R1 dirección (24164) de la localidad del ler.
	24164				; carácter.
22630	4737	JSR	PC,IMP		; A subrutina de Impresión.
	30000				
22634	012700	MOV	#12,R0		; Impresión de 12 caracteres "blancos".
	12				; En R0 se carga número (12) de caracteres a imprimirse.
22640	012701	MOV	36750,R1		; En R1 dirección (36750) de la localidad del ler.
	36750				; carácter.
22644	4737	JSR	PC, IMP		; A subrutina de impresión.
	30000				
22650	012700	MOV	#1,R0		; Impresión de carácter: "3".
	1				; En R0 se carga número (1) de caracteres a imprimirse.
22654	012701	MOV	25474,R1		; En R1 dirección (25474) de la localidad del ler.
	25474				; carácter.
22660	4737	JSR	PC,IMP		; A subrutina de Impresión.
	30000				
22664	012706				; Cálculo e Impresión del valor de la Duración Tasable correspondiente a la
	10000				; columna (a) del cuadro CCSTI: (llamadas efectivas)
22670	012702				; Se multiplica el número de Llamadas Efectivas (binario) por 100, el resultado
	40012				; (BCD) se almacena en el área Bd.
					; El Tiempo Tasable se almacena en el área BD(BCD).
					; Se divide en BCD el Tiempo Tasable para Número de Llamadas Efectivas y se im-
					; prime el resultado.
					; MOV #10000, SP ; Se inicializa el puntero de pila en 10000.
					; MOV #40012,R2 ; En R2 se carga dirección (40012) de la localidad
					; en que está almacenada la dirección de la locali-
					; dad en la que se encuentra el valor binario que
					; representa el Número de Llamadas Efectivas.

22674	012705 40004	MOV #40004,R5	; En R5 se carga dirección (40004) de la localidad ; en la que está almacenada la dirección de la lra. ; localidad del área de memoria en la que se encuen ; tra el valor BCD que representa el Tiempo Tasable ; En R1 se carga valor (2) al cual hay que elevar ; la base 10 para obtener el multiplicando 100. ; A subrutina que efectua los pasos 1) y 2).
22700	012701 2	MOV #2,R1	; Ejecución del paso 3).
22704	4737 26152	JSR PC,MUJAL	; En localidad 20776 se almacena dirección de retor ; no 22722.
22710	012737 22722 20776	MOV #22722,@#20776	
22716	4737 37570	JSR PC,DIVBCD	; A subrutina de división del contenido del área BD ; para Bd. ; Cálculo e Impresión del valor de la Duración Tasable de columna (b) del ; Cuadro CCSTI:(llamadas totales efectivas). ; Se multiplica el Número de Llamadas Efectivas Totales (binario) por 100, el re ; sultado (BCD) se almacena en el área Bd. El Tiempo Tasable Total (BCD) en el ; área Bd. ; Se divide en BCD el Tiempo Tasable Total para el Número de Llamadas Efecti- ; vas Total.
22722	012706 10000	MOV #10000, SP	; Inicialización del puntero de pila en 10000.
22726	012702 40026	MOV #40026,R2	; En R2 se carga dirección (40026) de la localidad ; en la que está almacenada la dirección de la loca ; lidad en la que se encuentra el valor binario que ; representa el Número de Llamadas Efectivas Tota- ; les.
22732	012705 40030	MOV #40030,R5	; En R5 se carga dirección (40030) de la localidad ; en la que está almacenada la dirección de la lra. ; localidad del área de memoria en la que se en ; cuentra el valor BCD que representa el Tiempo Ta- ; sable Total.
22736	012701 2	MOV #2,R1	; En R1 se carga el valor (2) el cual hay que ele-
22742	4737 26152	JSR PC,MUJAL	; var la base 10 para obtener el multiplicando 100. ; A subrutina que efectua los pasos 1) y 2).
22746	012737 22760 20776	MOV #22760,@#20776	; En localidad 20776 se almacena dirección de retor ; no 22760.

```

22754      4737      JSR PC,DIVBCD      ; A subrutina de división del contenido del área BD
          37570
22760      012706      MOV #20774,SP      ; Puntero de Pila se carga con dirección (20774) de
          20774      ; la localidad en la que está almacenada la direc-
          ; ción a la cual retorna esta subrutina.
22764      207        RTS PC      ; Retorno.
*****
*****
*****
; SUBROUTINA LINEA 10.1
; Argumentos:
; Entrada : Puntero de pila se carga con el valor 10000.
;          En localidades 36752 a 36767, 24125 a 24132, 24157 a 24163, están
;          almacenados los caracteres ASCCI a imprimirse.
;          Salida : Impresión: "blancos", "Media", "(min)".
22766      4737      JSR PC,CRLF      ; A subrutina de retorno de carro, avance de línea.
          30030
22772      012700      MOV #13,R0          ; Impresión de 13 caracteres "blancos".
          13
22776      012701      MOV #36752,R1       ; En R0 se carga número (13) de caracteres a impri-
          36752      ; mirse.
23002      4737      JSR PC,IMP          ; En R1 se carga dirección (36752) de la localidad
          30000      ; del ler. carácter.
          ; A subrutina de Impresión.
23006      012700      MOV #2,R0          ; Carga para que próximos caracteres se impriman
          2          ; verticalmente.
23012      012701      MOV #24112,R1       ; En R1 dirección (24112) de la localidad de pala-
          24112      ; bra del ler. carácter (escape)'
23016      4737      JSR PC,IMP          ; A subrutina de Impresión.
          30000
23022      012700      MOV #5,R0          ; Impresión de frase: "Media".
          5          ; En R0 se carga número (5) de caracteres a impri-
23026      012701      MOV #24125,R1       ; En R1 se carga dirección (24125) de la localidad
          24125      ; del ler. carácter.
23032      4737      JSR PC,IMP          ; A subrutina de Impresión.
          30000

```

```

23036 012700 MOV #5,R0 ; Impresión de frase: "(min)"
5 ; En R0 se carga número (5) de caracteres a impri-
; mirse.
23042 012701 MOV #24157,R1 ; En R1 se carga dirección (24157) de la localidad
24157 ; del 1er. carácter.
23046 4737 JSR PC,IMP ; A subrutina de Impresión.
30000
23052 207 RTS PC ; Retorno.
*****
*****
; PORCENTAJE: LINEA 11
; Argumentos:
; Entrada : En localidades a están almacenados los caracteres
; Salida : ASCII a imprimirse.
; : Se imprime: "Llamadas", "blancos" y "4".
; : Se calcula e imprime el valor correspondiente al PORCENTAJE DE LLA
; : MADAS INEFECTIVAS, de la columna (a) y (b) del cuadro CCSTI.
; :
23054 4737 LIN11 JSR PC,CRLF ; A subrutina retorno de carro, avance de línea.
30030
23060 4737 JSR PC,IMPLLA ; Impresión de frase: "Llamadas "
20474 ; A subrutina que imprime frase: "Llamadas y caracte-
; res" blancos".
23064 012700 MOV #1,R0 ; Impresión de carácter: "4" .
1 ; En R0 se imprime número (1) de caracteres a impri-
23070 012701 MOV #25476,R1 ; mirse.
25476 ; En R1 dirección (25476) de localidad del primer
23074 4737 JSR PC,IMP ; carácter
30000 ; A subrutina de Impresión.
; Cálculo e Impresión del valor de la columna (a) del Cuadro CCSTI: (Nº de
; Llamadas Inefectivas / Nº de Llamadas)x 100.
23100 013700 MOV @#40014,R0 ; En R0 se carga dirección (40014) de la localidad
40014 ; en que está almacenada la dirección de la locali-
; dad en la que se encuentra el valor binario co-
; rrespondiente al Nº de Llamadas Inefectivas.
23104 013702 MOV @#40010,R2 ; En forma análoga en R2 se carga dirección (40010)
40010 ; de la localidad que permite encontrar el valor
; binario del Nº de Llamadas.
;

