IBM System / 360 Assembler
Programmschema

Programm DES/370		Lochhinweise	Problem (73 75)
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 1 (76 77) von 17

Name (1–8)	Operation (10–14)	Operand und Bemerkungen (20–70)	Karte Nr. (78–80)
		TITLE 'DES/370 v1.0, (c)&(p) Bernd R. Fix 1988-1995'	010
*			020
**********		**********	030
*		*	040
* PROGRAM-ID.		DES/370 EN-/DECRYPTION FACILITY. *	050
* AUTHOR.		BERND R. FIX. *	060
* DATE WRITTEN.		09/16/88. *	070
* COPYRIGHT.		- DES/370 ENCRYPTION FACILITY *	080
*		CONTAINS RESTRICTED MATERIAL OF THE AUTHOR.*	090
*		(C) COPYRIGHT BERND R. FIX 1988-1995 *	100
*		LICENSED MATERIALS - PROPERTY OF B.R. FIX *	110
*		REFER TO COPYRIGHT INSTRUCTIONS *	120
* REMARKS.		THIS PROGRAM UTILIZES THE DATA ENCRYPTION *	130
*		STANDARD (DES) TO EN-/DECRYPT DATA SETS *	140
*		SPECIFIED WITH A KEY SUPPORTED BY THE USER. *	150
*		*	160
**********		**********	170
*			180
CIPHER	CSECT		190
	LR	R12,R15	200
	USING	CIPHER,R12	
	ST	R14,RETURN	
	LA	R13,SAVE	

IBM System / 360 Assembler
Programmschema

Programm	DES/370	Lochhinweise	Problem
		Zeichen	
Programmierer	Bernd Fix	Datum 1987	Lochung	Seite 2 von 17

Name (1–8)	Operation (10–14)	Operand und Bemerkungen (20–70)	Karte Nr. (78–80)
	LA	R2,CRLF	010
	SVC	209	020
	LA	R2,HELLO	030
	SVC	209	040
	LA	R2,CRLF	050
	SVC	209	060
	L	R15,=V(PET)	070
	BALR	R14,R15	080
	MVI	MODE,0	090
	LA	R1,LINE1	100
	LA	R2,MYKEY	110
	BAL	R14,CONVERT	120
	LA	R2,CRLF	130
	SVC	209	140
	LA	R2,TXTKEY0	150
	SVC	209	160
	LA	R2,TXTKEY	170
	SVC	209	180
	LA	R11,MYKEY	190
	BAL	R14,KEYSCHED	200
	LA	R1,LINE2	
	LA	R2,BLOCK	
	BAL	R14,CONVERT	

IBM System / 360 Assembler
Programmschema

Programm	DES/370	Lochhinweise	Problem
		Zeichen	
Programmierer	Bernd Fix	Datum 1987	Lochung	Seite 3 von 17

Anweisung				Karte Nr.
Name (1–8)		Operation (10–14)	Operand und Bemerkungen (20–70)	(78–80)
		LA	R2,CRLF	Ø1Ø
		SVC	209	Ø2Ø
		LA	R2,TXTBLOCK	Ø3Ø
		SVC	209	Ø4Ø
		LA	R2,TXTDES1	Ø5Ø
		SVC	209	Ø6Ø
		LA	R11,BLOCK	Ø7Ø
		BAL	R14,DES	Ø8Ø
		MVC	TEST(8),BLOCK	Ø9Ø
		MVI	MODE,1	1ØØ
		LA	R1,LINE2	11Ø
		LA	R2,BLOCK	12Ø
		BAL	R14,CONVERT	13Ø
		LA	R2,TXTBLOCK	14Ø
		SVC	209	15Ø
		LA	R2,TXTDES2	16Ø
		SVC	209	17Ø
		LA	R11,TEST	18Ø
		BAL	R14,DES	19Ø
		LA	R1,LINE2	2ØØ
		LA	R2,TEST	
		BAL	R14,CONVERT	
		LA	R2,TXTBLOCK	

