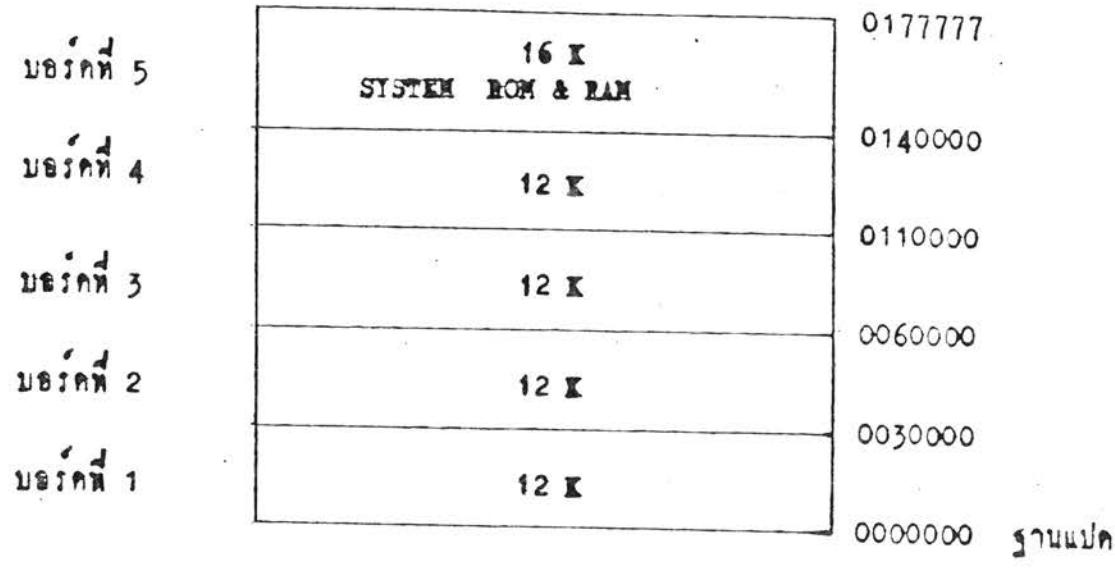
บทที่ 2

ส่วนประกอบและลักษณะทั่วไปของเครื่องคาคาพอยท์

2.1 ส่วนประกอบและลักษณะทางฟิสิกส์ (Physical Structure)

เครื่องคาคาพอยท์เป็นเครื่องคอมพิวเตอร์ที่ใช้ในการเตรียมข้อมูลโดยเป็นระบบเครื่องในลักษณะคลัสเตอร์ (Cluster) ประกอบด้วยตัวโปรเซสเซอร์ (Processor) จอภาพแสดงข้อมูล (CRT Display) แป้นสำหรับรับข้อมูล (Keyboard) เครื่องอ่านและบันทึกเทปคลับ (Cassette Tape Dicks) เครื่องอ่านและบันทึกจานแม่เหล็ก (Disk Drive) เครื่องอ่านและบันทึกเทปแม่เหล็ก (Tape Drive) และเครื่องที่ต่อจากภายนอก (External Peripherals) ดังมีรายละเอียดต่อไปนี้

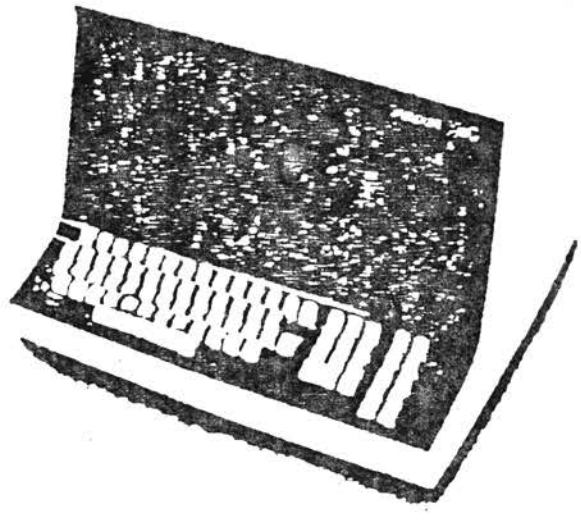
2.1.1 โปรเซสเซอร์ (Processor) ประกอบด้วยรีจิสเตอร์ (Registers) 2 ชุด แต่ละชุดประกอบด้วยรีจิสเตอร์ 8 ตัว รีจิสเตอร์แต่ละตัวมีขนาด 8 บิท (bit) ซึ่งสามารถโปรแกรมให้ทำงานในลักษณะต่าง ๆ ได้ มีธัญญานแฟลกควบคุม (Control Flag) 4 ธัญญาน, 16 บิท โปรแกรมเคาน์เตอร์ (16 - bit Program Counter), 8 บิทเบสรีจิสเตอร์ (8 - bit Base Register), พุชดาวน์สแตค (Push - down stack) 16 ระดับ, 8 บิทอินสตรัคชั่นรีจิสเตอร์ (8 - bit Instruction Register), 4 บิทอินสตรัคชั่นโมดิฟิเคชั่นรีจิสเตอร์ (4 - bit Instruction Modification Register) และ 16 - เวิร์ด เมมโมรีเซคเตอร์เทเบิล (16 - word Memory Sector Table) รวมทั้งส่วนความจำหลักที่ผู้ใช้สามารถใช้งานได้อีก 48 เคไบท์ (48 k - byte) ซึ่งส่วนความจำทั้งหมดของเครื่องมี 64 เคไบท์ แบ่งออกเป็นบอร์ด (Board) บอร์ดที่ 1 ถึงบอร์ดที่ 4 มีขนาดเท่ากันคือ 12 เคไบท์ ผู้ใช้สามารถโปรแกรมใช้งานได้ สำหรับบอร์ดที่ 5 มีขนาด 16 เคไบท์ ได้ถูกกำหนดไว้ใช้เป็นซิสเตมรอมและแรม (System ROM & RAM) ดังแสดงในรูปที่ 2.1



รูปที่ 2.1 แสดงการแบ่งส่วนความจำหลักในลักษณะของบอร์ด

2.1.2 จอภาพแสดงข้อมูล (CRT Display) เป็นจอภาพที่ใช้ติดต่อกับโปรเซสเซอร์ของเครื่องโดยตรง ซึ่งอาจเรียกได้อีกชื่อหนึ่งคือ "คอนโซล" (Console) จอภาพแสดงข้อมูลนี้สามารถแสดงข้อมูลได้มากที่สุด 960 ตัวอักษร โดยแบ่งจอภาพออกเป็น 12 บรรทัด บรรทัดละ 80 ตัวอักษร โปรเซสเซอร์ให้แอดเดรส(Address) แก่คอนโซลเป็น 0341 ฐานแปด นอกจากนี้ในส่วนของคอนโซลยังมีแรมซึ่งสามารถใช้ในการบันทึกและอ่านรูปแบบของตัวอักษรในลักษณะโครงสร้างที่เป็นจุดได้ถึง 128 ตัวอักษร โดยแต่ละตัวอักษรประกอบด้วยจุดในแนวระดับ 5 แถว และจุดในแนวราบ 7 แถว (5 X 7 Dot Matrix) ความเร็วสูงสุดในการนำข้อมูลจากส่วนความจำของคอนโซลให้ไปปรากฏบนจอภาพประมาณ 50,000 ตัวอักษรต่อวินาที

2.1.3 แป้นสำหรับรับข้อมูล(Keyboard) ทำหน้าที่รับคำสั่งที่ควบคุมโปรเซสเซอร์หรือการป้อนข้อมูลต่าง ๆ โดยที่แป้นสำหรับรับข้อมูลจะประกอบด้วยแป้นตัวอักษรปกติ 55 คีย์(Key) แป้นตัวเลข 11 คีย์ และคีย์ที่ใช้ในการควบคุมโปรเซสเซอร์อีก 5 คีย์ คีย์เหล่านี้เมื่อถูกผู้ใช้กดจะสร้างรหัส(Code) ของคีย์นั้น ๆ ส่งให้กับโปรเซสเซอร์คีย์ละ 8 บิท ซึ่งก็คือรหัสของหนึ่งตัวอักษรนั่นเอง นอกจากนี้ในหนึ่งคีย์ยังสามารถให้รหัสได้สองรหัสโดยการเลือกชิฟหรือไม่ชิฟ(Shift or Un - Shift) โปรเซสเซอร์ให้แอดเดรสกับแป้นสำหรับรับข้อมูลเป็น 0341 ฐานแปด เช่นเดียวกับจอภาพแสดงข้อมูล ดังนั้นถ้าบ่งแอดเดรสเป็น 0341 ฐานแปดและบอกว่าเป็นการส่งข้อมูลออกจากโปรเซสเซอร์ก็จะหมายถึงส่งข้อมูลไปยังจอภาพแสดงข้อมูลแต่ถ้าบ่งแอดเดรส 0341 ฐานแปด และบอกว่าเป็นการรับข้อมูลเข้าสู่โปรเซสเซอร์ก็จะหมายถึงการรับข้อมูลจากแป้นสำหรับรับข้อมูล จอภาพแสดงข้อมูลและแป้นสำหรับรับข้อมูลแสดงไว้ในรูปที่ 2.2 ส่วนรหัสมาตรฐานที่ได้จากแป้นสำหรับรับข้อมูลแสดงไว้ในรูปที่ 2.3



รูปที่ 2.2 แสดงจอภาพแสดงข้อมูลและแป้นสำหรับรับข้อมูล

รหัส	ตัวอักษร	รหัส	ตัวอักษร	รหัส	ตัวอักษร	รหัส	ตัวอัก
000	Null	040	SP	100	@	140	`
001		041	!	101	A	141	a
002		042	"	102	B	142	b
003	Roll down	043	#	103	C	143	c
004		044	$	104	D	144	d
005		045	%	105	E	145	e
006		046	&	106	F	146	f
007	Bell	047	'	107	G	147	g
010	BSP	050	(	110	h	150	h
011	TAB	051	)	111	I	151	i
012	Line Feed	052	*	112	J	152	j
013	Roll up	053	+	113	K	153	k
014	New Line	054	,	114	L	154	l
015	Enter	055	-	115	M	155	m
016		056	.	116	N	156	n
017		057	/	117	O	157	o
020		060	0	120	P	160	p
021		061	1	121	Q	161	q
022		062	2	122	R	162	r
023		063	3	123	S	163	s
024	PTR off	064	4	124	T	164	t
025	Home up	065	5	125	U	165	u
026	EEOL	066	6	126	V	166	v
027	EEOF	067	7	127	W	167	w
030	Cursor on	070	8	130	X	170	x
031	Cursor off	071	9	131	Y	171	y
032	PTR on	072	:	132	Z	172	z
033	Cancel	073	;	133	[	173	{
034	Interrup	074	<	134	\	174	|
035		075	=	135	]	175	}
036		076	>	136	^	176	~
037		077	?	137	_	177	DE