```



23162	012700	MOV	#5, R0	;	Impresión de frase " vas "
	5			;	En R0 se carga número (5) de caracteres a impresi-
				;	mirse.
23166	012701	MOV	#24212, R1	;	En R1 se carga dirección (24212) localidad del
	24212			;	1er. carácter.
23172	4737	JSR	PC, IMP	;	A subrutina de Impresión.
	30000				
23176	012700	MOV	#10, R0	;	Impresión de 10 caracteres "blancos".
	10			;	En R0 se carga número (10) de caracteres a impresi-
				;	mirse.
23202	012701	MOV	#36752, R1	;	En R1 se carga dirección (36752) de localidad del
	36752			;	1er. carácter.
23206	4737	JSR	PC, IMP	;	A subrutina de Impresión.
	30000				
23212	012700	MOV	#24, R0	;	Impresión de frase: "Duración media de"
	24			;	En R0 se carga número (24) de caracteres a impresi-
				;	mirse.
23216	012701	MOV	#24112, R1	;	En R1 se carga dirección (24112) de la localidad
	24112			;	del 1er. carácter.
23222	4737	JSR	PC, IMP	;	A subrutina de impresión.
	30000				
23226	012700	MOV	#6, R0	;	Impresión de 6 caracteres "blancos".
	6			;	En R0 se carga número (6) de caracteres a impresi-
				;	mirse.
23232	012701	MOV	#36752, R1	;	En R1 se carga dirección (36752) de la localidad
	36752			;	del 1er carácter.
23236	4737	JSR	PC, IMP	;	A subrutina de impresión.
	30000				
23242	012700	MOV	#1, R0	;	Impresión del número "5"
	1			;	En R0 se carga número (1) de caracteres a impresi-
				;	mirse.

23246	012701	MOV #25500,R1	; En R1 se carga dirección (25500) de la localidad
	25500		; del 1er carácter.
23252	4737	JSR PC,IMP	; A subrutina de Impresión.
	30000		
23256	240	NOP	
23260	240	NOP	
23262	240	NOP	
23264	240	NOP	
23266	240	NOP	
23270	240	NOP	
23272	240	NOP	
23274	240	NOP	
			; Cálculo e Impresión del valor de la Duración Media de Establecimiento corres -
			; pondiente a la columna (a) del cuadro CCSTI: (Llamadas Inefectivas)
			; Se efectua el producto del Tiempo de Estabecimiento(BCD) por 6.El resultado se
			; almacena en área BD: TSUT X 6 <u>      </u> BD(BCD)
			; Se efectua el producto del Nº de Llamadas Inefectivas(binario).por 10.El resul
			; tado se almacena en área Bd en código BCD: NLE(TRO) X 10 <u>      </u> Bd (BCD).
			; Se divide el contenido del área BD para Bd y se imprime el resultado
23276	012706	MOV #10000,SP	; Inicialización del puntero de pila.
	10000		
23302	012701	MOV #1,R1	; En R1 se carga valor (1) al cual hay que elevar
	1		; base 10 para obtener el multiplicando 10.
23306	012702	MOV #40014,R2	; En R2 se carga dirección (40014) de la localidad
	40014		; en la que está almacenada la dirección de la loca
			; lidad en la que se encuentra el valor binario co-
			; rrespondiente al Nº de Llamadas Inefectivas.
23312	012705	MOV #40016,R5	; En R5 se carga dirección (40016) de la localidad
	40016		; en la que está almacenada la dirección de la Ira.
			; localidad del área de memoria en la que se encuen
			; tra el valor BCD Tiempo Establecimiento-Inefecti-
			; vas.
23316	4737	JSR PC,ATS NLE	; A subrutina que efectua el producto del contenido
	22340		; del área de memoria (direccionada por R5) por 6
			; y el resultado (BCD) se almacena en el área BD.
			; Además del producto contenido en la localidad (di
			; reccionado por R2) por 10 (R1) y el resultado (BCD)
			; se almacena en el área Bd.



; División:(Tiempo Establecimiento (Inefectivas))X6  
; ; /Nº Llamadas Inefectivas

23322 240 NOP  
23324 240 NOP  
23326 012737 MOV #23340,@#20776 ; En localidad 20776 se carga dirección (23340) de  
23340 20776 ; retorno de próxima rutina.

23334 4737 JSR PC,DIVBCD ; A subrutina que divide en BCD el contenido de las  
37570 ; áreas de memoria BD para Bd, e imprime el resulta  
; do.

23340 240 NOP  
23342 240 NOP  
23346 240 NOP  
23350 240 NOP  
23352 240 NOP  
23354 240 NOP  
23356 240 NOP

; Cálculo del valor Duración Media de Establecimiento (Inefectivas) de columna  
; (b) del cuadro CCSTI: (Llamadas Inefectivas).  
; El valor BCD, Tiempo de Establecimiento (Inefectivas-Totales), almacenado en  
; el área de memoria totales TSUTI se multiplica por 6 y el resultado en  
; BCD se almacena en el área BD.  
; Se efectúa el producto del Nº de Llamadas Inefectivas (binario) por 10 y el re  
; sultado en BCD se almacena en el área Bd.

23360 012706 MOV #10000,SP ; Inicialización del puntero de pila en 10000.

10000  
23364 012701 MOV #1,R1 ; En R1 se carga valor (1) al cual hay que elevar  
1 ; base 10 para obtener el multiplicando 10.

23370 012702 MOV #40034,R2 ; En R2 se carga dirección (40034) de la localidad  
40034 ; en la que está almacenada la dirección de la loca  
; lidad en la que se encuentra el valor binario co-

23374 012705 MOV #40036,R5 ; rrespondiente al Nº Llamadas Inefectivas Totales.  
40036 ; En R5 se carga dirección (40036) de la localidad  
; en la que está almacenada la dirección de la lra

```

23400 4737 JSR PC, ATSNLLE ; localidad del área de memoria en la que se encuen
23400 22340 ; tra el valor BCD, Tiempo de Establecimiento Ine-
; fectivas-Total.
; A subrutina que efectua el producto del contenido
; del área de memoria ( direccionada por R5) por 6
; y el resultado (BCD) se almacena en el área BD.
; Además del producto contenido en la localidad (di
; reccionado por R2) por 10 (R1) y el resultado (BCD)
; se almacena en el área Bd.
; División e Impresión de: (Tiempo Establecimiento (Inefectivas-Totales) x 6)/N°
; Llamadas Inefectivas Totales.
23404 240 NOP
240 NOP
23410 012737 MOV #23422, @#20776 ; En localidad 20776 se carga dirección (23422) de
23422 20776 ; retorno de próxima subrutina.
23416 4737 JSR PC, DIVBCD ; A subrutina que divide en BCD el contenido de las
37570 ; áreas de memoria BD para Bd e imprime el resulta-
; do.
23422 240 NOP
23424 012706 MOV #20774, SP ; En puntero de pila se almacena dirección (20774)
20774 ; de la localidad en la que se encuentra dirección
; de retorno.
23430 207 RTS PC ; Retorno.