IBM System / 360 Assembler
Programmschema

Programm DES/370		Lochhinweise	Problem (73 75)
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 4 (76 77) von 17

Name (1–8)	Operation (10–14)	Operand und Bemerkungen (20–70)	Karte Nr. (78–80)
	SVC	209	Ø1Ø
	LA	R2,CRLF	Ø2Ø
	SVC	209	Ø3Ø
	L	R15,=V(PET)	Ø4Ø
	BALR	R14,R15	Ø5Ø
	LA	R2,CRLF	Ø6Ø
	SVC	209	Ø7Ø
	LA	R2,BYE	Ø8Ø
	SVC	209	Ø9Ø
	LA	R2,CRLF	1ØØ
	SVC	209	11Ø
	L	R14,RETURN	12Ø
	BR	R14 Programm beenden.	13Ø
*			14Ø
SAVE	DS	18F Sicherungsbereich.	15Ø
RETURN	DS	1F	16Ø
MYKEY	DC	X'8000000000000000'	17Ø
BLOCK	DC	X'0000000000000000'	18Ø
TEST	DC	X'0000000000000000'	19Ø
LINE	DC	16C' ',C'$'	2ØØ
HELLO	DC	C'DES/370 VERSION 1.00 (C) BERND FIX, 1987, 88$'	
TXTKEY0	DC	C'DEFINED KEY: &'	
LINE1	DC	16X'F0',C'$'	

IBM System / 360 Assembler
Programmschema

Programm DES/370		Lochhinweise	Problem (73 75)
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 5 (76 77) von 17

Name (1 8)	Operation (10 14)	Operand und Bemerkungen (20 25 30 35 40 45 50 55 60 65 70)	Karte Nr. (78 80)
TXTKEY	DC	C'CALL KEYSCHEDULE-ROUTINE...$'	010
TXTDES1	DC	C'ENCRYPTION OF BLOCK...$'	020
TXTDES2	DC	C'DECRYPTION OF BLOCK...$'	030
BYE	DC	C'DES/370 TERMINATED.$'	040
TXTBLOCK	DC	C'BLOCK: &'	050
LINE2	DC	16X'F0'	060
CRLF	DC	C'$'	070
*			080
CONVERT	UNPK	0(15,R1),0(8,R2)	090
	MVC	15(1,R1),14(R1)	100
	IC	R2,15(R1)	110
	SRL	R2,4	120
	STC	R2,15(R1)	130
	OC	14(2,R1),=X'F0F0'	140
	TR	0(16,R1),CVTTAB	150
	BR	R14	160
CVTTAB	DC	240C'*'	170
	DC	X'F0F1F2F3F4F5F6F7F8F9C1C2C3C4C5C6'	180
*			190
	LTORG		200
DES	STM	R14,R12,12(R13) Sichern der Register.	
	LA	R2,STACK Stackbereiche miteinander	
	ST	R2,8(R13) verketten.	

IBM System / 360 Assembler
Programmschema

Programm DES/370		Lochhinweise	Problem (73 75)
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 6 (76 77) von 17

Name (1–8)	Operation (10–14)	Operand und Bemerkungen (20–70)		Karte Nr. (78–80)
	ST	R13,4(R2)		010
	LR	R13,R2		020
	MVC	INBLK(8),0(R11)	Zu kodierenden Block holen.	030
	BAL	R14,INIPER	Eingangspermutation ausfuehren.	040
	LA	R8,KS	R8 = momentaner Schluessel.	050
	LA	R9,8	R9 = Offset naechster Schluessel	060
	TM	MODE,1	Umsetzungsmodus = Decipher ?	070
	BZ	$32	Nein: -->	080
	LA	R8,120(R8)	Reihenfolge K16 ... K1	090
	LNR	R9,R9	=> negativer Offset	100
$32	LA	R10,16	16 mal die Ver.-Funktion aufr.	110
	L	R1,OUTBLK	Die Variablen R und L festlegen.	120
	ST	R1,L		130
	L	R1,OUTBLK+4		140
	ST	R1,R		150
$14	L	R0,R	R0 = f(R0,Kn) ist die	160
	BAL	R14,FUNCTION	Verschluesselungsfunktion.	170
	X	R0,L	Erg. R0 ex-or mit L	180
	MVC	L(4),R	neues L = altes R	190
	ST	R0,R	neues R = altes L + f(R,Kn)	200
	AR	R8,R9	Naechsten Schluessel.	
	BCT	R10,$14	Schon 16 mal durch? Nein:-->	
	L	R1,R	Den Block fuer die Ausgangs-	