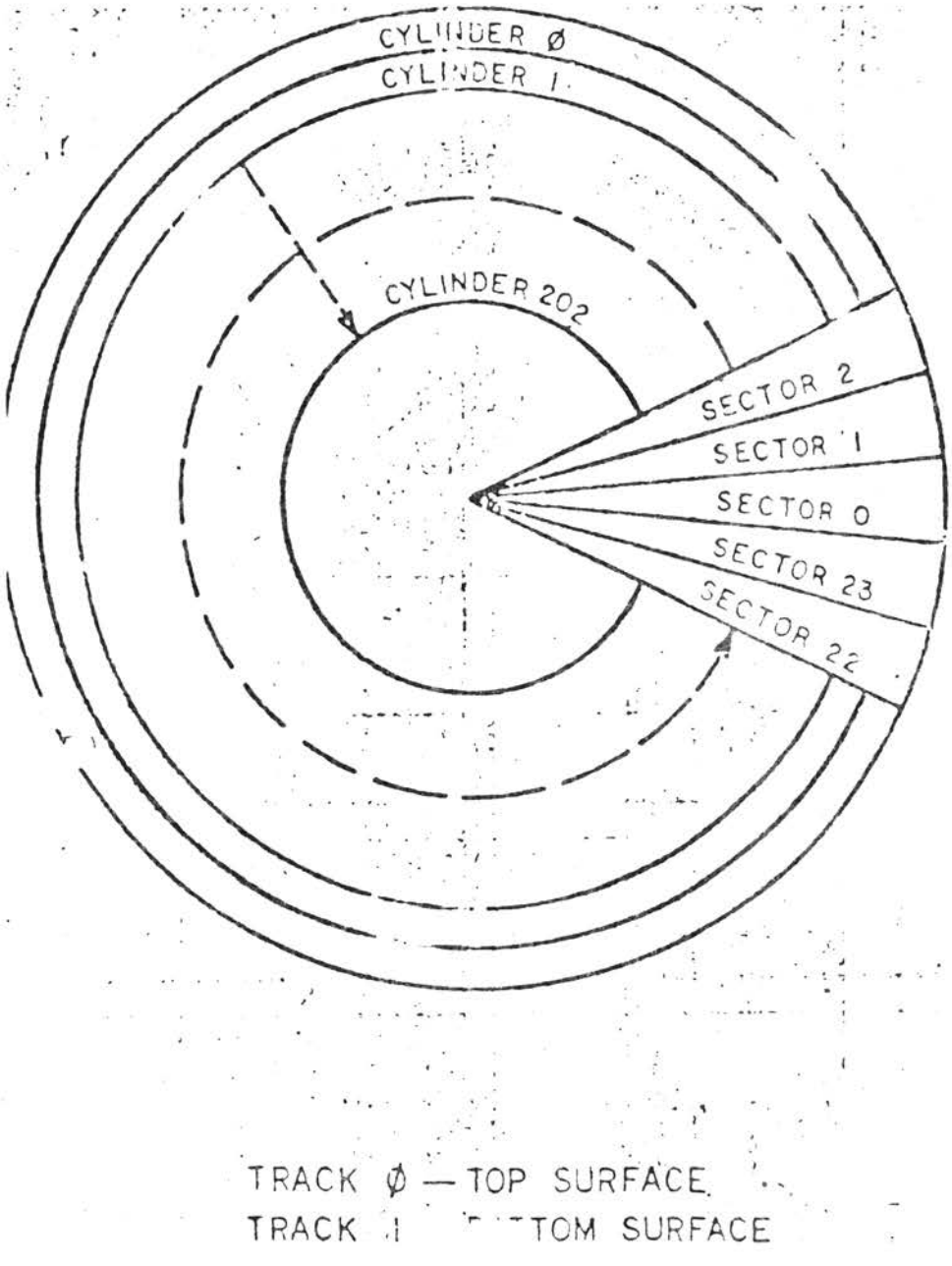
รูปที่ 2.3 แสดงรหัสมาตรฐานที่ได้จากแป้นสำหรับรับข้อมูล

2.1.4 เครื่องอ่านและบันทึกเทปคลับ(Cassette Tape Decks) สำหรับเครื่องคาดาพอยน์ 5500 จะมีเครื่องอ่านและบันทึกเทปคลับ 2 ตู้ เรียกตามตำแหน่งที่อยู่ว่าตู้หน้า(Front Deck) และตู้หลัง(Rear Deck) สามารถใช้ในการอ่านและบันทึกโปรแกรมหรือข้อมูลที่ใช้กับเครื่องคาดาพอยน์ได้ สาเหตุที่ต้องมีเครื่องอ่านและบันทึกเทปคลับประจำอยู่กับเครื่องเป็นเพราะในการเริ่มต้นใช้เครื่องทุกครั้ง โปรเซสเซอร์จะทำการอ่านโปรแกรมเริ่มต้นจากเทปคลับตู้หลังเสมอ โปรแกรมเริ่มต้นนี้เรียกว่าบูทสแตร์ป(Bootstrap) เมื่อถูกอ่านเข้าไปเก็บในหน่วยความจำแล้วจะทำหน้าที่นำเอาแฟ้มข้อมูล SYSTEM0/SYS เข้าไปไว้ในหน่วยความจำหลักเพื่อใช้งานต่อไป แอดเดรสของเครื่องอ่านและบันทึกเทปคลับได้แก่ 0360 ฐานแปด

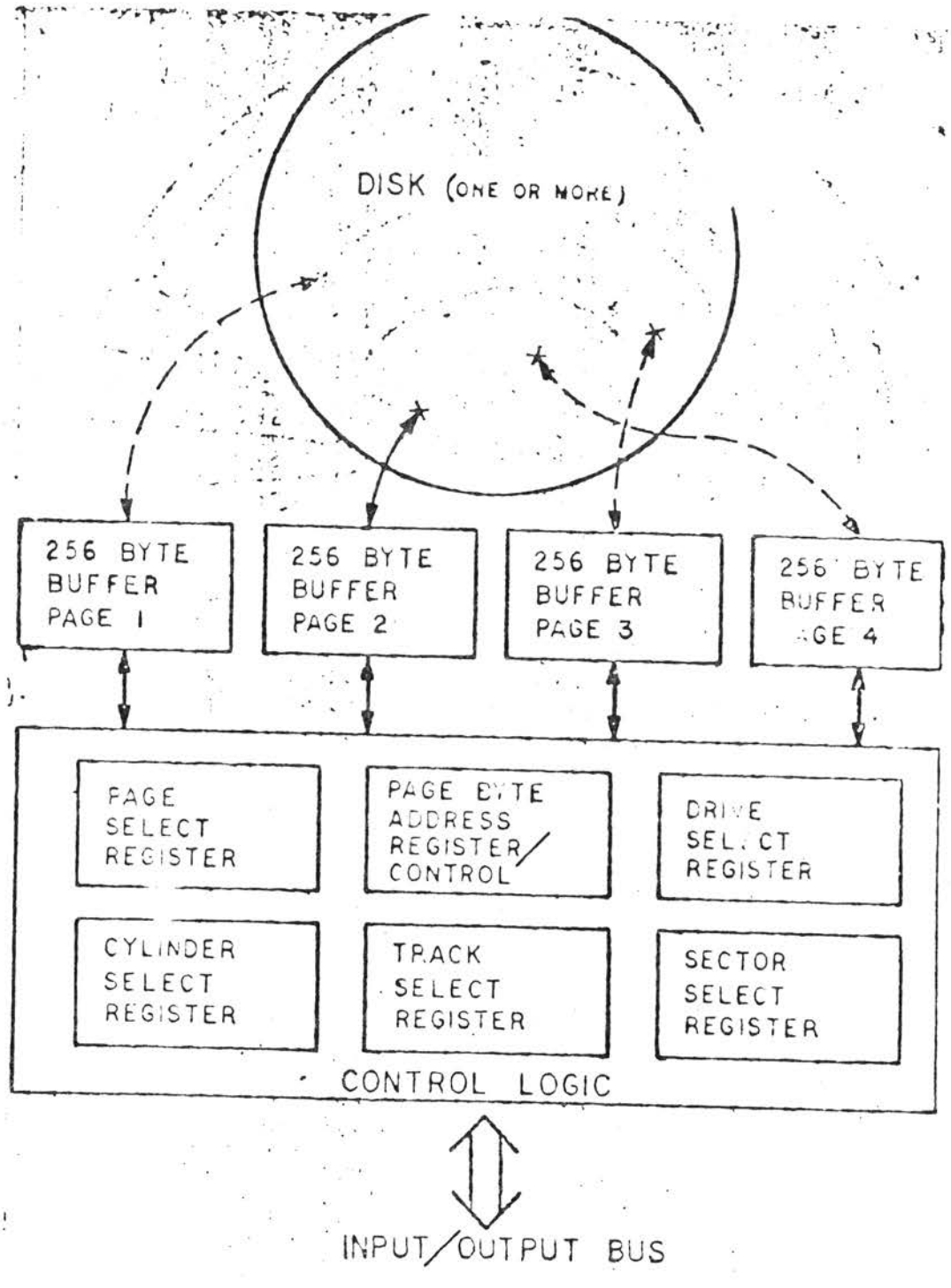
2.1.5 เครื่องอ่านและบันทึกจานแม่เหล็ก(Disk Drive) สามารถใช้ในการอ่านและบันทึกโปรแกรมหรือข้อมูลที่ใช้กับเครื่องคาดาพอยน์ได้ เครื่องอ่านและบันทึกจานแม่เหล็กที่ใช้กับเครื่องคาดาพอยน์ 5500 นี้ เป็นรุ่น 9350 ซึ่งได้ออกแบบมาเพื่อใช้กับจานแม่เหล็กที่เป็นแบบคาร์ทริด(Cartridge) จานแม่เหล็กที่ใช้มีลักษณะเป็นแบบ Fixed และ Removable จานแม่เหล็ก 1 แผ่น จะใช้บันทึกข้อมูลได้ 2 หน้า(Track) แบ่งออกเป็น 203 ไซลินเดอร์(Cylinders) มีจำนวนเซกเตอร์ต่อแทรค(Track) เท่ากับ 24 เซกเตอร์ มีจำนวนเซกเตอร์ต่อไดรว์(Drive) เท่ากับ 9,744 เซกเตอร์ มีจำนวนไบท์ต่อเซกเตอร์เท่ากับ 256 ไบท์ มีจำนวนไบท์ต่อไดรว์เท่ากับ 2,494,464 ไบท์ มีจำนวนเซกเตอร์ต่อ 1 คลัสเตอร์(Cluster) เท่ากับ 6 เซกเตอร์ มีจำนวนคลัสเตอร์ต่อแทรคเท่ากับ 4 คลัสเตอร์ มีจำนวนคลัสเตอร์ต่อ 1 ไซลินเดอร์ เท่ากับ 8 คลัสเตอร์ จำนวนเซกเตอร์ที่มากที่สุดที่แฟ้มข้อมูล(file) หนึ่ง ๆ จะมีได้เท่ากับ 9,600 เซกเตอร์ โปรเซสเซอร์กำหนดแอดเดรสแก่เครื่องอ่านและบันทึกจานแม่เหล็กเป็น 0170 ฐานแปด

2.1.6 เครื่องอ่านและบันทึกเทปแม่เหล็ก(Tape Drive) สามารถใช้ในการอ่านและบันทึกข้อมูลที่ใช้กับเครื่องคาดาพอยน์ได้ เครื่องอ่านและบันทึกเทปแม่เหล็กที่ใช้กับเครื่องคาดาพอยน์รุ่น 5500 นี้ เป็นรุ่น 9550 ซึ่งได้ออกแบบมาเพื่อใช้กับเทปแม่เหล็กที่มีขนาดกว้าง 1/2 นิ้ว การบันทึกและการอ่านข้อมูลจะเป็นแบบ 9 แทรค ความหนาแน่น(Density)

เป็น 1600 BPI (Bits per Inch) โปรแกรมเซอร์กำหนดแอดเดรสให้แก่เครื่องอ่านและบันทึกเทปแม่เหล็กเป็น 0264 ฐานแปด



รูปที่ 2.4 แสดงโครงสร้างของจานแม่เหล็ก



รูปที่ 2.5 แสดงโครงสร้างของส่วนควบคุมจานแม่เหล็ก

2.1.7 เครื่องที่ต่อจากภายนอกอื่น ๆ (Other External Peripherals) เครื่องดาตาพอยน์สามารถที่จะทำการต่อกับอุปกรณ์อื่น ๆ ได้มากมาย เช่น เอ็ม พี ซี เอ (MPCA = Multi Part Communication Adaptor) เครื่องพิมพ์ (Printer) เป็นต้น รายละเอียดสามารถดูได้จากหนังสือ Datapoint Equipment Catalog Model Code 60001

ชื่ออุปกรณ์	เลขที่	แอดเดรส(ฐานแปด)
เครื่องอ่านและบันทึกเทปคลัง	0	360
จอภาพแสดงผลและแป้นรับข้อมูล (ของโปรเซสเซอร์)	1	341
Asynchronous Communication Adaptors	2	322
เครื่องพิมพ์ประเภทโลคัล (Local)	3	303
เครื่องอ่านและบันทึกเทปแม่เหล็ก	4	264
Synchronous Communication Adaptors	5	245
Parallel Communication Adaptors	6	226
เครื่องอ่านบัตร	7	207
เครื่องควบคุมดิสก์รุ่น 9350	8	170
เอ็ม พี ซี แอ	9	151
เครื่องพิมพ์ (ฉาหน้าการเพิ่ม)	10	132
เครื่องควบคุมดิสก์รุ่น 9370	11	113
เครื่องควบคุมดิสก์ขนาดเล็ก (Diskkette)	12	074
ไมระบุ	13	055
ไมระบุ	14	036
ไมระบุ	15	017

รูปที่ 2.6 แสดงแอดเดรสที่โปรเซสเซอร์กำหนดให้กับอุปกรณ์

2.2 ระบบโครงสร้าง(System Structure)

2.2.1 โครงสร้างของจานแม่เหล็ก(Disk Structure)