```

```

*****
*****

```

; SUBROUTINA: LINEA 13.- CORTES DURANTE EL ESTABLECIMIENTO

```

23432 240 LIN13 NOP ; Impresión
23434 240 NOP
23436 240 NOP
23440 4737 JSR PC, CRLF ; A subrutina retorno de carro, avance de línea.
30030

```

23444	4737 30030	JSR PC,CRLF	; A subrutina retorno de carro, avance de línea.
23450	012700 36	MOV #36,R0	; Impresión de frase: "Análisis Cortes Durante el" ; En R0 se carga número (36) de caracteres a impri- ; mirse.
23454	012701 24220	MOV #24220,R1	; En R1 se carga dirección (24220) de la localidad ; del 1er. carácter.
23460	4737 30000	JSR PC,IMP	; A subrutina de Impresión.
23464	012700 7	MOV #7,R0	; Impresión de 7 caracteres "blancos". ; En R0 se carga número (7) de caracteres a impri- ; mirse.
23470	012701 36752	MOV #36752,R1	; En R1 se carga dirección (36752) de la localidad ; del 1er. carácter.
23474	4737 30000	JSR PC,IMP	; A subrutina de impresión.
23500	012700 1	MOV #1,R0	; Impresión de carácter "l" ; En R0 se carga número (1) de caracteres a impri- ; mirse.
23504	012701 25502	MOV #25502,R1	; En R1 se carga dirección (25502) de la localidad ; del 1er carácter.
23510	4737 30000	JSR PC,IMP	; A subrutina de impresión.
23514	240	NOP	
23516	012705 23664	MOV #23664,R5	; Se carga en área de memoria DESP los "desplaza- ; mientos". Cada palabra de 2 bytes (2 localidades ; adyacentes) constituye un valor del desplazamien- ; to cargado, se finaliza carga con el valor 0.
23522	012725 50	MOV #50,(R5)+	; En R5 se carga dirección (23664) de la localidad ; desde la cual se almacenan los desplazamientos. ; En localidades 23664/5 se almacena el valor 50.
23526	012725 52	MOV #52,(R5)+	; En localidades 23666/7 se almacena el valor 52.

23532	012725	MOV #64,(R5)+	; En localidades 23700/1 se almacena el valor 64.
	64		
23536	012725	MOV #66,(R5)+	; En localidades 23702/3 se almacena el valor 66.
	66		
23542	012725	MOV #70,(R5)+	; En localidades 23704/5 se almacena el valor 70.
	70		
23546	012725	MOV #72,(R5)+	; En localidades 23706/7 se almacena el valor 72.
	72		
23552	012725	MOV #102,(R5)+	; En localidades 23710/1 se almacena el valor 102.
	102		
23556	012725	MOV #104,(R5)+	; En localidades 23712/3 se almacena el valor 104.
	104		
23562	012725	MOV #116,(R5)+	; En localidades 23714/5 se almacena el valor 116.
	116		
23566	012725	MOV #120,(R5)+	; En localidades 23716/7 se almacena el valor 120.
	120		
23572	012725	MOV #126,(R5)+	; En localidades 23720/1 se almacena el valor 126.
	126		
23576	012725	MOV #130,(R5)+	; En localidades 23722/3 se almacena el valor 130.
	130		
23602	012725	MOV #132,(R5)+	; En localidades 23724/5 se almacena el valor 132.
	132		
23606	012725	MOV #134,(R5)+	; En localidades 23726/7 se almacena el valor 134.
	134		
23612	012725	MOV #136,(R5)+	; En localidades 23730/1 se almacena el valor 136.
	136		
23616	012725	MOV #140,(R5)+	; En localidades 23732/3 se almacena el valor 140.
	140		
23622	012725	MOV #226,(R5)+	; En localidades 23734/5 se almacena el valor 226.
	226		
23626	012725	MOV #242,(R5)+	; En localidades 23736/7 se almacena el valor 242.
	242		
23632	240	NOP	
23634	240	NOP	
23636	012725	MOV #0,(R5)+	; En localidades 23740/1 se almacena el valor 0.
	0		

23642	240	NOP	
23644	240	NOP	
23646	240	NOP	
23650	240	NOP	
		; Cálculo e Impresión del valor "Cortes Durante el Establecimiento" correspon-	
		; a la columna (a) y (b) del Cuadro CCSTI.	
23652	4737	JSR	PC,IMP ; A subrutina "Análisis de Llamadas Inefectivas.
	20630		
23656	240	NOP	
23660	240	NOP	
23662	207	RTS	PC ; Retorno.

; SUBROUTINA UTILIZADAS EN LAS RUTINAS DE LINEAS 3 A 25 (Capítulo IV-3)  
 ; SUBROUTINA: HORA INTERNACIONAL (GMT)

; Argumentos:

; Entrada: En localidad 20776 se almacena dirección de retorno  
 ; En registro R1 dirección de la 1ra localidad del área de memoria en  
 ; la que están almacenados los caracteres (EBCDIC) que representan  
 ; una fecha. Dicha área son 12 localidades secuenciales y en cada una  
 ; de una byte, está un carácter numérico. Los datos en el área son:  
 ; mes, día, año, hora y minuto; en que cada dato está representado por  
 ; 2 caracteres el (dígito más significativo ocupa el LOW BYTE de la  
 ; palabra y el menos significativo el HIGH BYTE).  
 ; SALIDA: En área de memoria HGMT se obtiene el equivalente de la Hora Inter-  
 ; nacional de la fecha contenida en el área direccionado por R1, ade-  
 ; más los datos de dicha área (localidad 24712 a 24744) están en cód-  
 ; go ASCII y en el siguiente orden: Año, mes, día hora y minuto la re-  
 ; presentación de los datos es la misma.

24770	012703	GMT	MOV	#31340, R3	; Transferencia del contenido del área direccionada
	31340				; R1 (área de origen) a área FCHC.
24774	012704		MOV	#5, R4	; En R3 se carga dirección de la 1ra localidad del
	5				; área FCHC (localidades 31340 a 31350).
25000	012706		MOV	#10000, SP	; En R4 se carga el número (5) de veces que se repi-
	10000				; te la transferencia del contenido de 2 localida-
25004	4737		JSR	PC, TRANS	; des (1 palabra de 2 bytes) del área de origen a
	32474				; área FCHC.
25010	240		NOP		; Inicialización de puntero de pila en 10000.
25012	012737		MOV	#5, @#100	; A subrutina de transferencia.
	5				; Inicialización del Contador en 5: CN = 5
25020	240		NOP		; En localidad 100 se carga el valor 5.
25022	240		NOP		

\*\*\*\*\*  
\*\*\*\*\*

nes de  
s (pa-  
or (NI

un nú-  
da en  
ada en  
reso

arte

I re-  
RO, R1.

25024	012737	LAZO	MOV	#25042, @#7006	; La fecha contenida en FCHC se suma 5 horas (Hora GMT).
	25042				; GMT).
	7006				; En localidad 7006 se almacena dirección de retorno (25042) de la próxima subrutina.
25032	012706		MOV	#10000, SP	; Inicialización de puntero de pila en 10000.
	10000				
25036	4737		JSR	PC, INCFCH	; A subrutina que incrementa en 1 hora la fecha contenida en FCHC.
	35200				; CN ——— CN - 1
25042	5337		DEC	@#100	; Decremento en una unidad el valor contenido en la localidad 100.
	100				; ¿ Es CN = 0 ?
25046	1366		BNE	LAZO	; Si, bifurca a LAZO.
25050	240		NOP		; No, continúa:
25052	240		NOP		
25054	012706		MOV	#10000, SP	; Conversión de los datos contenidos en FCHC de EBCDIC a ASCII.
	10000				; Inicialización del puntero de pila en 10000.
25060	012700		MOV	#31340, RO	; En RO se carga dirección (31340) de la lra.localidad del área de memoria FCHC.
	31340				; A subrutina de Conversión de EBCDIC a ASCII.
25064	4737		JSR	PC, CONEB/ASC	
	24600				
25070	240		NOP		
25072	012700		MOV	#31340, RO	; Distribución de los datos de FCHC a área HGMT.
	31340				; En RO se carga dirección (31340) de la lra.localidad del área FCHC.
25076	012701		MOV	#24716, R1	; En R1 se carga dirección (24716) de la lra.localidad del área de memoria HGMT.
	24716				; A subrutina de Distribución.
25102	4737		JSR	PC, DISTR	
	24750				
25106	240		NOP		
25110	240		NOP		
25112	012706		MOV	#20776, SP	; En puntero de pila se carga dirección (20776) de la localidad en la que se encuentra la dirección de retorno.
	20776				