IBM System / 360 Assembler
Programmschema

Programm DES/370		Lochhinweise	Problem (73 75)
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 7 (76 77) von 17

Name (1–8)	Operation (10–14)	Operand und Bemerkungen (20–70)		Karte Nr. (78–80)
	ST	R1,INBLK	permutation nach INBLK bringen.	Ø1Ø
	L	R1,L		Ø2Ø
	ST	R1,INBLK+4		Ø3Ø
	BAL	R14,FINPER		Ø4Ø
	MVC	0(8,R11),OUTBLK		Ø5Ø
	L	R13,STACK+4		Ø6Ø
	LM	R14,R12,12(R13)	Register zurueckholen und	Ø7Ø
	BR	R14	Ruecksprung.	Ø8Ø
STACK	DS	18F	Rettungsbereich f. Register.	Ø9Ø
R	DC	F'0'	Variablen f. Kodierung.	1ØØ
L	DC	F'0'		11Ø
MODE	DC	X'00'		12Ø
*				13Ø
INIPER	LA	R2,8	Acht Bytes (64 Bits) werden	14Ø
	XR	R1,R1	permutiert.	15Ø
$1	IC	R1,PERTAB-1(R2)	Byte#(PERTAB) wird generiert.	16Ø
	LA	R3,8	Acht Bits sind zu bearbeiten.	17Ø
$2	LA	R6,INBLK-1(R3)	Permutation wird generiert,	18Ø
	ICM	R5,B'1000',0(R6)	indem in ein Ziel-Byte das jew.	19Ø
	SLDL	R4,1	linkeste Bit der 8 Quell-Bytes	2ØØ
	STCM	R5,B'1000',0(R6)	geschoben wird.	
	BCT	R3,$2	Schon alle Bits durch? Nein:-->	
	STC	R4,OUTBLK-1(R1)	Speichere Resultat.	

Programm	DES/370	Lochhinweise	Problem
		Zeichen	
Programmierer	Bernd Fix	Datum 1987	Lochung	Seite 8 von 17

Name (1–8)	Operation (10–14)	Operand und Bemerkungen (20–70)	Karte Nr. (78–80)
	BCT	R2,$1	Alle Bytes generiert? Nein:-->	010
	BR	R14	Fertig.	020
*				030
FINPER	LA	R2,8	Acht Bytes zu bearbeiten in	040
	XR	R1,R1	der Schluss-Permutation.	050
$3	IC	R1,PERTAB-1(R2)	Byte#(PERTAB) wird bearbeitet.	060
	LA	R6,INBLK-1(R1)		070
	ICM	R5,B'1000',0(R6)		080
	LA	R3,8	Acht Bits pro Durchgang.	090
$4	IC	R4,OUTBLK-1(R3)	Die 8 Bits des Q-Bytes werden	100
	SLDL	R4,1	von links her bitweise in die	110
	STC	R4,OUTBLK-1(R3)	8 Z-Bytes geschoben.	120
	BCT	R3,$4	Schon alle Bits durch? Nein:-->	130
	BCT	R2,$3	Schon alle Q-Bytes? Nein:-->	140
	BR	R14	Fertig.	150
*				160
	DS	0F		170
INBLK	DC	8X'00'	Quell-Bytes	180
OUTBLK	DC	8X'00'	Ziel-Bytes	190
PERTAB	DC	X'4,8,3,7,2,6,1,5'	Tabelle fuer FP	200
*				
KEYSCHED	STM	R14,R12,12(R13)	Sichern aller Register.	
	MVC	KEY(8),0(R11)		