จานแม่เหล็กที่นำมาใช้กับเครื่องคาดคาพอยท์ จะต้องมีรายละเอียด(Information) ต่าง ๆ บรรจุอยู่บนจานแม่เหล็กชุดนั้น ซึ่งมีความจำเป็นต่อ ดี โอ เอส (Disk Operating System) โครงสร้างพื้นฐานของจานแม่เหล็กคือ แฟ้มข้อมูลหรือไฟล์ (File) แฟ้มข้อมูลแต่ละแฟ้มจะประกอบด้วยจำนวนเซกเตอร์(Sector) ได้มากที่สุดไม่เกิน 38,000 เซกเตอร์ หรือไม่เกินจำนวนของเซกเตอร์ซึ่งจานแม่เหล็กแต่ละชุดได้กำหนดไว้ ซึ่งแล้วแต่ว่าจำนวนใดจะน้อยกว่าเนื้อที่ในจานแม่เหล็กที่แฟ้มข้อมูลแต่ละแฟ้มครอบครองอยู่จะถูกบันทึกเก็บไว้ในเซกเตอร์แรกของแต่ละแฟ้มข้อมูล ซึ่งเรียกเซกเตอร์นี้ว่า รีทริวัล อินฟอร์เมชั่น บลอก (Retrieval Information Block) หรือริบ(RIB) การจัดสรรเนื้อที่ในจานแม่เหล็กให้กับแฟ้มข้อมูลจะเป็นไปในลักษณะที่เรียกว่าคลัสเตอร์(Cluster) คลัสเตอร์คือ หน่วยของเนื้อที่ที่เล็กที่สุดซึ่งสามารถจัดสรรได้สำหรับจานแม่เหล็กใน 1 ไซลินเดอร์ จะแบ่งออกเป็น 8 คลัสเตอร์เท่า ๆ กัน และใน 1 คลัสเตอร์จะแบ่งออกเป็น 6 เซกเตอร์ เซกเตอร์ซึ่งประกอบกันเป็น 1 คลัสเตอร์นั้นจะต้องอยู่ติดต่อกันเป็นผืนเดียวตลอด(Contiguous) และต้องอยู่ภายในไซลินเดอร์เดียวกัน คลัสเตอร์ เอลโลเคชั่น เทเบิล(Cluster Allocation Table) หรือแคท(CAT) และลอกเอาท์ แคท(Lockout CAT) จะใช้ในการบันทึกคลัสเตอร์ที่ถูกใช้และคลัสเตอร์ที่ยังว่างอยู่ RIB จะทำการแมป(Map) เนื้อที่ของแฟ้มข้อมูลในหน่วยของเซกเมนต์ (Segment) เซกเมนต์คือชุด(Set) ของคลัสเตอร์ที่ติดต่อกัน(Contiguous Clusters) ดังนั้นแฟ้มข้อมูลจึงประกอบขึ้นด้วยชุดของเซกเมนต์ซึ่งกระจายอยู่ทั่วไปในจานแม่เหล็ก แต่ละเซกเตอร์ซึ่งประกอบขึ้นเป็นแฟ้มข้อมูลจะมีลักษณะที่เหมือนกันคือ ไบท์แรกของเซกเตอร์ใช้เก็บค่าของฟิสิกส์คัล ไฟล์ นัมเบอร์(Physical File Number) หรือ พี เอฟ เอ็น(PFN) ซึ่งครอบครองเซกเตอร์นี้อยู่ พี เอฟ เอ็น จะมีค่าเริ่มจาก 000 ฐานแปด จนถึง 0377 ฐานแปด ไบท์ที่สองและสามจะใช้เก็บค่าของ โลจิกคัล เรคคอร์ด นัมเบอร์(Logical Record Number) หรือ แอล อาร์ เอ็น (LRN) แอล อาร์ เอ็น คือหมายเลขลำดับที่ของเซกเตอร์ในแต่ละแฟ้มข้อมูล มีค่าเริ่มต้นเป็น 0 และเพิ่มขึ้นทีละ 1 เมื่อมีเซกเตอร์ใหม่เพิ่มขึ้นในแฟ้มข้อมูล

009142

2.2.1.1 คลัสเตอร์ แอลโลเคชั่น เทเบิล และ ล็อคเอาท์คลัสเตอร์ แอลโลเคชั่น เทเบิล(Cluster Allocation Table & Lockout CAT)

แคท(CAT) มีหน้าที่ควบคุมเนื้อที่ในจานแม่เหล็กที่ ดี โอ เอส สามารถนำไปใช้งานได้ แคทจะถูกแก้ไขโดยอัตโนมัติ เมื่อมีการจัดสรรเนื้อที่ในจานแม่เหล็ก หรือเมื่อมีการเลิกใช้เนื้อที่ในจานแม่เหล็ก(Space Allocation or Deallocation) แต่ละไบท์ของแคทจะหมายถึงไซลินเดอร์ต่าง ๆ เช่น ไบท์ 0 หมายถึง ไซลินเดอร์ 0 ไบท์ 1 หมายถึงไซลินเดอร์ 1 เป็นต้น แต่ละบิทของแต่ละไบท์จะหมายถึงคลัสเตอร์ต่าง ๆ ของไซลินเดอร์หนึ่ง ๆ เช่น บิท 7 หมายถึง คลัสเตอร์ 0 บิท 6 หมายถึง คลัสเตอร์ 1 บิท 5 หมายถึง คลัสเตอร์ 2 เป็นต้น ถ้าบิทใดมีค่าเป็น 0 แสดงว่าคลัสเตอร์นั้นเป็นคลัสเตอร์ที่ยังไม่ได้ถูกนำไปใช้งาน ถ้าบิทใดมีค่าเป็น 1 แสดงว่าคลัสเตอร์นั้นเป็นคลัสเตอร์ที่ถูกใช้งานไปแล้ว หรืออาจถูกล็อคเอาท์(Lockout) ล็อคเอาท์ แคท มีหน้าที่ควบคุมไซลินเดอร์ที่ ดี โอ เอส ไม่สามารถนำไปใช้งานได้ ไซลินเดอร์จะถูกล็อคเอาท์โดยอัตโนมัติในตอนทำ ดี โอ เอส เยนเนอเรชั่น(DOS Generation) ถ้าหากพบว่าไซลินเดอร์เหล่านั้นเสียไม่สามารถที่จะนำไปใช้งานได้ นอกจากนี้ผู้ใช้ยังสามารถที่จะทำการล็อคเอาท์ไซลินเดอร์ที่ต้องการในระหว่างการทำซิสเตม เยนเนอเรชั่น(System Generation) หรือในระหว่างการใช้คำสั่ง REPAIR ไซลินเดอร์หมายเลข 0 จะถูกล็อคเอาท์สำหรับให้ระบบใช้ แต่ละไบท์ของล็อคเอาท์แคท ใช้แทน 1 ไซลินเดอร์ ไบท์ที่ 0 ใช้อ้างอิงถึงไซลินเดอร์หมายเลข 0 ไบท์ที่ 1 ใช้อ้างอิงถึงไซลินเดอร์หมายเลข 1 ไบท์ที่ 2 ใช้อ้างอิงถึงไซลินเดอร์หมายเลข 2 เป็นต้น ถ้าไซลินเดอร์ใดถูกล็อคเอาท์ ค่าของไบท์ที่ใช้อ้างอิงถึงไซลินเดอร์นั้นจะเป็น 0377 ฐานแปด และจะมีค่าเป็น 0 ถ้าหากไซลินเดอร์นั้นไม่ถูกล็อคเอาท์

รูปแบบของแคทและล็อคเอาท์แคท จะเป็นไปตามกฎเกณฑ์ดังต่อไปนี้

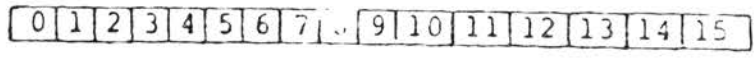
ไบท์ 0 มีค่าเป็น 0377 ฐานแปด เสมอ เหมาะเป็นไซลินเดอร์ที่ระบบใช้

ไบท์ 1 ถึง ไบท์ n มีค่าเป็นอะไรก็ได้ตามที่กล่าวมาข้างต้น(n หมายถึงจำนวนของไซลินเดอร์บนจานแม่เหล็ก)

ไบท์ n+1 ถึง ไบท์ 0376 ฐานแปด มีค่าเป็น 0377 ฐานแปด (ยกเว้นไดเรคทอรี แมปปิ้ง ไบท์(Directory Mapping Bytes)ถ้าหากจะใช้)

ไบท์ 0377 ฐานแปด มีค่าเป็นอะไรก็ได้ ปกติจะเป็น 0

2.2.1.2 สารบัญ(Directory) สารบัญของแต่ละแฟ้มข้อมูลมีขนาด 16 ไบท์ เก็บอยู่ในจานแม่เหล็ก โดยใช้เนื้อที่ในจานแม่เหล็กทั้งหมด 16 เซกเตอร์ อ้างอิงถึงได้โดย เซกเตอร์ 0 - 15 แต่ละเซกเตอร์จะเก็บสารบัญได้ 16 รายการ แต่ละรายการจะเก็บชื่อของแฟ้มข้อมูล ลักษณะการป้องกันแฟ้มข้อมูล(Protection) สารบัญย่อยของแฟ้มข้อมูล(Subdirectory) และตำแหน่งของ RIB ของแฟ้มข้อมูล รูปที่ 2.7 แสดงรูปแบบของสารบัญ



รูปที่ 2.7 แสดงรูปแบบของสารบัญ (Directory)

ไบท์ที่ 0 - 1 ใช้บันทึกตำแหน่งของ RIB และลักษณะการป้องกันแฟ้มข้อมูล รายละเอียดจะได้กล่าวถึงภายหลัง

ไบท์ที่ 2 - 3 ไม่ใช้เป็น 0 เสมอ

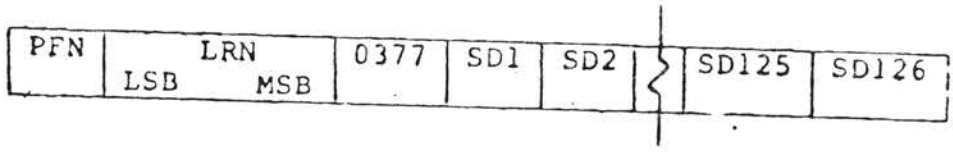
ไบท์ที่ 4 - 11 ใช้บันทึกชื่อแฟ้มข้อมูล จะเติมที่ว่าง(Space) ให้ ถ้าไม่ครบ 8 ตัว

ไบท์ที่ 12 - 14 ใช้บันทึกส่วนขยายชื่อแฟ้มข้อมูล(File Extension) จะเติมที่ว่างให้ ถ้าไม่ครบ 3 ตัว

ไบท์ที่ 15 ใช้บันทึกหมายเลขของสารบัญย่อย ปกติจะเป็น 0377 ฐานแปด ซึ่งเป็นหมายเลขของสารบัญย่อย System

2.2.1.3 แฮช ไดเรคทอรี อินเดกซ์(Hashed Directory Index) ใช้ช่วยในการเข้าถึง(Access) และควบคุมการจัดสรรสารบัญ(Directory) แต่ละไบท์ของ เอช ดี ไอ (HDI) จะใช้แทนสารบัญแต่ละรายการ โดยไบท์ที่ 0 จะแทน พี เอฟ เอ็น(PFN) 0 ไบท์ ที่ 1 จะแทน พี เอฟ เอ็น 1 ไบท์ ที่ 2 จะแทน พี เอฟ เอ็น 2 เป็นต้น ค่าของไบท์ใดมีค่าเป็น 0377 ฐานแปด แสดงว่าสารบัญที่ไบท์นั้นเป็นตัวแทนยังไม่ได้ถูกใช้ เมื่อ พี เอฟ เอ็น ถูกใช้รหัสที่เรียกว่า แฮช(Hash Code) ซึ่งมีค่าอยู่ระหว่าง 0 ถึง 0376 ฐานแปด ที่ถูกสร้าง(Generate) ขึ้นจากชื่อของแฟ้มข้อมูลจะถูกนำมาเก็บไว้ในไบท์ซึ่งเป็นตัวแทนค่าของรหัสแฮชจะใช้แสดงว่า พี เอฟ เอ็น ได้ถูกใช้และช่วยเพิ่มความเร็วในการค้นหาสารบัญ เมื่อแฟ้มข้อมูลถูกเรียกใช้โดยชื่อของแฟ้มข้อมูล