25116 207 RET PC ; Retorno.  
 \*\*\*\*\*  
 \*\*\*\*\*

; SUBROUTINA PORCENTAJE (NI/NT) X 100

; Argumentos:  
 ; Entrada : En registros R0 y R2 se cargan, respectivamente, las direcciones de  
 ; localidades en las que están almacenados los valores binarios (pa-  
 ; labras de 16 bits) correspondientes al numerador y denominador (NI  
 ; y NT).  
 ; Salida : Se obtiene el valor del porcentaje, en el área NUASCCI, como un nú-  
 ; número de 4 dígitos: 2 dígitos para la parte entera almacenada en  
 ; la localidad 36435/6 y 2 para la parte fraccionaria, almacenada en  
 ; localidad 25464/5 (palabras de 2 bytes). Dicho número es impreso  
 ; la unidad I/O con el respectivo punto decimal

25560	5037	PORC	CLR	@#25464	; Se encera localidades donde se almacenará parte ; fraccionaria.
	25464				
25564	011000		MOV	(R0), R0	; Se obtiene numerador NI en registro R0.
25566	070027		MUL	#144, R0	; Se efectua el producto NI x 100 — R
	144				
25572	240		NOP		; Se multiplica I44 por el contenido de R0. El re-
25574	240		NOP		; sultado se almacena en el par de registros R0,R1.



25576	011202	MOV	(R2),R2	; Se divide R/NT.La parte entera del resultado son ; 2 caracteres ASCII almacenados en las localidades ; 36435/6 y la fraccionaria 2 caracteres BCD almace- ; nados en localidades 25464/5. ; Se obtiene denominador N2 en registro R2. ; A subrutina de división en Binario.
25600	4737	JSR	PC,DIBV	
25604	25372			
	240			
25606	012700	MOV	#14,R0	; Impresión de 14 caracteres blancos. ; En R0 se carga número (14) de caracteres a impresi- ; mirse. ; En R1 de carga dirección (36750) de la localidad ; del 1er carácter. ; A subrutina de Impresión.
25612	012701	MOV	#36750,R1	
25616	36750	JSR	PC,IMP	
	4737			
25622	30000	NOP		
	240			
25624	012700	MOV	#3,R0	; Impresión de la parte entera y del punto decimal. ; En R0 se carga número (3) de caracteres a impresi- ; mirse. ; En R1 se carga dirección (36435) de la localidad ; del 1er carácter. ; A subrutina de impresión.
25630	012701	MOV	#36435,R1	
25634	36435	JSR	PC,IMP	
	4737			
25640	30000	NOP		
	240			
25642	012700	MOV	#25464,R0	; Conversión de la parte fraccionaria (BCD) a ASCII ; En R0 se carga dirección (25464) de la localidad ; del 1er. carácter. ; A subrutina de conversión de BCD a ASCII.
25646	25464	JSR	PC,CONVB CDASC	
25652	4737			
	25540			
	240			
25654	012700	MOV	#2,R0	; Impresión de la parte fraccionaria. ; En R0 se carga número (2) de caracteres a impresi- ; mirse. ; En R1 se carga dirección (25464) de la localidad ; del 1er carácter.
25660	012701	MOV	#25464,R1	
	2			
	25464			

```

25664      4737      JSR PC,IMP      ; A subrutina de impresión.
           30000
25670      240       NOP
25672      240       NOP
25764      240       NOP
25676      207       RTS PC          ; Retorno.

```

```

*****
*****

```

**; SUBROUTINA: DIVISION EN BINARIO (D/d)**

```

; Argumentos:
; Entrada : En par de registros R0, R1 (palabras de 32 bits) se carga el
;           Dividendo.
; Salida  : En registro R2 se carga divisor. (palabra 16 bits).
;           En localidades 36435/6, se obtiene parte entera representada con 2
;           caracteres en ASCII y la parte fraccionaria con 2 dígitos BCD alma-
;           cenados en localidades 25464/5.

```

```

25372 012704 DIVBY MOV #25464,R4      ; En R4 se carga dirección (25464) de la lra locali-
           25464                      ; dad de la parte fraccionaria.
25376 071002 DIV R2,R0                ; Se divide D/d — R = (q, r)
25400 010003 MOV R0,R3                ; Se divide el contenido del par de registros R0,
25402 012705 MOV #2,R5                ; R1 para el contenido de R2. Cociente se obtiene
           2                          ; en registro R0 y residuo en R1 (r).
25406 022701 LAZO CMP #0,R1           ; Cociente (parte entera) se almacena en registro
           0                          ; R3.
25412 1407 BEQ SI                      ; Se inicializa el contador de dígitos de la parte
           1407                       ; fraccionaria, R5 en 2.
           0                          ; ¿Residuo = 0?
           1407                       ; El contenido de R1 es cero?
25414 010100 MOV R1,R0                          ; Si bifurca a "SI", no continua en la próxima ins-
25416 070027 MULT #12,R0               ; trucción.
           12                          ; Se multiplica : r X 10 ——— r
           12                          ; Se carga en registro R0 el valor contenido en R1.
           12                          ; Se multiplica el contenido de R0 por 12. El re-
           12                          ; sultado se obtiene en el par de registros R0,R1.

```

25422	071002	DIV	R2,R0	;	Se divide: r / d — R(q,r)
				;	Se divide el contenido del par de registros R0,
				;	R1 para el contenido de R2. El cociente se almace-
				;	na en R0 y residuo en R1.
25424	110024	MOVB	R0,(R4)+	;	El cociente se almacena en localidad apuntada
				;	por R4 (palabra de 1 byte). (Dígitos de parte
				;	fraccionaria).
				;	Se decrementa en 1 contador. Si Contador es dife-
				;	rente de cero va a LAZO.
25426	077511	SOB	R5,LAZO	;	El contenido de R5 se decrementa en 1. En caso
				;	de que sea diferente de cero bifurca a LAZO, si
				;	es cero continúa en próxima instrucción.
25430	240	NOP		;	
25432	010301	MOV	R3,R1	;	Conversión de la parte entera (Binario) a ASCCI.
				;	Se almacena en registro R1 la parte entera del re-
				;	sultado (q) contenido en R3.
25434	4737	JSR	PC,CONBYASCII	;	A subrutina de Conversión de Binario a ASCCI
	36336				
25440	240	NOP			
25442	240	NOP			
25444	240	NOP			
25446	207	RTS	PC		Retorno.