IBM System / 360 Assembler
Programmschema

Programm DES/370		Lochhinweise	Problem (73 75)
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 9 (76 77) von 17

Name (1–8)	Operation (10–14)	Operand und Bemerkungen (20–70)		Karte Nr. (78–80)
	XR	R1,R1		010
	LA	R2,7	Sieben Bytes ist TEMP lang.	020
$5	IC	R1,TAB3-1(R2)	Davon Byte#(TAB3) berechnen.	030
	IC	R4,TEMP-1(R1)	TEMP ist das Resultat.	040
	LA	R3,8	Acht Bytes ist der KEY lang.	050
$6	LA	R6,KEY-1(R3)	Die Permuted Choice 1 (PC-1)	060
	ICM	R5,B'1000',0(R6)	laesst sich auch durch einfaches	070
	SLDL	R4,1	bitweises Schieben des jeweils	080
	STCM	R5,B'1000',0(R6)	linkesten KEY-Byte-Bits nach	090
	BCT	R3,$6	TEMP erzeugen. 8 x durchlaufen.	100
	STC	R4,TEMP-1(R1)	Speichere Resultat-Byte.	110
	BCT	R2,$5	Alle Bytes von TEMP erzeugen.	120
	L	R1,TEMP	Aus TEMP wird die Variable	130
	SRL	R1,4	C berechnet.	140
	ST	R1,C0	Speichere die Variable C.	150
	XR	R1,R1	Variable D gemaess der Defi-	160
	XR	R0,R0	nition berechnen.	170
	ICM	R1,B'1110',TEMP+4		180
	SRL	R1,4		190
	NI	TEMP+3,X'0F'		200
	IC	R0,TEMP+3		
	OR	R1,R0		
	ST	R1,D0	Variable D speichern.	

IBM System / 360 Assembler
Programmschema

Programm	DES/370	Lochhinweise	Problem
		Zeichen	
Programmierer	Bernd Fix	Datum 1987	Lochung	Seite 10 von 17

Name	Operation	Operand und Bemerkungen		Karte Nr.
	LA	R2,16	Insgesamt sind die Schluessel	010
	LA	R3,KS	K1 bis K16 (in KS) zu erzeugen.	020
$7	IC	R4,TAB4-1(R2)	Dazu werden C0 und D0 um (R4)	030
	XR	R0,R0	Berechne das rotierte C0.	040
	L	R1,C0		050
	SLDL	R0,4(R4)		060
	SRL	R1,4		070
	OR	R1,R0		080
	ST	R1,C0		090
	XR	R0,R0	Berechne das rotierte D0.	100
	L	R1,D0		110
	SLDL	R0,4(R4)		120
	SRL	R1,4		130
	OR	R1,R0		140
	ST	R1,D0		150
	XR	R1,R1		160
	LA	R7,4	Fuehre die Permuted Choice 2	170
	LA	R6,24	aus durch direktes Abfragen	180
$33	LA	R8,6	von Bitstellen der Bitstellen-	190
$34	IC	R1,PC2C-1(R6)	kombination CD.	200
	L	R0,MASKE		
	SRL	R0,3(R1)		
	L	R5,MASKE		

IBM System / 360 Assembler
Programmschema

Programm DES/370		Lochhinweise	Problem (73 75)
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 11 (76 77) von 17