2.2.1.4 รีทริวัล อินฟอร์เมชั่น บล็อก(Retreive Information Block) หรือริบ(RIB) ริบทำหน้าที่แบ่งขนาดของแฟ้มข้อมูลหนึ่ง ๆ ออกเป็นเซกเมนต์(Segment) ต่าง ๆ แต่ละเซกเมนต์จะประกอบขึ้นจากคลัสเตอร์ต่าง ๆ ที่ต่อเนื่องกัน(Contiguous Clusters) แต่ละริบจะสามารถบรรจุเซกเมนต์ ดิสคริบเตอร์(Segment Descripter) หรือ เอส ดี (SD) ได้สูงสุด 126 ชุด ซึ่งเพียงพอที่จะใช้ในการจัดสรรคลัสเตอร์ให้แก่แฟ้มข้อมูลหนึ่ง ๆ ริบมีรูปแบบดังนี้ แสดงในรูปที่ 2.8



รูปที่ 2.8 แสดงรูปแบบของ อาร์ ไอ บี (RIB)

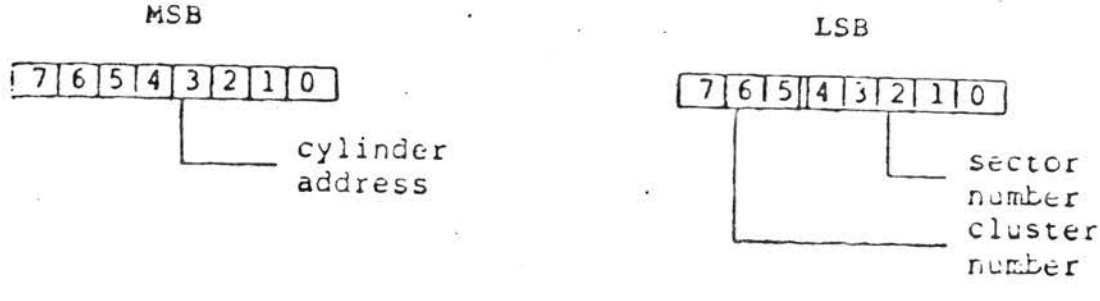
เอส ดี แต่ละรุ่นมีขนาด 2 ไบท์ เอส ดี ที่มีค่าเป็น 0377, 0377 ฐานแปด แสดงถึงรุ่นสิ้นสุดของริบ ไบท์ที่ 4 ของริบมีค่าเป็น 0377 ฐานแปด เสมอ ริบจะถูกบันทึกลงบนเซกเตอร์ที่หนึ่งของแพ็มข้อมูล และริบจะถูกบันทึกลงในเซกเตอร์ที่สอง เพื่อใช้เป็นตัวสำรอง(Backup) ในกรณีที่เซกเตอร์ที่หนึ่งของแพ็มข้อมูลถูกทำให้เสียหาย ริบทั้งคู่มีค่าเหมือนกันทุกประการ ยกเว้นค่าของ แอล อาร์ เอ็น ของริบ ตัวที่สอง(RIB Copy) จะมีค่าเป็น 1

2.2.1.5 การอ้างอิงเซกเตอร์(Sector Identification)

ไบท์ที่ 1 ของทุก ๆ เซกเตอร์ของแพ็มข้อมูลจะใช้สำหรับเก็บค่าของ พี เอฟ เอ็น ไบท์ที่ 2 และ 3 จะใช้สำหรับเก็บค่าของ แอล อาร์ เอ็น(ดูรูปที่ 2.8) เมื่อมีการข้อมูลจากแพ็มข้อมูล ค่าของ พี เอฟ เอ็น และ แอล อาร์ เอ็น จะถูกทำการตรวจสอบความถูกต้อง ถ้าไม่ถูกต้อง แสดงว่าเกิดการผิดพลาดทางรูปแบบ(Record Format Error) ขึ้น เมื่อมีการบันทึกข้อมูล ค่าของ พี เอฟ เอ็น จะถูกบันทึกลงในไบท์ที่ 1 ของเซกเตอร์ ส่วนค่าของ แอล อาร์ เอ็น จะถูกบันทึกลงในไบท์ที่ 2 และ 3 ของเซกเตอร์ เซกเตอร์ที่ยังไม่ได้ถูกใช้(Unused) 3 ไบท์แรกของเซกเตอร์จะมีค่าเป็นอะไรก็ได้

2.2.1.6 โครงสร้างของไบท์ที่ใช้ในการกำหนดแอดเดรส Addressing Byte Structures)

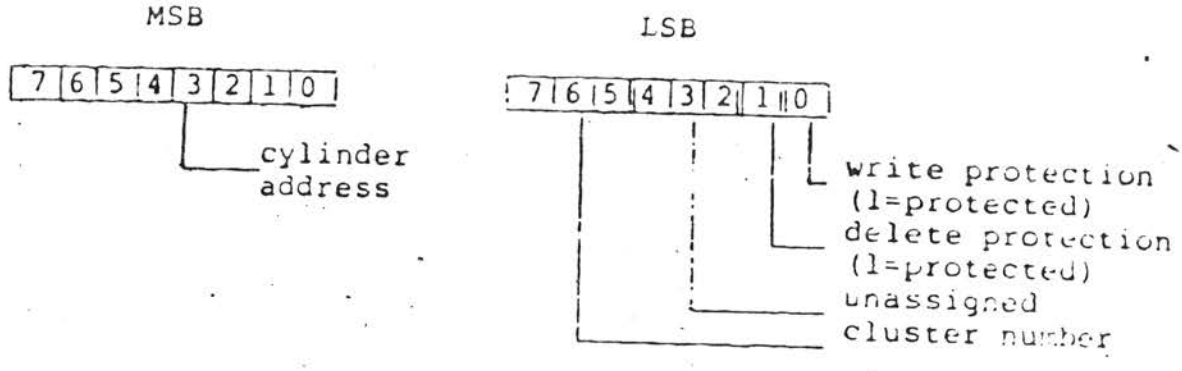
2.2.1.6.1 ฟิสิกส์คัล ดิส แอดเดรส(Physical Disk Address) หรือ พี ดี เอ (PDA) ประกอบด้วยจำนวนไบท์ 2 ไบท์ มีลักษณะดังรูปที่ 2.9



รูปที่ 2.9 แสดงรูปแบบของฟิสิกส์คัล ดิส แอดเดรส

คลัสเตอร์ นัมเบอร์ จะอ้างอิงถึงแต่ละคลัสเตอร์ของไดรีนเตอร์หนึ่ง ๆ ซึ่งคลัสเตอร์ นัมเบอร์นี้จะมีค่าอยู่ระหว่าง 0 - 7 ส่วนเซกเตอร์ นัมเบอร์ จะอ้างอิงถึงแต่ละเซกเตอร์ของคลัสเตอร์หนึ่ง ๆ

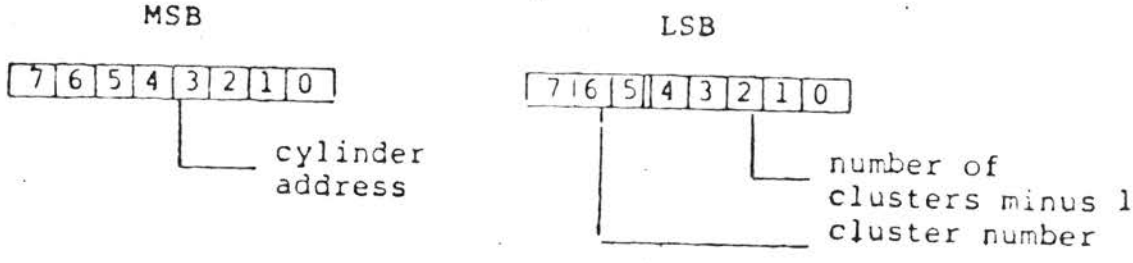
2.2.1.6.2 ริบ แอดเดรส/โปรเทคชั่น(RIB Address/Protecing) เป็นพอยน์เตอร์(Pointer) ที่เก็บอยู่ในสารบัญ(Directory) ที่จะชี้ไปยังจุดเริ่มต้นของแฟ้มข้อมูล ริบ แอดเดรส/โปรเทคชั่น มีขนาด 2 ไบท์ แยกเป็น MSB และ LSB ดังรูปที่ 2.10



รูปที่ 2.10 แสดงริบแอดเดรส/โปรเทคชั่น

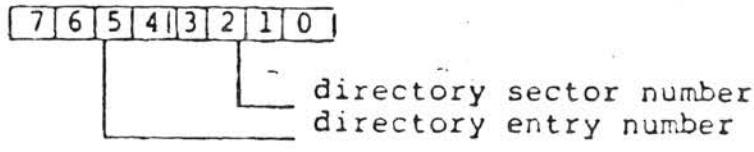
ริบ แอดเดรส/โปรเทคชั่น จะบ่งชี้แอดเดรสของไซลินเดอร์และหมายเลขของคลัสเตอร์ที่เป็นจุดเริ่มต้นของแฟ้มข้อมูล โดยถือว่าเซกเตอร์ที่ 0 เป็นเซกเตอร์แรกของแฟ้มข้อมูลนั้น ๆ

2.2.1.6.3 เซกเมนต์ ดิสคริปเตอร์ (Segment Descriptor) หรือ เอส ดี (SD) ถูกใช้ในริบ (RIB) เพื่อบอกแอดเดรสของแต่ละเซกเมนต์ ประกอบขึ้นด้วยไบท์จำนวน 2 ไบท์ มีลักษณะดังรูปที่ 2.11



รูปที่ 2.11 แสดงรูปแบบของเซกเมนต์ ดิสคริปเตอร์

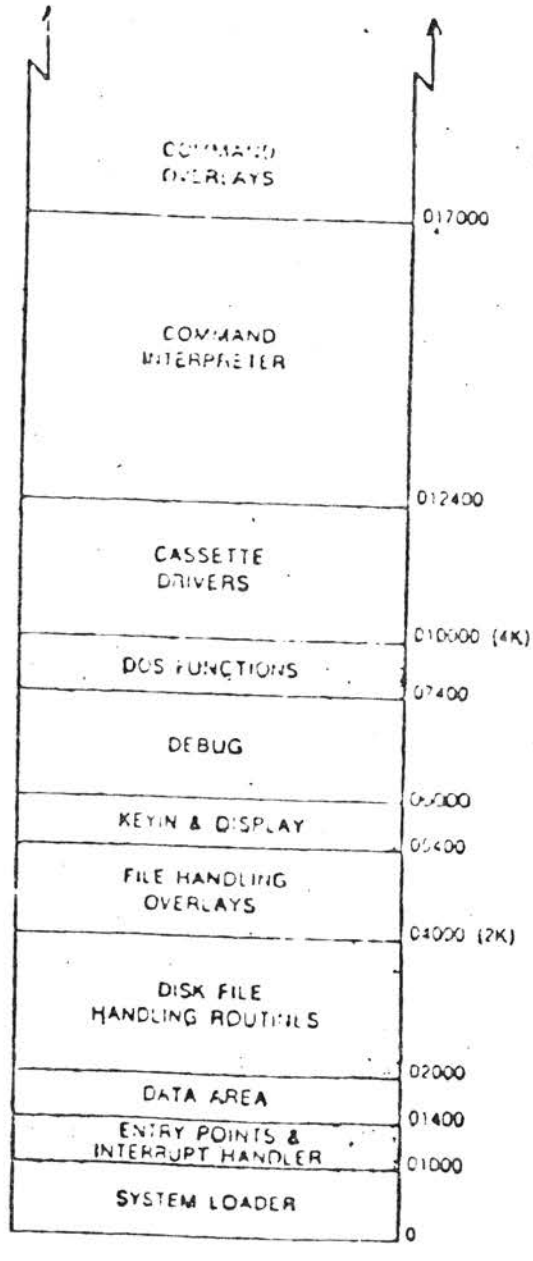
2.2.1.6.4 พิธีกร์คัล ไฟล์ นัมเบอร์(Physical File Number) หรือ พี เอฟ เอ็น (PFN) ใช้ในการเข้าถึงสารบัญ(Directory) และ เอช ดี ไอ(HDI) มีรูปแบบดังรูปที่ 2.12



รูปที่ 2.12 แสดงพิธีกร์คัล ไฟล์ นัมเบอร์(PFN)