\*\*\*\*\*  
\*\*\*\*\*

```

; SUBRTUINA: T x 6.N x 10EK
; Argumentos:
; Entrada : Inicialización Registro "puntero de pila" en 1000.
;           Almacenar en registro R1 (palabras de 2 bytes) el exponente (K) a
;           la que se eleva la base 10.
;           En R2,dirección de la localidad en la que está almacenado el valor
;           el valor binario (16 bits) a ser multiplicados por el resultado
;           de 10K.
;           En R5, dirección de la 1ra localidad del área de memoria en la que
;           está almacenado el valor BCD que se multiplica por 6. Esta área
;           tiene 22 localidades . Cada dígito BCD es una palabra de 16 bits
;           (2 localidades) *Dígito menos significativo

```

; la primera palabra, y el más significativo última palabra.  
 ; Salida : El resultado del producto del número binario (direccionado por  
 ; R2), por el valor obtenido al elevar 10 al contenido de R1, se al-  
 ; macena en BCD área Bd (localidades 40260-40304).  
 ; El resultado del producto del valor contenido en el área de memo-  
 ; ria direccionado por R5, por 6 se almacena en código BCD en el  
 ; área BD (40230-40254).  
 ; \* Se utiliza Buffer Auxiliar localidades (25700 a 25724).

22340	010146	T6N10EK	MOV	R1,-(SP)	;	Se guardan en pila los argumentos de entrada. Pun-
22342	010246		MOV	R2,-(SP)	;	tero de pila apunta a localidad 7776).
22344	010546		MOV	R5,-(SP)	;	En localidades 7774/5 se almacena el valor conte-
22346	240		NOP		;	nido en R1 (K).
22350	4737		JSR	PC,Tx6	;	En localidades 7772/3 se almacena el valor conte-
	13120				;	nido en R2 (Número binario).
22354	240		NOP		;	En localidades 7770/1 se almacena el valor conte-
					;	nido en R5 (TBCD).
					;	Se efectúa el producto: T x 6 — BD (BCD).
22356	4737		JSR	PC,N10EK	;	A subrutina que obtiene el producto del valor BCD
	13220				;	contenido en el área direccionada por R5 por 6.
22362	012706		MOV	#7776,SP	;	Resultado en área Bd.
	7776				;	Se efectúa el producto N (memoria) x 10K — Bd
22366	207		RTS	PC	;	(BCD).
					;	A subrutina que obtiene el producto del valor bi-
					;	nario direccionado por R2 por el valor que resul-
					;	ta de elevar 10 a K.
					;	Se carga en puntero de pila la dirección de la lo
					;	calidad en la que está almacenado la dirección de
					;	retorno.
					;	Retorno.

; SUBROUTINA T X 6

; Argumentos:

; Entrada: En localidad 7770 dirección de la primera localidad del área de memoria en que se encuentra el valor BCD que se multiplica por 6.  
 ; Buffer Auxiliar (localidades 25700 a 25724).

; SALIDA: Resultado se obtiene en código BCD en el área BD (localidades 40230 - 40254). Cada dígito es una palabra de 2 bytes.

13120	012700	Tx 6	MOV	#13, R0	; Encerado de Buffer Auxiliar. ; En R0 se cargan número (13) de palabras a encerrar ; se. Cada palabra 2 bytes (2 localidades de memoria). ; En R1 se carga dirección de la primera localidad del Buffer ; A subrutina de Encerado.
13124	012701		MOV	#25700, R1	
13130	4737		JSR	PC, ENC	
13134	32520				
	240		NOP		
13136	013705		MOV	@#7770, R5	; Se calcula el producto del valor BCD, sumando 6 ; veces dicho valor, El resultado se obtiene en Buffer Auxiliar. ; En R5 se carga dirección (7770) de la localidad en la que se encuentra la dirección de la localidad del área de memoria en la que se encuentra el valor. ; A subrutina que suma 6 veces.
	7770				
13142	4737		JSR	PC, SUMBCD	
13146	25726				
	240		NOP		
13150	012701		MOV	#25700, R1	; Se transfiere el contenido del Buffer Auxiliar a área BD. ; En R1 se carga dirección (25700) de la localidad del Buffer Auxiliar.
13154	012703		MOV	#40230, R3	; En R3 se carga dirección (40230) de la localidad del área de memoria BD.
	40230				

```

13160 012704      MOV #13,R4      ; En R4 se carga el número de palabras (13) a trans-
; ferirse. Cada palabra 2 bytes (2 localidades).

13166      4737      JSR PC,TRANS      ; A subrutina de transferencia.
32474

13170      240       NOP
13172      240       NOP
13174      207       RTS
*****
; Retorno
*****

```

**; SUBROUTINA: N X 10 EK**

```

; Argumentos:
; Entrada : Buffer Auxiliar, localidades 25700 a 25724
;          ; En localidades 7774/5 (palabra de 2 bytes) el valor que representa
;          ; el exponente K.
;          ; En localidades 7772/3 (palabra de 2 bytes) dirección de la locali-
;          ; dad en la que está almacenada el valor binario representado por N.
; Salida  : En el área de memoria Bd (localidad 40260 a 4034 se obtiene el pro-
;          ; ducto en código BCD. Cada dígito una palabra de 2 bytes). En las 2
;          ; últimas palabras son ocupadas para K(K=1 o 2) Orden Dms a DMS.
;          ; Encerado de área Bd.
;          ; En R0 se carga número (13) de palabras a encerrar-
;          ; se. (cada palabra 2 bytes (2 localidades de memo-
;          ; ria).
;          ; En R1 dirección (40260) de la lra.localidad del
;          ; área de memoria Bd.
;          ; A subrutina de encerado.

13220 012700 N10EK      MOV #13,R0
13      13

13224 012701      MOV #40260,R1
40260

13230      4737      JSR PC,ENC
32520

13234      240       NOP

13236 012700      MOV #13,R0
13
; Encerado de Buffer Auxiliar.
; En R0 se carga número (13) de palabras a encerrar
; se cada palabra de 2 bytes (2 localidades de memo-
; ria).

```

13242	012701	MOV	#25700,R1	; En R1 dirección (25700) de la lra localidad del
	25700			; Buffer Auxiliar.
13246	4737	JSR	PC, ENC	; A subrutina de Encerado.
13252	32520			
	240	NOP		
13254	013702	MOV	@#7772,R2	; Se obtiene en área Bd el equivalente en BCD del
	7772			; número N(binario)
13260	4737	JSR	PC, CONBBBCD	; En R2 se carga dirección (7772) de la localidad
	25770			; en la que se encuentra la dirección del valor bi-
13264	240	NOP		; nario N.
13266	013701	MOV	@#7774,R1	; A subrutina de Conversión de binario a BCD.
	7774			
13272	012700	MOV	#40260,R0	; Al área Bd se multiplica x 10 <sup>k</sup>
	40260			; En R1 se carga dirección (7774) de la localidad
13276	4737	JSR	PC, DESP	; en la que se encuentra el exponente K.
	35600			; En R0 se carga con dirección (40260) de la lra lo
13302	240	NOP		; calidad del área Bd.
				; A subrutina de desplazamiento.
13304	010001	MOV	R0,R1	; Resultado anterior se transfiere a Buffer Auxi-
13306	012703	MOV	#25700,R3	; liar.
	25700			; Se carga en R1 dirección de la lra localidad del
13312	012704	MOV	#13,R4	; resultado almacenado en R0.
	13			; En R3 se carga dirección (25700) de la lra locali
13316	4737	JSR	PC, TRANS	; dad del Buffer Auxiliar.
	32474			; En R4 número (13) de palabras a ser transferidas
				; A subrutina de transferencia.
13322	012701	MOV	#25700,R1	; Transferencia del contenido del Buffer Auxiliar a
	25700			; área Bd.
				; En R1 se carga dirección (25700) de la lra locali
				; dad del Buffer Auxiliar.

```

13326 012703 MOV #40260,R3 ; En R3 se carga dirección (40260) de la lra localid
      40260 ; dad del área Bd.
13332 012704 MOV #13,R4 ; En R4 se carga número (13) de palabras a transfe-
      13 ; rirse.
13336 4737 JSR PC,TRANS ; A subrutina de transferencia.
      32474 ;
13342 207 RET PC ; Retorno.