Name (1–8)	Operation (10–14)	Operand und Bemerkungen (20–70)	Karte Nr. (78–80)
	IC	R4,WORK-1(R7)	010
	N	R0,C0	020
	BNZ	$35	030
	LA	R5,0	040
$35	SLDL	R4,1	050
	STC	R4,WORK-1(R7)	060
	IC	R1,PC2D-1(R6)	070
	L	R0,MASKE	080
	SRL	R0,3(R1)	090
	L	R5,MASKE	100
	IC	R4,WORK+3(R7)	110
	N	R0,D0	120
	BNZ	$36	130
	LA	R5,0	140
$36	SLDL	R4,1	150
	STC	R4,WORK+3(R7)	160
	BCTR	R6,0	170
	BCT	R8,$34	180
	BCT	R7,$33	190
	NC	WORK(8),SIGBIT	200
	MVC	0(8,R3),WORK	
	LA	R3,8(R3) Adressiere naechsten Schluessel.	
	BCT	R2,$7 Alle 16 Schluessel erzeugen.	

IBM System / 360 Assembler
Programmschema

Programm DES/370		Lochhinweise	Problem (73 75)
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 12 (76 77) von 17

Name (1–8)	Operation (10–14)	Operand und Bemerkungen (20–70)		Karte Nr. (78–80)
	LM	R14,R12,12(R13)	Hole Register.	Ø1Ø
	BR	R14	Fertig.	Ø2Ø
*				Ø3Ø
TAB3	DC	X'5,6,7,4,3,2,1'	Tabelle fuer KeySchedule	Ø4Ø
TAB4	DC	X'1,2,2,2,2,2,2,1,2,2,2,2,2,2,1,1'		Ø5Ø
PC2C	DC	AL1(5,1,24,11,17,14,10,21,6,15,28,3)		Ø6Ø
	DC	AL1(8,26,4,12,19,23,2,13,20,27,7,16)		Ø7Ø
PC2D	DC	AL1(27,19,9,3,24,13,20,5,17,23,12,2)		Ø8Ø
	DC	AL1(25,6,28,11,21,16,4,1,8,22,14,18)		Ø9Ø
MASKE	DC	B'10000000',AL3(0)	Maske fuer Bitstellenabfrage	1ØØ
SIGBIT	DC	8B'00111111'	Signifikante Bits	11Ø
TEMP	DC	2F'0'	Resultat-Speicher.	12Ø
WORK	DC	2F'0'	Arbeitsspeicher.	13Ø
C0	DC	F'0'	Variable	14Ø
D0	DC	F'0'	Variable	15Ø
KEY	DC	2F'0'	Schluessel fuer DES!	16Ø
KS	DC	32F'0'	16x 64-Bit-Schluessel Kn	17Ø
*				18Ø
FUNCTION	STM	R14,R12,12(R13)		19Ø
	LA	R2,8	Acht Bits pro WORK-Byte.	2ØØ
	LR	R5,R0	Auf 48 Bit expandieren.	
$8	LA	R3,4	4 Byte ist R regulaer.	
$9	IC	R4,WORK(R3)	Schiebe das jeweils linkeste	

Programm DES/370		Lochhinweise	Problem
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 13 von 17

Anweisung				Karte Nr.
Name (1–8)	Operation (10–14)	Operand und Bemerkungen (20–70)		(78–80)
	SLDL	R4,1	Bit von R in das naechste	010
	STC	R4,WORK(R3)	zykl. Byte von WORK.	020
	BCT	R3,$9	Alle 4 Bytes mit	030
	BCT	R2,$8	jeweils 8 Bit durch? Nein:-->	040
	XR	R1,R1		050
	IC	R1,WORK+1	Die beiden "Rand"Bytes von	060
	LR	R0,R1	WORK sind um eine Stelle	070
	SLL	R0,7	nach links bzw. rechts ge-	080
	SRL	R1,1	schobene Bytes des inneren	090
	OR	R1,R0	WORK-Feldes.	100
	STC	R1,WORK+5	Auf diese Weise werden aus	110
	XR	R1,R1	den 32 "inneren" Bits durch	120
	IC	R1,WORK+4	Hinzufuegen von 2 Bytes (16 Bit)	130
	SLL	R1,1	die benoetigten 48 Bit.	140
	LR	R0,R1		150
	SRL	R0,8		160
	OR	R1,R0		170
	STC	R1,WORK		180
	LA	R2,8	Die Bits des Feldes WORK	190
	LA	R1,0	muessen noch in die nieder-	200
*			wertigsten 6 Bits der Bytes	
$10	LA	R3,6	im Feld TEMP gebracht werden,	
$11	LA	R6,WORK-1(R3)	was durch shiften des jeweils	