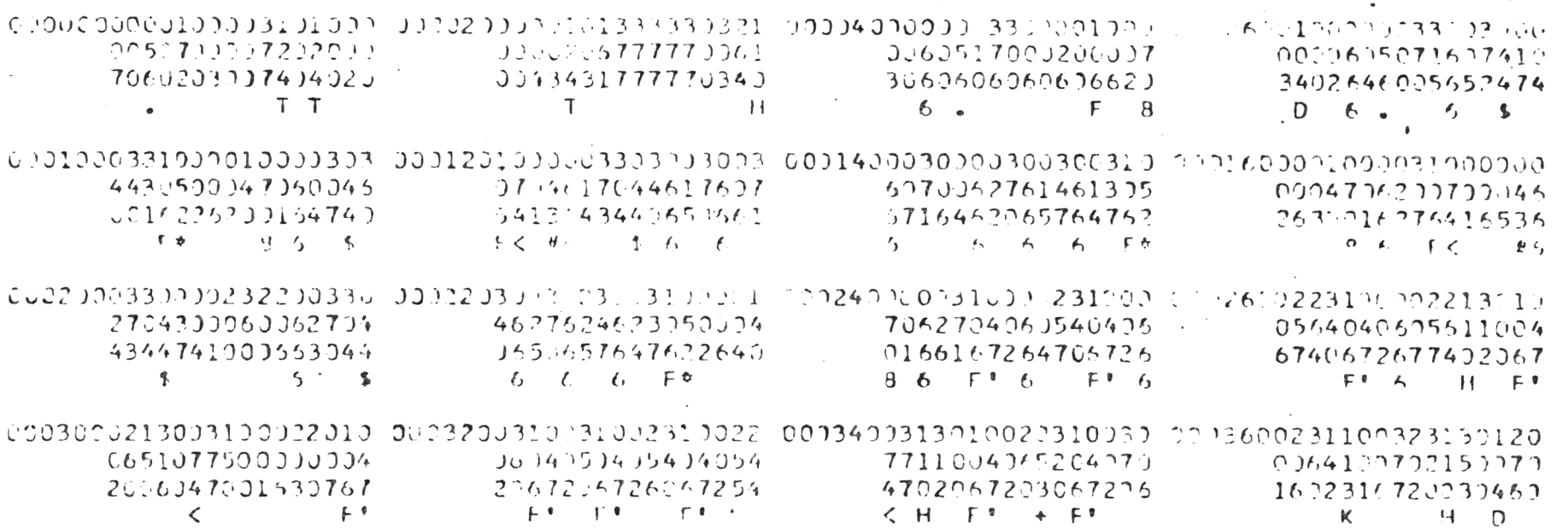
โดยไดเรคทอรี เซกเตอร์ นัมเบอร์ จะเป็นตัวกำหนดหมายเลขของเซกเตอร์ที่เป็นสารบัญ และไดเรคทอรี เอนทรี นัมเบอร์ จะเป็นตัวกำหนดหมายเลขของรายการ(Entry) ที่อยู่บนเซกเตอร์นั้น

2.2.2 การแบ่งหน่วยความจำ(Memory Mapping) ดี ไอ เอช ได้ทำการแบ่งหน่วยความจำออกเป็นส่วน ๆ มีลักษณะดังรูปที่ 2.13

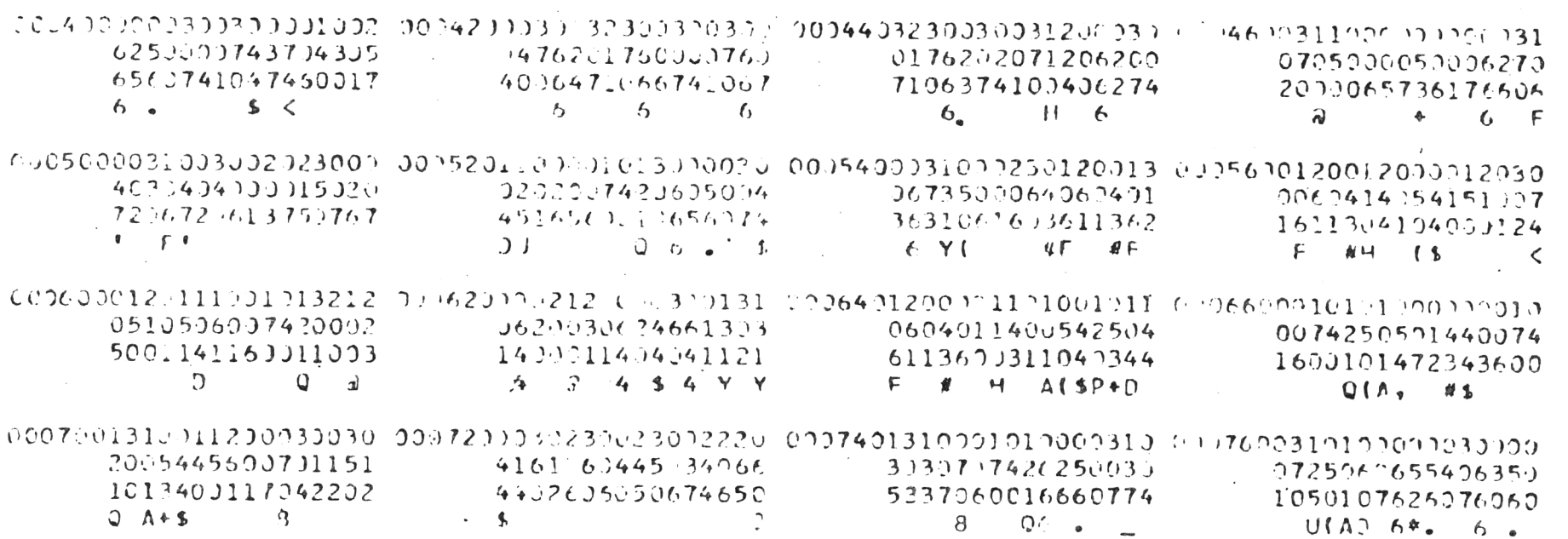


รูปที่ 2.13 แสดงการแบ่งหน่วยความจำ

2.2.2.1 ซิสเตม โลดเดอร์ (System Loader) ถูกนำเข้าไปไว้ในหน่วยความจำหลักตั้งแต่แอดเดรสที่ 0 ถึง 0777 ฐานแปด เพื่อทำหน้าที่นำเอาโปรแกรมอื่นที่ต้องการใช้เข้าไปเก็บไว้ในหน่วยความจำของเครื่อง รายละเอียดของซิสเตม โลดเดอร์ ซึ่งเป็นภาษาเครื่องแสดงไว้ในรูปที่ 2.14

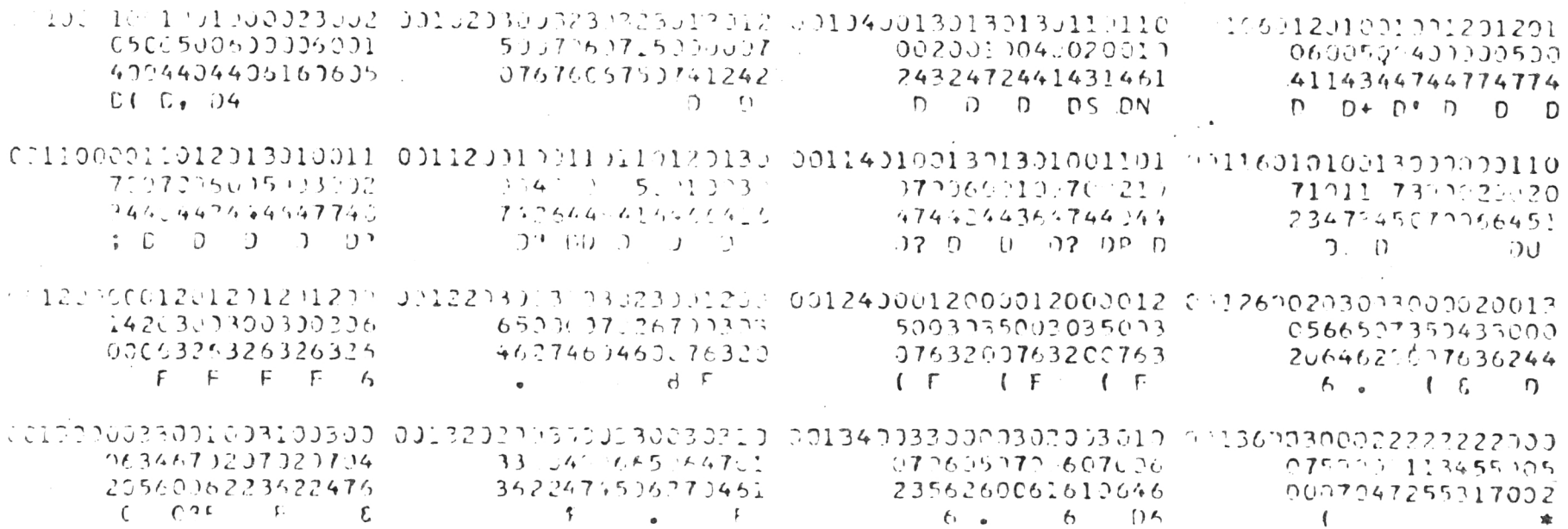


รูปที่ 2.14 แสดงซิสเทม โลดเดอร์



รูปที่ 2.14 (ต่อ) แสดงซิสเทม โอดเดอร์

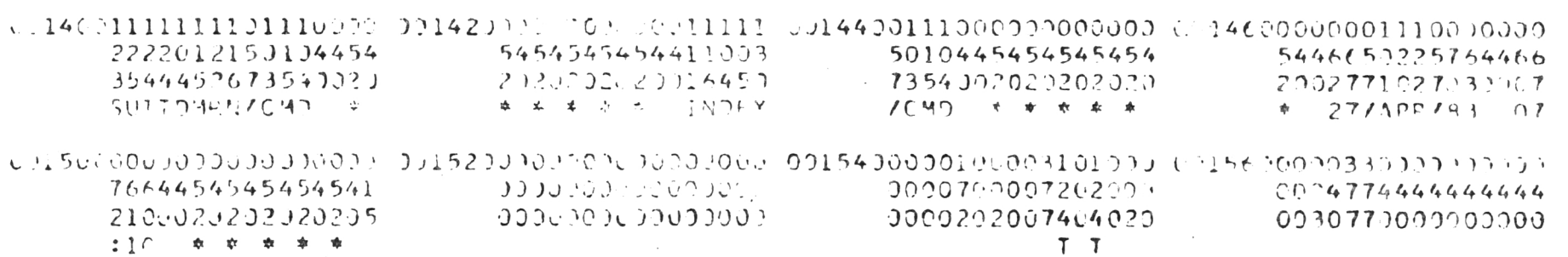
2.2.2.2 เอนทริพอยท์และอินเตอร์รัพท์ แฮนด์เลอร์ (Entry Point & Interrupt Handler) ประกอบด้วยคำสั่งที่ให้ข้ามไปยังส่วนของโปรแกรมย่อยที่ใช้ในการทำงาน ส่วนพื้นฐานของเครื่อง เช่น โปรแกรมที่จำเป็นสำหรับการอ่านและบันทึกจานแม่เหล็ก โปรแกรมที่ใช้ในการควบคุมการจัดแฟ้มข้อมูลในเครื่องอ่านและบันทึกจานแม่เหล็ก รวมทั้งโปรแกรมส่วนที่เกี่ยวข้องกับการทำงานของรีจิสเตอร์ (Registers) ของเครื่องเองเหล่านี้เป็นต้น เหตุที่ต้องใช้หน่วยความจำตั้งแต่ แอดเดรสที่ 01000 ฐานแปด ถึง 01377 ฐานแปด ในการเก็บเฉพาะคำสั่งเพียงคำสั่งเดียว สำหรับโปรแกรมแต่ละส่วนซึ่งอยู่ในหน่วยความจำส่วนอื่นนั้นก็เพื่อความสะดวกในการแก้ไขโปรแกรมที่กล่าวไว้ข้างต้น โดยผู้ใช้ไม่ต้องแก้ไขโปรแกรมที่มีอยู่แล้วแต่อย่างไร รายละเอียดของเอนทริพอยท์ และอินเตอร์รัพท์ แฮนด์เลอร์ ได้แสดงไว้ในรูปที่ 2.15



รูปที่ 2.15 แสดงส่วนของเอนทริพอยต์และอินเตอร์รัพต์แฮนด์เลอร์

2.2.2.3 ดาตาเอเรีย(Data Area)

2.2.2.3.1 มอนิเตอร์ คอมมิวนิเคชั่น รีเจียน (Monitor Communication Region) หรือ MCRs ใช้ที่ในหน่วยความจำตั้งแต่แอดเดรสที่ 01400 ฐานแปดถึง 01543 ฐานแปด เพื่อใช้ในการเก็บข้อมูลที่ได้รับจากแป้นสำหรับรับข้อมูลของตัวโปรเซสเซอร์ มีรายละเอียดดังรูปที่ 2.16



รูปที่ 2.16 แสดงโมนิเตอร์ คอมพิวนิเคชั่น รีเจียน .