```

```

*****
*****
*****

```

**; SUBROUTINA: "ANALISIS DE LAS LLAMADAS INEFECTIVAS" (capítulo IV-3)**

```

; Argumentos:
; Entrada : Es necesario que en la tabla de Desplazamientos se encuentren alma-
;          cenados los valores que representan dichos desplazamientos y en
;          que la última localidad esté el valor cero.
;          Se obtiene el valor correspondiente a la columna (a) del paráme-
;          tro en proceso del cuadro CCSII.
20630 013703 ALLI MOV @#40020,R3 ; R3 se carga con dirección (40020) de la localidad
      40020 ; en la que está almacenada la dirección de la lra
20634 012704 MOV #23664,R4 ; localidad del área C.R.D. (Tabla Troncal).
      23664 ;
20640 4737 JSR PC,SCRD ; R4 se carga con la dirección (23664) de la lra lo-
      20570 ; calidad del área "Desplazamiento"
      240 ; A subrutina suma Código Razón de Desconexión. Se
      ; obtiene, en localidad 20564, valor del parámetro.
      ;
      ; Se obtiene e imprime el valor:
      ; (ZCRD(TRO)) X 100 / NLLT
20646 012700 MOV #20564,R0 ; En R0 se carga dirección (20564) de la localidad
      20564 ; en que esté el valor binario (ZCRD)(acumulados de
      ; de los diferentes códigos).
20652 013702 MOV @#40010,R2 ; En R2 se carga dirección (40010) de la localidad
      40010 ; en la que está almacenado la dirección de la loca-
      ; lidad en que se encuentra el valor binario corres-
      ; pondiente al número de Llamadas (NLL) (Tablas de
      ; Troncal).

```



```

26214 47373 JSR PC, TRANSF ; A subrutina de transferencia.
      32474
26220 012706 MOV #7776, SP ; Se carga puntero de pila con direcci3n (7776) de
      7776 ; la localidad que est3 almacenada direcci3n de re-
      ; torno.
26224 207 RTS PC ; Retorno.
*****
***** ; SUBROUTINA: DIVISION BCD
***** ; Argumentos:
***** ; Entrada : En localidad 20776, direcci3n de retorno.
***** ; En 3reas de memoria BD (40230 a 40254) y Bd (localidades: 40260
***** ; 40304), en palabras de 10 bits (2 localidades de memoria) est3n el
***** ; dividendo y el divisor que representan 13 n3meros en c3digo BCD
***** ; El orden de los d3gitos en las 3reas es desde el Dms al DMS.
***** ; Salida : Se imprime en la unidad I/O el resultado de dividir BD para Bd.
***** ; Constituye un n3mero con 5 d3gitos para la parte entera y 2 para
***** ; la parte fraccionaria.
***** ; Evaluaci3n del Divisor y Dividendo.
***** ; En localidad 7770 se almacena direcci3n de retor-
***** ; no 37602
***** ; A subrutina de Evaluaci3n.
***** ; Divisi3n de BD para Bd, si se cumple que:BD> Bd
***** ; En localidad 7770 se almacena direcci3n de retor-
***** ; no 37620.
***** ; A subrutina de Divisi3n Condicionada.
*****
37570 012737 MOV #37602,@#7770
      37602
      7770
37576 4737 JSR PC,EVA
      37300
37602 240 NOP
37604 240 NOP
37606 012737 MOV #37620,@#7770
      37620
      7770
37614 4737 JRS PC,DIVCON
      37000
37620 240 NOP
37622 240 NOP

```

```

37624 012737      ; Impresión del resultado de división.
      37636      ; En localidad 7770 se almacena dirección de retor-
      7770      ; no.
37632      JSR PC,IMPDI      ; A subrutina de Impresión
37636      NOP
37640 012706      MOV #20776,SP      ; Puntero de pila se carga con dirección (20776) de
      20776      ; la localidad en que está almacenada dirección de
37644      RTS PC      ; Retorno.

```

```

*****
*****

```

#### SUBROUTINA: EVALUACION DEL DIVIDENDO Y DIVISOR

```

; Argumentos:
; Entrada : En localidad 7770 dirección de retorno.
;          Areas de memoria Bd y BD en código (BCD)
; Salida  : Si el valor contenido en Bd es mayor que el contenido en BD,
;          efectúa las multiplicaciones por 10 hasta que el valor de BD
;          sea mayor o igual que Bd y este valor lo almacena en el área
;          BD.
;          En localidad 40356 se almacena el número de ceros después del
;          decimal del número fraccionario en caso de que Bd > BD.
;          Se guarda contenido de la bandera de la impresión en localida
;          des 40362 : 1 BD > Bd
;                  0 Bd > BD
;          ; Es el dividendo (BD) mayor o igual que el Divisor
;          ; (Bd)?
;          ; En R0 se carga dirección (40230) de la lra locali
;          ; dad del área BD.
;          ; En R2 se carga dirección (40260) de la lra locali
;          ; dad del área Bd.
;          ; A subrutina de Comparación (si se cumple condi-
;          ; ción el contenido de R4 es 1, en caso contrario
;          ; es 0).
37300 012700      MOV #40230,R0
      40230
37304 012702      MOV #40260,R2
      40260
37310      JSR PC,COMP
      4737
37314      NOP
      35764
      240

```

37316	010437 40362	MOV R4, @#40362	<ul style="list-style-type: none"> <li>; Se guarda el resultado de la anterior comparación</li> <li>; (Si Bd&gt;BDD no se cumple condición: R4 = 0 el re-</li> <li>; sultado de la división solo tiene parte fracciona</li> <li>; ria).</li> <li>; Contenido de R4 se almacena en localidad 40362.</li> </ul>
37322	022704 0	CMP #0, R4	<ul style="list-style-type: none"> <li>; Si <math>Bd \geq Bd</math> se bifura a Salida?</li> <li>; Es el contenido de R4 igual a cero?</li> </ul>
37326	1106	BNE SALIDA	<ul style="list-style-type: none"> <li>; Si a SALIDA</li> </ul>
37330	240	NOP	<ul style="list-style-type: none"> <li>; No, continúa en la próxima instrucción</li> </ul>
37332	240	NOP	
37334	240	NOP	
37336	012703 40230	MOV #40230, R3	<ul style="list-style-type: none"> <li>; No implica que: <math>Bd &gt; BD</math>.</li> <li>; Obtención del número de dígitos más significati-</li> <li>; vos iguales a cero, del dividendo BD.</li> </ul>
37342	012705 13	MOV #13, R5	<ul style="list-style-type: none"> <li>; En R3 se carga dirección (40320) de la lra locali</li> <li>; dad del área BD.</li> </ul>
37346	4737 35620	JSR PC, DSIG	<ul style="list-style-type: none"> <li>; En R5 se carga número de palabras (13) del área</li> <li>; de memoria BD.</li> <li>; A subrutina de cálculo de dígitos.</li> </ul>
37352	240	NOP	
37354	240	NOP	
37356	240	NOP	
37360	010437 40356	MOV R4, @#40356	<ul style="list-style-type: none"> <li>; Se guarda número de ceros significativos N de BD.</li> <li>; Contenido de R4 se almacena en localidad 40356.</li> </ul>
37364	240	NOP	
37366	240	NOP	
37370	012703 40260	MOV #40260, R3	<ul style="list-style-type: none"> <li>; Obtención del número dígitos más significativos</li> <li>; iguales a cero del Divisor Bd.</li> <li>; En R3 se carga dirección (40260) de la lra locali</li> <li>; dad del área de memoria Bd.</li> </ul>