IBM System / 360 Assembler
Programmschema

Programm DES/370		Lochhinweise	Problem (73 75)
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 14 (76 77) von 17

Name (1 8)	Operation (10 14)	Operand und Bemerkungen (20 25 30 35 40 45 50 55 60 65 70)		Karte Nr. (78 80)
	ICM	R5,B'1000',0(R6)	linkesten Bits eines der 6 Bytes	Ø1Ø
	SLDL	R4,1	in ein Byte von TEMP geschieht.	Ø2Ø
	STCM	R5,B'1000',0(R6)		Ø3Ø
	BCT	R3,$11		Ø4Ø
	N	R4,SIGBIT		Ø5Ø
	STC	R4,TEMP(R1)		Ø6Ø
	LA	R1,1(R1)		Ø7Ø
	BCT	R2,$10		Ø8Ø
	XC	TEMP(8),0(R8)	Exklusiv-Oder Verknuepfung	Ø9Ø
*			zwischen TEMP und Kn !	1ØØ
	XR	R4,R4		11Ø
	LA	R2,8	Aus den 48 Bits, die in 8 Bytes	12Ø
	LA	R3,STAB	untergebracht sind, werden jetzt	13Ø
$12	IC	R4,TEMP-1(R2)	nach Tabellen S1 bis S8 jeweils	14Ø
	IC	R4,0(R4,R3)	6 Bit auf 4 Bit komprimiert.	15Ø
	SRDL	R4,4	Das Ergebnis hat dann 4 Bit x 8,	16Ø
	LA	R3,64(R3)	also genau die benoetigten	17Ø
	BCT	R2,$12	32 Bit, die auch R und L lang	18Ø
	ST	R5,TEMP	sind.	19Ø
	LA	R2,32	Diese 32 Bit werden vor Rueck-	2ØØ
$37	IC	R1,PTAB-1(R2)	kehr zum rufenden Programm noch	
	L	R0,MASKE	permutiert.	
	SRL	R0,0(R1)		

IBM System / 360 Assembler
Programmschema

Programm DES/370		Lochhinweise	Problem
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 15 von 17

Name (1–8)	Operation (10–14)	Operand und Bemerkungen (20–70)	Karte Nr. (78–80)
	L	R5,MASKE	010
	N	R0,TEMP	020
	BNZ	$38	030
	LA	R5,0	040
$38	SLDL	R4,1	050
	BCT	R2,$37	060
	ST	R4,TEMP	070
	LM	R14,R12,12(R13)	080
	L	R0,TEMP In R0 steht das Ergebnis.	090
	BR	R14 Fertig.	100
*			110
PTAB	DC	AL1(24,3,10,21,5,29,12,18,8,2,26,31,13,23,7,1)	120
	DC	AL1(9,30,17,4,25,22,14,0,16,27,11,28,20,19,6,15)	130
STAB	DC	AL1(13,1,2,15,8,13,4,8,6,10,15,3,11,7,1,4)	140
	DC	AL1(10,12,9,5,3,6,14,11,5,0,0,14,12,9,7,2)	150
	DC	AL1(7,2,11,1,4,14,1,7,9,4,12,10,14,8,2,13)	160
	DC	AL1(0,15,6,12,10,9,13,0,15,3,3,5,5,6,8,11)	170
	DC	AL1(4,13,11,0,2,11,14,7,15,4,0,9,8,1,13,10)	180
	DC	AL1(3,14,12,3,9,5,7,12,5,2,10,15,6,8,1,6)	190
	DC	AL1(1,6,4,11,11,13,13,8,12,1,3,4,7,10,14,7)	200
	DC	AL1(10,9,15,5,6,0,8,15,0,14,5,2,9,3,2,12)	
	DC	AL1(12,10,1,15,10,4,15,2,9,7,2,12,6,9,8,5)	
	DC	AL1(0,6,13,1,3,13,4,14,14,0,7,11,5,3,11,8)	