2.2.2.3.2 โลจิกคัล ไฟล์ เทเบิล(Logical File Table) หรือ แอล เอฟ ที (LFT) ใช้สำหรับเก็บรายละเอียดต่าง ๆ ของแฟ้มข้อมูลที่จำเป็นสำหรับ ดิส ไฟล์ แฮนด์ลิ่ง รูทีน(Disk File Handling Routines) ใช้เนื้อที่ในหน่วยความจำ ตั้งแต่แอดเดรสที่ 01544 ฐานแปดถึง 01643 ฐานแปด ผู้ใช้สามารถที่จะเปิด(open) แฟ้ม ข้อมูลได้มากที่สุด 3 แฟ้มข้อมูลในเวลาเดียวกัน เมื่อผู้ใช้ทำการเปิดแฟ้มข้อมูล โดยใช้ชื่อแฟ้ม ข้อมูล โปรแกรมระบบจะทำการกำหนด โลจิกคัล ไฟล์ นัมเบอร์(Logical File Number) หรือ แอล เอฟ เอ็น (LFN) ให้ แอล เอฟ เอ็น จะกำหนดที่ใน แอล เอฟ ที และเพจ (Page) ของ ดิส เมมมอรี บัฟเฟอร์(Disk Memory Buffer) ที่แฟ้มข้อมูลนี้จะต้องใช้ ให้ แอล เอฟ ที มีอยู่ทั้งสิ้น 4 ชุด อ้างอิงได้โดย LFT0, LFT1, LFT2 และ LFT3 แต่ละ ชุดมีขนาด 16 ไบท์ โดยแต่ละไบท์จะเก็บข้อมูล ดังรูปที่ 2.17 (ตัวเลขที่อยู่ในวงเล็บเป็นจำ-นวนไบท์ที่ใช้)

LFT0 ใช้หน่วยความจำจากแอดเดรส 01544 ฐานแปดถึง 01563 ฐานแปด ถูกใช้โดย ดี โอ เอส

LFT1 ใช้หน่วยความจำจากแอดเดรส 01564 ฐานแปดถึง 01603 ฐานแปด

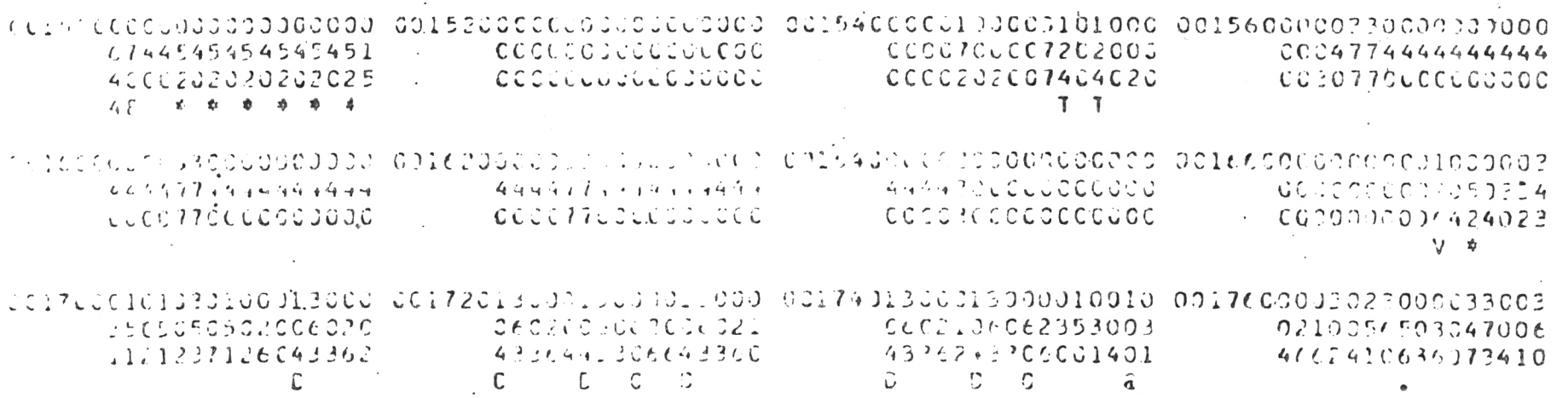
LFT2 ใช้หน่วยความจำจากแอดเดรส 01604 ฐานแปดถึง 01623 ฐานแปด

LFT3 ใช้หน่วยความจำจากแอดเดรส 01624 ฐานแปดถึง 01643 ฐานแปด

PFN	(1)	- Physical File Number, PFN of the file referenced by this LFT
PDN	(1)	- Logical Drive Number (bits 3 - 0) Protection (bit 7 set indicates delete protection, bit 6 set indicates write protection) New Space Allocated flag (bit 5) set if new space has been allocated to this file.
LRN	(2)	- Next Logical Record Number, system LRN of next sequential sector
BLRN	(2)	- Base LRN, first LRN in current segment (system LRN)
CSD	(2)	- Current Segment Descriptor The CSD and BLRN describe the current file segment and allow quick calculation of the PDA to be read/written by treating the LRN as an offset from BLRN. If the desired LRN is not in the current segment, the RIB is re-read and a new current segment established.
RIBCYL	(1)	- Physical Disk Address of RIB, MSB
RIBSEC	(1)	- Physical Disk Address of RIB, LSB Storing the RIB PDA allows quickly locating the RIB when it must be accessed for getting a new segment descriptor, for allocation updates, or for file closing.
MAXLRN	(2)	- Largest system LRN referenced (read, written, or positioned to) since the file was opened. Used for space deallocation at close if new space allocated flag is set.
LRNLIM	(2)	- Largest LRN allowed. Obsolete field, now unused.
BUFADR	(1)	- Current controller buffer byte address, used for byte transfers to or from the disk controller buffer.
XXXXXX	(1)	- Unused

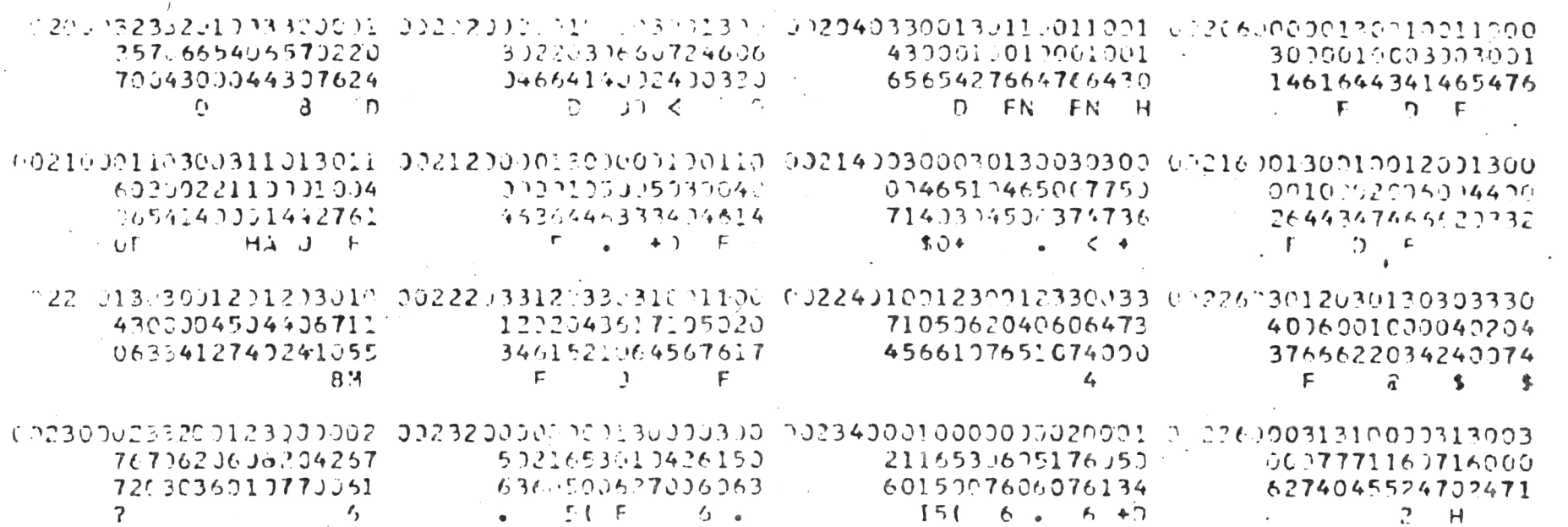
16 bytes total

รูปที่ 2.17 แสดงรายละเอียดของโลจิกคัล ไฟล์ เทเบิล



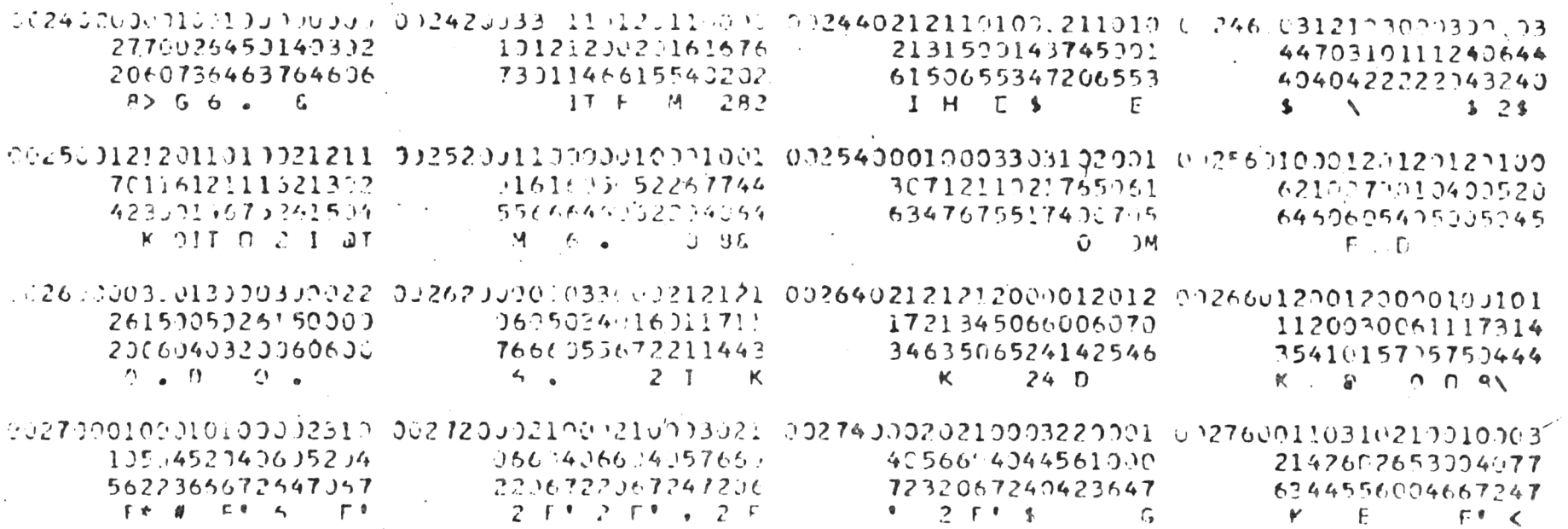
รูปที่ 2.18 แสดงส่วนของโอจิกคัล โพล์ เทเบิล

2.2.2.4 ดิส ไฟล์ แฮนด์ลิ่ง รูทีน(Disk File Handling Routines) ใช้ในการเก็บส่วนของโปรแกรม ซึ่งใช้ในการควบคุมการจัดแฟ้มข้อมูลในเครื่องอ่าน และบันทึกจานแม่เหล็ก ใช้เนื้อที่ในหน่วยความจำตั้งแต่ แอดเดรสที่ 02000 ฐานแปดถึง 03777 ฐานแปด รายละเอียดของหน่วยความจำที่ใช้เก็บ ดิส ไฟล์ แฮนด์ลิ่ง รูทีน แสดงไว้ในรูปที่ 2.19