37374	012705	MOV	#13,R5	; En R5 se carga número (13) de palabras del área
	13			; Bd.
37400	4737	JSR	PC,DSIG	; A subrutina de cálculos de dígitos.
	35620			
37404	240	NOP		
37406	240	NOP		
				; Se guarda número de ceros significativos, N2 del
37410	010437	MOV	R4,@#40360	; área Bd.
	40360			; Contenido de R4 se almacena en localidad 40360.
37414	240	NOP		
37416	240	NOP		
				; Se efectúa la resta: N1 — N1 - N2
37420	163737	SUB	@#40360,@#40356;	; El contenido de la localidad (40356) menos el
	40360			; contenido de la localidad (40360) y el resultado
	40356			; se almacena en localidad 40356.
				; Es N1 = 0? (Es la cantidad de dígitos del dividen
				; do y divisor iguales?).
				; Si la diferencia anterior es cero se bifurca a A,
37426	1404	BEQ	A	; No continúa.
				; Si N1 = 0 la Bandera Auxiliar se pone en cero.
				; Se encera localidad 40364.
37430	5037	CLR	@#40364	
	40364			
37434	240	NOP		
37436	404	BR	B	; Bifurcación a "B".
				; Si N1 = 0 la Bandera Auxiliar se pone en 1.
37440	012737	MOV	#1,@#40364	; Se almacena en localidad 40364 el valor 1.
	1			
	40364			
37446	406	BR	C	; Se bifurca a "C".
			;N1 x BD	; El número N1 es el múltiplo de 10 por el cual hay
				; que multiplicar el dividendo BD para que la canti
				; dad de dígitos del Divisor y Dividendo sean igua-
				; les.
				; En R1 se carga contenido de la localidad 40356,
37450	013701	MOV	@#40356,R1	; (N1).
	40356			

37454	012700	MOV	#40230, R0	; En R0 se carga dirección (40230) de la lra localid-
	40230			dad del Dividendo Bd.
37460	4737	JSR	PC, DESPLA	; A subrutina de desplazamiento.
	35600			
37464	012702	MOV	#40260, R2	; Es N1 x BD > Bd? Número obtenido en anterior mul-
	40260			tiplicando es mayor o igual que el divisor?
37470	4737	JSR	PC, COMP	; En R2 se carga dirección de la lra localidad del
	35764			área de memoria Bd (Divisor).
37474	022704	CMP	#0, R4	; A subrutina de Comparación (si se cumple condi-
	0			ción R4 = 1 en caso contrario R4 = 0).
37500	1016	BNE	D	; Si NBD > Bd, se bifurca a D.
37502	022737	CMP	#0, @#40364	; Es el contenido de R4 igual a cero?
	0			
	40364			
37510	1402	BEQ	E	; Si va a E (N1 representa el número de ceros
				después de punto decimal).
				; No se encera N1 (No hay ceros del número fraccio-
				nario).
37512	5037	CLR	@#40356	; Se encera el contenido de la localidad 40356.
	40356			
37516	012701	MOV	#1, R1	; Se iguala cantidad de dígitos del dividendo y di-
	1			visor (NBD x 10).
37522	4737	JSR	PC, DESP	; En R1 se carga el valor 1
	35600			; A subrutina de desplazamiento.
37526	240	NOP		
37530	240	NOP		
37532	404	BR	SALIDA	
37534	240	NOP		

```

; El valor N1 restado 1, representa el número de ce
; ros, después de la coma, de la parte fraccionaria
; Se decrementa en 1 contenido de la localidad
; 40356.
; El contenido de R0 se transfiere a R5.
; En puntero de pila se carga dirección 7770 de la
; localidad en la que se encuentra la dirección de
; retorno.
; Retorno

```

```

37536 5337 D DEC @#40356
40356
37542 240 NOP
37544 010005 SALIDA MOV R0,R5
37546 012706 MOV #7770,SP
7770
37552 240 NOP
37554 207 RTS PC

```

```

*****
*****

```

**; SUBROUTINA: DIVISION CONDICIONADA**

```

; Argumentos:
; Entrada : En localidad 7770 dirección de retorno.
;          En áreas de memoria BD y Bd se almacenan los 13 (cada área) valo-
;          res BCD que representan el Divisor y el Dividendo
; Salida  : Si se cumple que BD > Bd , se efectua la división, obteniéndose
;          en la localidad 40350 el cociente (binario) y en localidades
;          40352 y 40354 el residuo (BCD),
;          Inicialización del puntero de pila en 10000.
; Transferencia del Divisor Bd a Buffer Auxiliar
; (BA).
; En R1 se carga dirección (40260) de la lra locali-
; dad del área de memoria Bd.
; En R3 se carga dirección (40310) de la lra locali-
; dad del Buffer Auxiliar.

```

```

37000 012706 MOV #10000,SP
10000
37004 012701 MOV #40260,R1
40260
37010 012703 MOV #40310,R3
40310

```

37014	012704	MOV	#13,R4	; En R4 se carga número (13) de palabras a transferirse.
	13			
37020	4737	JSR	PC, TRANS	; A subrutina de transferencia.
	32474			
37024	240	NOP		
37026	240	NOP		
37030	240	NOP		
37032	4737	JSR	PC, DIVRES	; Se efectúa la división de BD para BA. Se obtiene cociente (binario) de la parte entera del resultado en localidad 1400 y el residuo (BCD) el área BD.
	35670			; A subrutina que efectúa división (BD>BA)
37036	240	NOP		
37040	240	NOP		
37042	240	NOP		
37044	013737	MOV	@#1400, @#40350	; Se almacena anterior resultado en localidad 40350 (RPEN).
	1400			; El contenido de la localidad 1400 se transfiere a localidad 40350.
	40350			
37052	240	NOP		
37054	240	NOP		
37056	012737	MOV	#0, @#1420	; Se inicializa en 0 al contador del número de dígitos de la parte fraccionaria del resultado.
	0			; (CN — 0).
	1420			; Se encera localidad 1420.
37064	240	NOP		
37066	240	NOP		
37070	240	NOP		
37072	240	NOP		

; Es residuo de la anterior división cero?  
; (BD = 0?).

37074	012706	LAZO	MOV	#10000, SP	; Inicialización de puntero de pila en 10000.
	10000				
37100	012737		MOV	#37250, @#7774	; En localidad 7774 se carga dirección 37250 a que
	37250				; bifurca si se cumple condición BD = 0.
	7774				
37106	012700		MOV	#13, R0	; En R0 se carga número de palabras (2 bytes) que
	13				; contiene el área BD.
37112	010501		MOV	R5, R1	; Se transfiere el contenido del registro R5 a R1.
					; (R1 se carga con dirección de la lra localidad
					; del área BD).
37114	240		NOP		
37116	4737		JSR	PC, ACERO	; A subrutina que compara si el contenido del área
	36310				; de memoria BD contiene únicamente los valores 0?
37122	240		NOP		
37124	5237		INC	@#1420	; No es (BD = 0): CN — CN + 1
	1420				; Se incrementa en uno el contenido de la localidad
					; 1420.
37130	010500		MOV	R5, R0	; Multiplicación de 10 por BD (10 x Residuo).
37132	012701		MOV	#1, R1	; El contenido de R5 se transfiere a R0. R0 contiene
	1				; la lra localidad de área BD.
37136	4737		JSR	PC, DESP	; En R1 se carga potencia (1) a la cual hay que
	35600				; elevar base 10.
	240				; A subrutina de desplazamiento.
37142	240		NOP		
37144	012702		MOV	#40260, R2	; Es el producto 10 x BD (10 x residuo) mayor o
	40260				; igual que el divisor Bd.
37150	4737		JSR	PC, COMP	; En R2 se carga dirección (40260) de la lra localid
	35764				; dad del área BD (residuo).
					; A subrutina que compara 2 áreas de memoria. (Si
					; se cumple condición (mayor o igual) entonces
					; R4 = 1, caso contrario R4 = 0).
37154	240		NOP		
37156	022704		CMP	#1, R4	; Si 10 x BD > Bd entonces bifurca a B.
	1				; Es el contenido del R4 igual a 1?