IBM System / 360 Assembler
Programmschema

Programm DES/370		Lochhinweise	Problem
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 16 von 17

Name (1–8)	Operation (10–14)	Operand und Bemerkungen (20–70)	Karte Nr. (78–80)
	DC	AL1(9,4,14,3,15,2,5,12,2,9,8,5,12,15,3,10)	010
	DC	AL1(7,11,0,14,4,1,10,7,1,6,13,0,11,8,6,13)	020
	DC	AL1(2,14,12,11,4,2,1,12,7,4,10,7,11,13,6,1)	030
	DC	AL1(8,5,5,0,3,15,15,10,13,3,0,9,14,8,9,6)	040
	DC	AL1(4,11,2,8,1,12,11,7,10,1,13,14,7,2,8,13)	050
	DC	AL1(15,6,9,15,12,0,5,9,6,10,3,4,0,5,14,3)	060
	DC	AL1(7,13,13,8,14,11,3,5,0,6,6,15,9,0,10,3)	070
	DC	AL1(1,4,2,7,8,2,5,12,11,1,12,10,4,14,15,9)	080
	DC	AL1(10,3,6,15,9,0,0,6,12,10,11,1,7,13,13,8)	090
	DC	AL1(15,9,1,4,3,5,14,11,5,12,2,7,8,2,4,14)	100
	DC	AL1(10,13,0,7,9,0,14,9,6,3,3,4,15,6,5,10)	110
	DC	AL1(1,2,13,8,12,5,7,14,11,12,4,11,2,15,8,1)	120
	DC	AL1(13,1,6,10,4,13,9,0,8,6,15,9,3,8,0,7)	130
	DC	AL1(11,4,1,15,2,14,12,3,5,11,10,5,14,2,7,12)	140
	DC	AL1(15,3,1,13,8,4,14,7,6,15,11,2,3,8,4,14)	150
	DC	AL1(9,12,7,0,2,1,13,10,12,6,0,9,5,11,10,5)	160
	DC	AL1(0,13,14,8,7,10,11,1,10,3,4,15,13,4,1,2)	170
	DC	AL1(5,11,8,6,12,7,6,12,9,0,3,5,2,14,15,9)	180
	DC	AL1(14,0,4,15,13,7,1,4,2,14,15,2,11,13,8,1)	190
	DC	AL1(3,10,10,6,6,12,12,11,5,9,9,5,0,3,7,8)	200
	DC	AL1(4,15,1,12,14,8,8,2,13,4,6,9,2,1,11,7)	
	DC	AL1(15,5,12,11,9,3,7,14,3,10,10,0,5,6,0,13)	
*			

IBM System / 360 Assembler
Programmschema

Programm DES/370		Lochhinweise	Problem (73 75)
		Zeichen	
Programmierer Bernd Fix	Datum 1987	Lochung	Seite 17 (76 77) von 17

Name (1 8)	Operation (10 14)	Operand und Bemerkungen (20 25 30 35 40 45 50 55 60 65 70)	Karte Nr. (78 80)
R0	EQU	0	010
R1	EQU	1	020
R2	EQU	2	030
R3	EQU	3	040
R4	EQU	4	050
R5	EQU	5	060
R6	EQU	6	070
R7	EQU	7	080
R8	EQU	8	090
R9	EQU	9	100
R10	EQU	10	110
R11	EQU	11	120
R12	EQU	12	130
R13	EQU	13	140
R14	EQU	14	150
R15	EQU	15	160
*			170
	END	CIPHER	180
*			190
			200