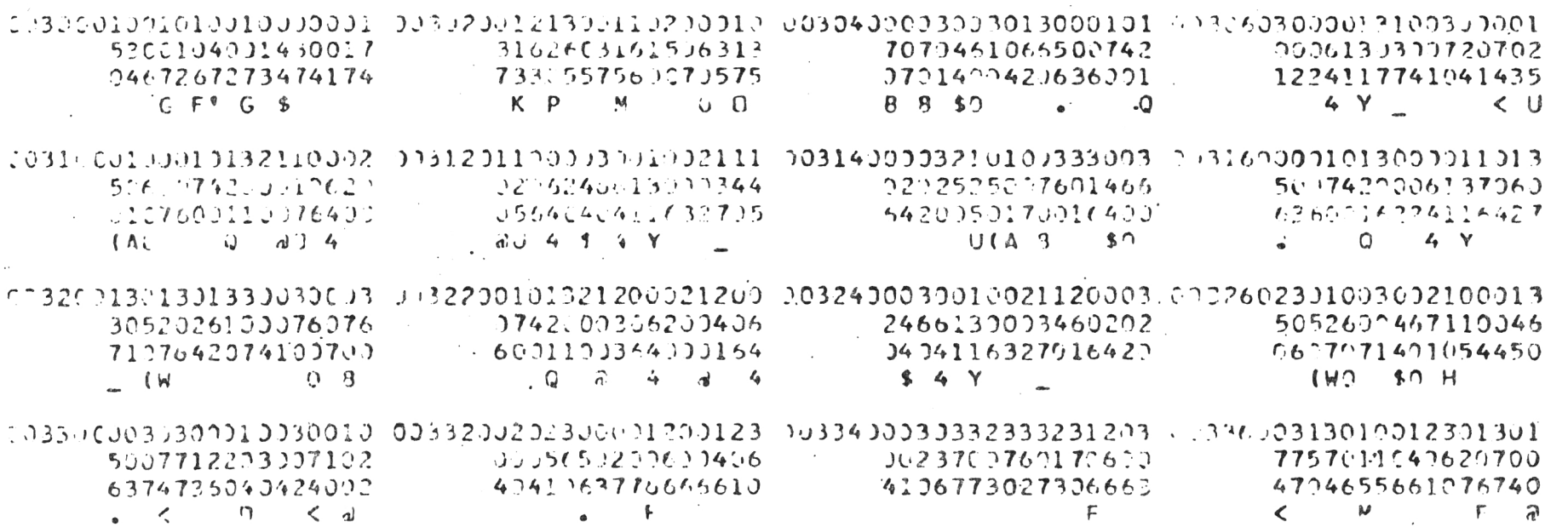


รูปที่2.19 แสดงส่วนของ ดิส ไฟล์ แฮนด์ลิ่ง รูทีน

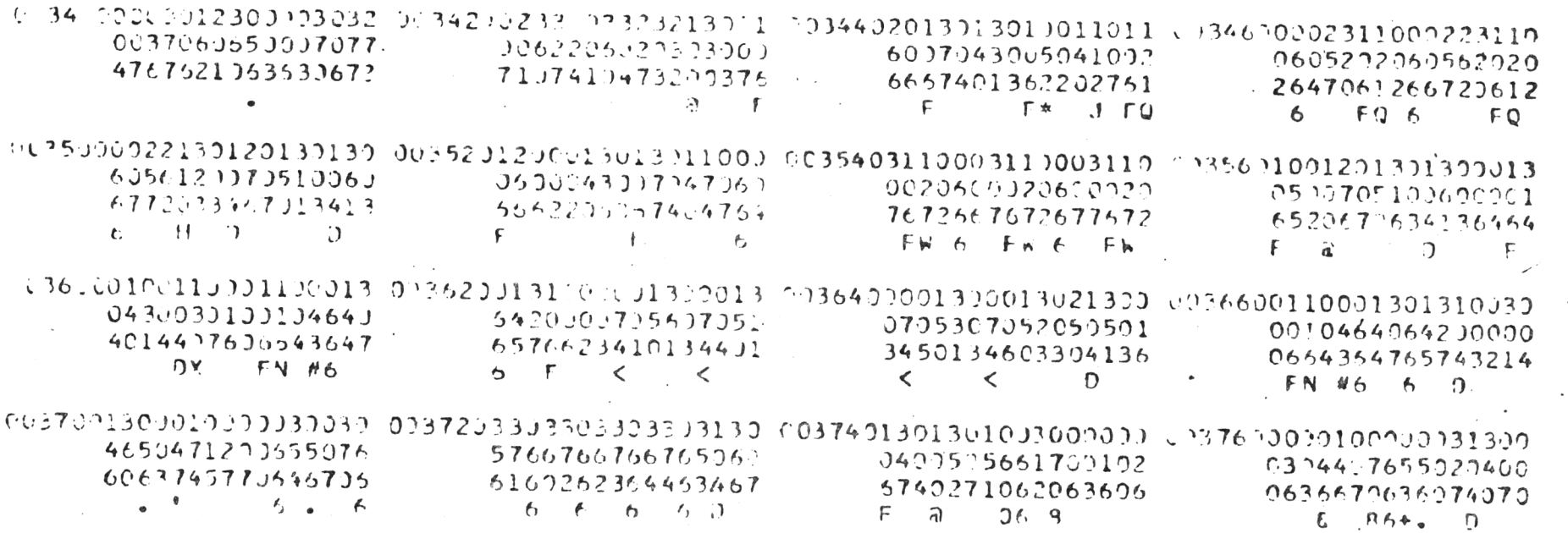
1196916



รูปที่ 2.19 (ต่อ) แสดงส่วนของ ดิส ไฟล์ แกนคลิ่ง รูทีน



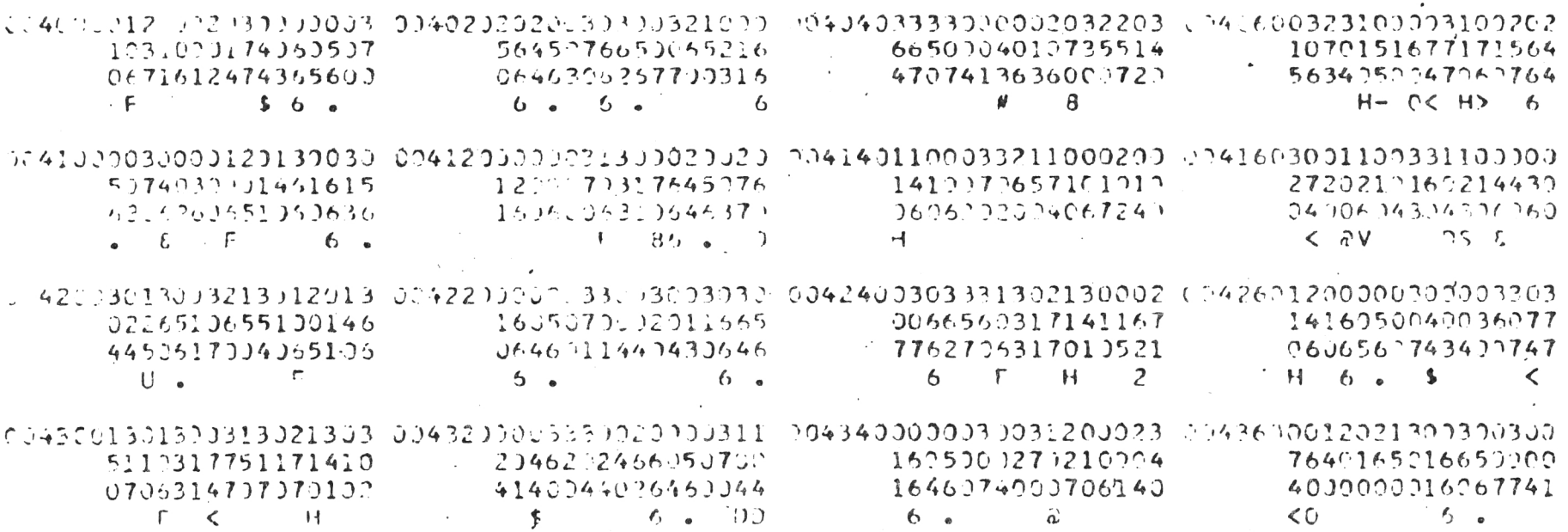
รูปที่ 2.19(ต่อ) แสดงส่วนของ ศีส ไฟล์ แฮนดลิ่ง รูทีน



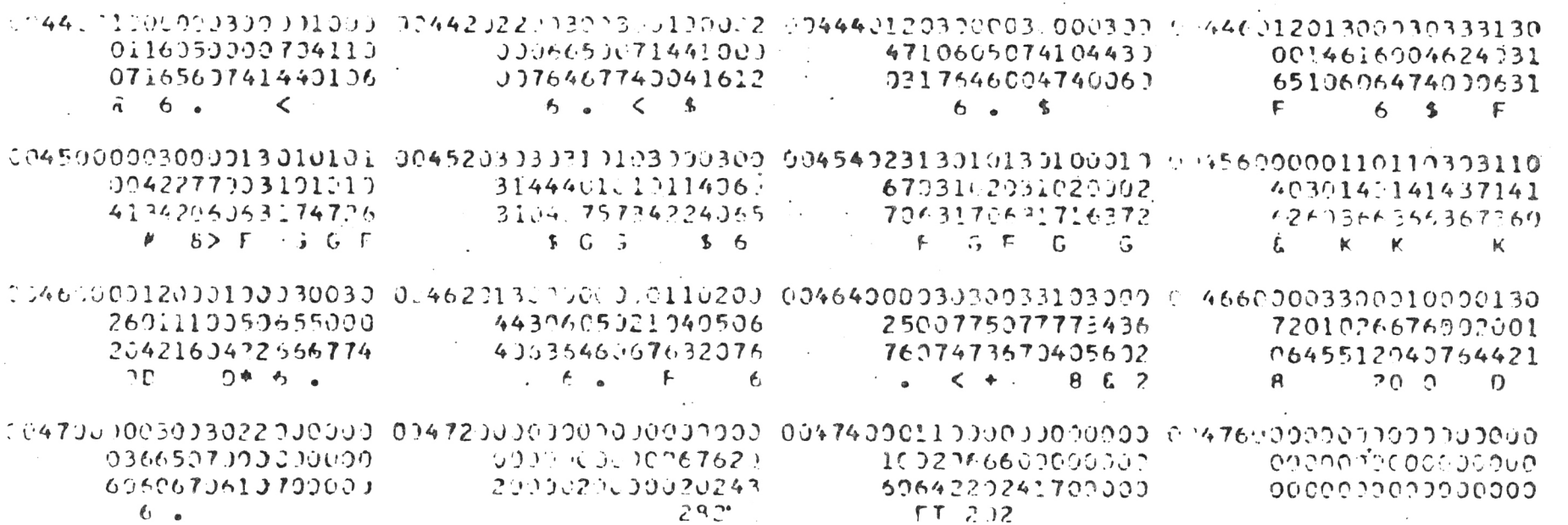
รูปที่2.19 (ต่อ) แสดงส่วนของ ดิส ไฟล์ แฮนด์ลิ่ง รูทีน

2.2.2.5 ไฟล์ แฮนด์ลิ่ง โอเวอร์เลย์(File Handing Overlays) มีขึ้นเพื่อลดการใช้เนื้อที่ในหน่วยความจำ โดยใช้เนื้อที่ตั้งแต่ แอดเดรสที่ 04000 ฐานแปด ถึง 05377 ฐานแปด โดยจะใช้เป็นที่เก็บโปรแกรมจากแฟ้มข้อมูล SYSTEM1/SYS ถึง SYSTEM6/SYS ซึ่งในขณะใดขณะหนึ่งจะถูกเรียกใช้งานเพียงโปรแกรมเดียว นั่นคือในขณะหนึ่ง ๆ เนื้อที่หน่วยความจำส่วนนี้ จะเก็บโปรแกรมจากแฟ้มข้อมูลในจานแม่เหล็กเพียงแฟ้มเดียวและเมื่อเรียกใช้โปรแกรมใหม่ แฟ้มข้อมูลใหม่ก็จะถูกเรียกจากที่เก็บในจานแม่เหล็กเข้าไปทับโปรแกรมเดิมที่มีอยู่ เพื่อใช้งานใหม่ต่อไป โดยที่อยู่ของ ดี โอ เอส ในหน่วยความจำหลักจะถูกเก็บอยู่ในแฟ้มข้อมูล SYSTEM0/SYS ส่วนแฟ้มข้อมูลอื่นมีหน้าที่ดังนี้