37162	1412	BEQ B	;	Si a dirección 37210, no continúa:
37164	5037 1400	CLR @#1400	;	No es 10 BD > Bd:
			;	Se encera contenido de al localidad 1400.
37170	4737 26100	JSR PC,ALMR	;	Se almacena el cociente del resultado de las localidades 40352 - 40354 división (Si CN=0, dichas localidades se hacen cero).
37174	010005	MOV RO,R5	;	A subrutina de almacenamiento del resultado de división. Si CN = 1 cociente en localidad 40352
			;	CN = 2 cociente en localidad 40354.
37176	022737 2	CMP #2,@#1420	;	El contenido de RO se transfriere a R5.
37204	1420	BEQ SALIDA	;	Es CN = 2? (se finaliza cálculos de la parte fraccionaria).
37206	1424 746	BR LAZO	;	Es el contenido de la localidad 1420 igual a 2?
37210	010005	MOV RO,R5	;	Si a SALIDA.
37212	4737 35670	JSR PC,DIVRES	;	No a LAZO.
			;	Se efectúa división de 10 x BD para Bd (10 veces el residuo para divisor).Cociente en localidad 1400, residuo en área BD).
37216	4737 26100	JSR PC,ALMR	;	Se transfriere contenido del registro RO a R5.
37222	240	NOP	;	A subrutina que efectúa división (Si BD > Bd). cociente número binario, residuo número BCD.
37224	240	NOP	;	Se almacena cociente de división (Si CN = 0, parte fraccionaria 0).
37226	240	NOP	;	A subrutina de almacenamiento de la parte fraccionaria del resultado Si CN = 1 cociente en localidades 40352, CN = 2 en 40354.
37230	022737 2	CMP #2,@#1420	;	Es CN = 2?
	1420		;	Es el contenido de localidad 1420 igual a 2?

```

37236 1316 BNE LAZO ; Si a LAZO.
37240 405 BR SALIDA ; No a SALIDA.
37242 240 NOP
37244 240 NOP
37246 240 NOP

37250 4737 A JSR PC,ALMR ; Se almacena cociente de la división (Si CN = 0,
26100 ; entonces parte fraccionaria es cero).
; A subrutina que almacena la parte fraccionaria
; del resultado de la división. Si CN = 1 cociente
; en localidad 40352, CN = 2 en localidad 40354.

37254 240 NOP
37256 012706 SALIDA MOV #7770,SP ; En puntero de pila se carga dirección (7770) de
7770 ; la localidad en que se encuentra dirección de
37262 240 NOP
37264 207 RTS PC ; Retorno.

```

```

*****
*****

```

```

; SUBROUTINA: IMPRESION DE DIVISION (26272 - 26516)
; Impresión de 11 caracteres ASCCI "blancos".
26272 012700 MOV #11,R0 ; En R0 se carga número (11) de caracteres a impri
11 ; mirse.
26276 012701 MOV #36750,R1 ; En R1 se carga dirección (36750) de la localidad
36750 ; del ler. carácter.
26302 4737 JSR PC,IMP ; A subrutina de Impresión.
30000

26306 022737 CMP #0,#40362 ; Resultado de la división tiene solo parte fraccio
0 ; naria? ( F = 0).
40362 ; Contenido de la localidad 40362 es cero?
26314 1440 BNE A ; Si a A (Impresión de resultado solo con parte
; fraccionaria).

```

26316	013701 40350	MOV @#40350,R1	; No a Impresión del resultado (Parte entera + Frac ; ción). ; Conversión de binario a ASCII de la parte ; entera. ; En R1 se carga dirección (40350) de la localidad ; en la que se encuentra número binario (parte ente ; ra). ; A subrutina de conversión de binario a ASCII.
26322	4737 36336	JSR PC,CONVBAS	
26326	240	NOP	
26330	012700 5	MOV #5,RO	; Impresión de la parte entera del resultado. ; En RO se carga número (5) de localidades a ser im ; presos
26334	012701 36432	MOV #36432,R1	; En R1 se carga dirección (36432) de la lra locali ; dad del carácter.
26340	4737 30000	JSR PC,IMP	; A subrutina de Impresión.
26344	240	NOP	
26346	113737 40354 40353	MOVB @#40354,@#40353	; Conversión de BCD a ASCII de la parte fracciona- ; ria. ; Contenido de la localidad 40354 se transfriere a ; localidad 40353.
26354	012700 40352	MOV #40352,RO	; En RO se carga dirección (40352) de la localidad ; de los caracteres BCD.
26360	4737 25540	JSR PC,COMBCDASCCI	; A subrutina de Conversión.
26364	240	NOP	
26366	112737 56 40351	MOVB #56,@#40351	; Impresión de punto decimal y de la parte fraccio- ; naria. ; En localidad 40351 se almacena carácter ASCII 56 ; (punto decimal).
26374	240	NOP	
26376	012700 3	MOV #3,RO	; En RO se carga número (3) de caracteres a ; imprimirse.

26402	012701	MOV	#40351,R1	; En R1 dirección (40351) de la localidad del ler.
	40351			; carácter.
26406	4737	JSR	PC,IMP	; A subrutina de Impresión.
	30000			
26412	240	NOP		
26414	436	BR	SALIDA	; Salto a SALIDA.
				; Impresión de resultado con solo parte fraccionaria.
26416	062737	A		; ria.
	2	ADD	#2,@#40356	; En localidad 40356 se almacena número total de
	40356			; caracteres ceros a imprimirse después del punto
26424	013700	MOV	@#40356,R0	; decimal(caracteres cero, punto decimal).
	40356			; Se transfriere contenido de la localidad 40356 a
26430	012701	MOV	#26530,R1	; R0.
	26530			; En R1 se carga dirección (26530) de la localidad
26434	4737	JSR	PC,IMP	; del ler.carácter.
	30000			; A subrutina de impresión.
26440	240	NOP		
				; Conversión de BCD a ASCII de la parte fraccionaria.
26442	113737	MOVB	@#40352,@#40351	; ria.
	40352			; El contenido de la localidad 40352 se transfriere
	40351			; a localidad 40351.
26450	013737	MOV	@#40354,@#40352	; El contenido de la localidad 40354 se transfriere
	40354			; a localidad 40352.
	40352			
26456	012704	MOV	#3,R4	; En R4 se carga número (3) de caracteres a imprimirse.
	3			; mirse.
26462	012700	MOV	#40350,R0	; En R0 dirección (40350) de la localidad del ler.
	40350			; carácter
26466	4737	JSR	PC,COMBCDASCCI	; A subrutina de Conversión de BCD o ASSCCI.
	25544			
26472	240	NOP		
				; Impresión de la parte fraccionaria.
26474	012700	MOV	#3,R0	; En R0 se carga número (3) de caracteres a imprimirse.
	3			

26500	012701	MOV	R#40350, R1	; En R1 dirección (40350) de la localidad del 1er.
	40350			; carácter.
26504	4737	JSR	PC, IMP	; A subrutina de Impresión.
	30000			
26510	240	NOF		
26512	012706	MOV	#7770, SP	; En puntero de pila se carga dirección de la loca
	7770			; lidad en que está almacenada dirección de retorno
26516	207	RTS	PC	; Retorno.

ANEXO 6