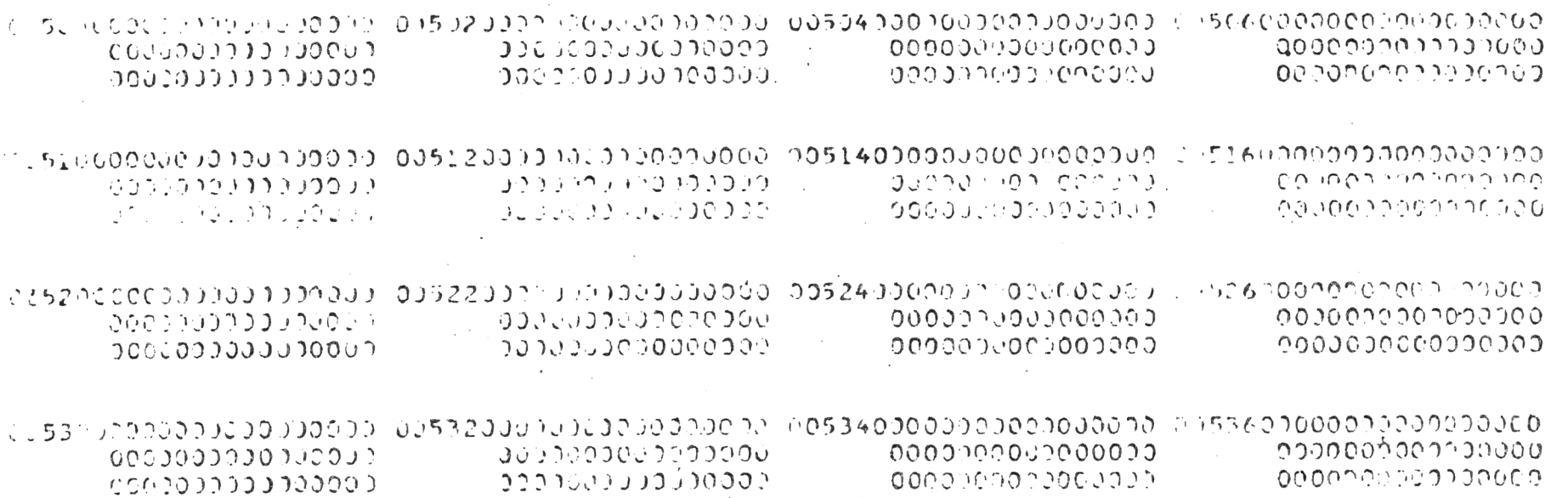
SYSTEM1/SYS	ทำหน้าที่สร้างแฟ้มข้อมูลใหม่ในจานแม่เหล็ก
SYSTEM2/SYS	ทำหน้าที่ปิดแฟ้มข้อมูลหนึ่ง ๆ หลังจากถูกใช้งานแล้ว
SYSTEM3/SYS	ทำหน้าที่เปิดแฟ้มข้อมูลที่มีอยู่แล้ว เพื่อเตรียมที่จะใช้งาน
SYSTEM4/SYS	ทำหน้าที่หาเนื้อที่ ขยายให้แฟ้มข้อมูลที่มีอยู่
SYSTEM5/SYS	ทำหน้าที่แสดงข้อผิดพลาดของคำสั่งหนึ่ง ๆ บนจอภาพ
SYSTEM6/SYS	ทำหน้าที่บอกจุดเริ่มต้นของแรม(RAM) ของจอภาพ
SYSTEM7/SYS	ทำหน้าที่เก็บส่วนของ DOS FUNCTION



รูปที่ 2.20 แสดงส่วนของไฟล์ แฮนด์ลิ่ง โอเวอร์เลย์

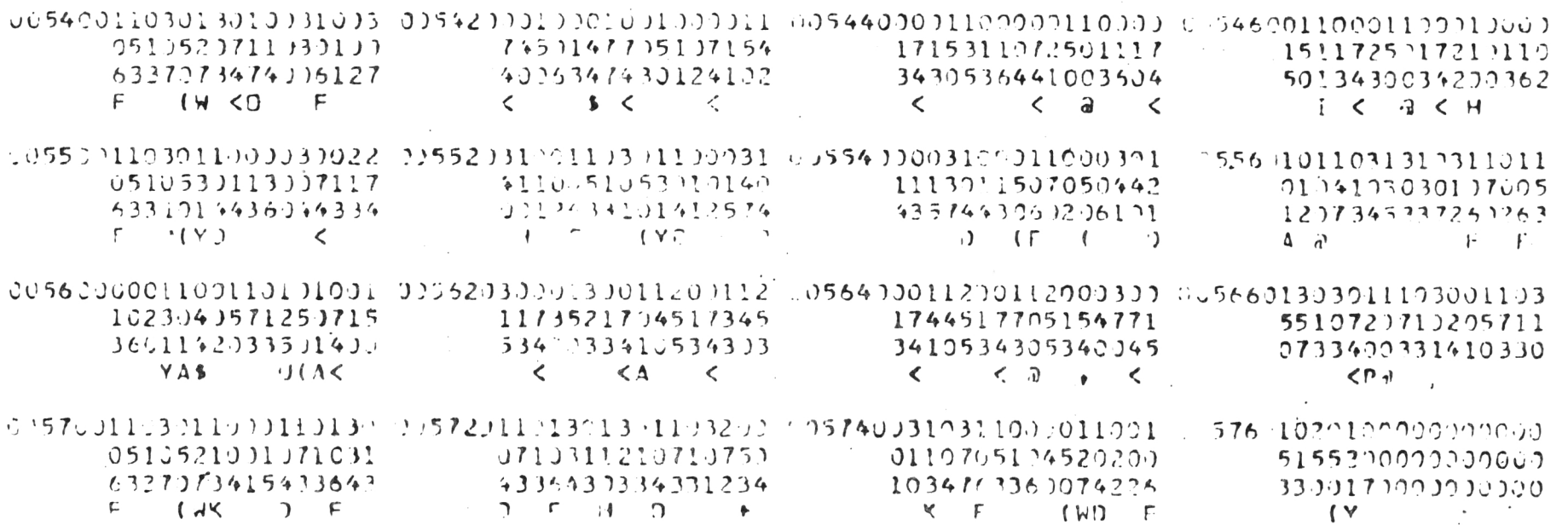


รูปที่2.20 (ต่อ) แสดงส่วนของไฟล์ แฮนด์ลิ่ง โคเวอร์เอย์



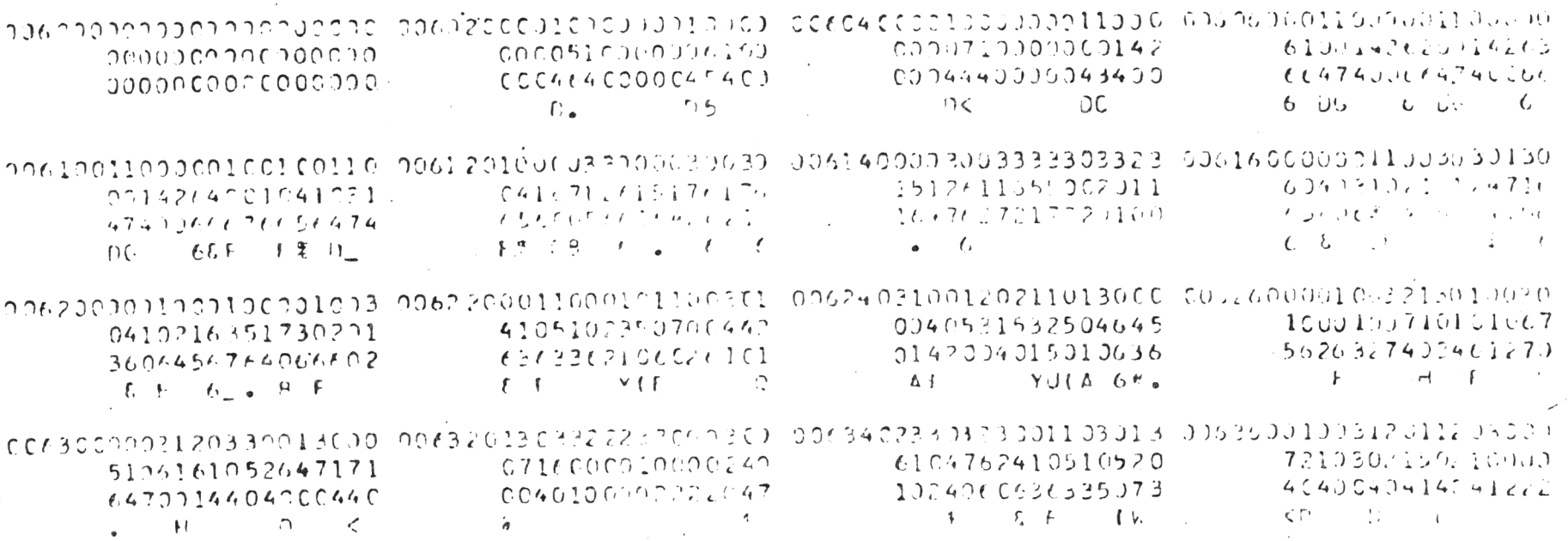
รูปที่2.20 (ต่อ) แสดงส่วนของไฟล์ แฮนด์ลิ่ง โอเวอร์เลย์

2.2.2.6 KEYIN & DISPLAY เป็นส่วนของโปรแกรมที่ทำหน้าที่ในการรับข้อมูลเข้าทางแป้นสำหรับรับข้อมูล(KEYBOARD) และแสดงข้อมูลบนจอภาพแสดงข้อมูล (CRT DISPLAY) อยู่ในหน่วยความจำ ตั้งแต่แอดเดรสที่ 05400 ฐานแปด ถึง 05777 ฐานแปด ดังรูปที่ 2.21

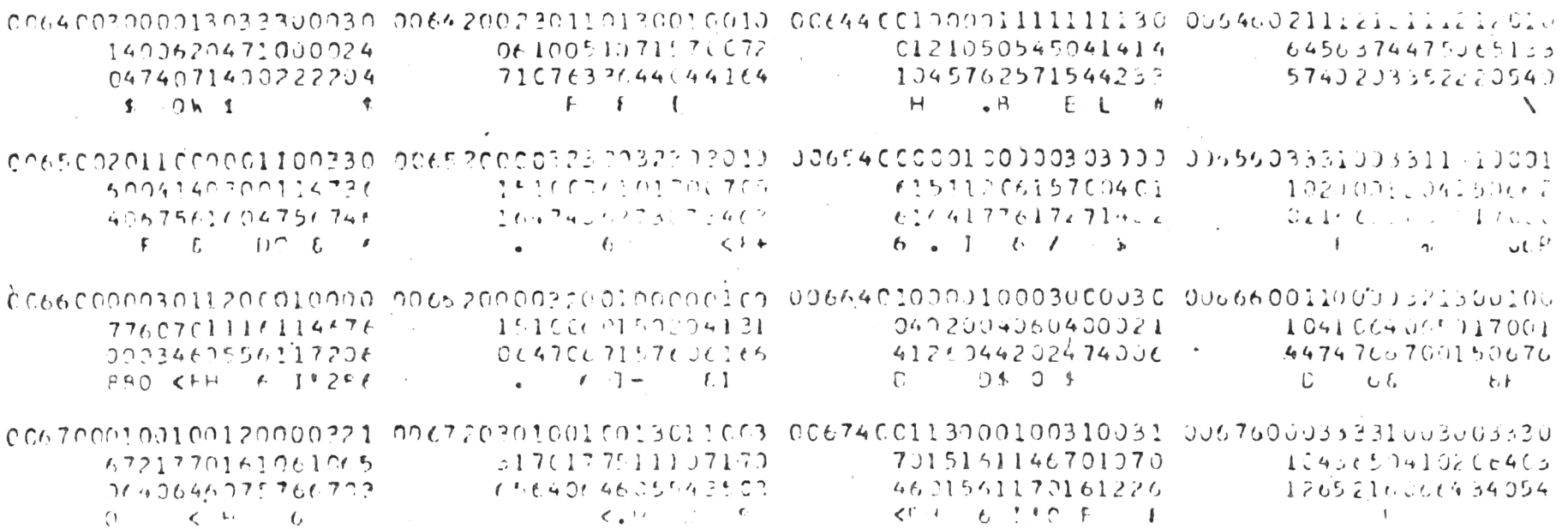


รูปที่2.21 แสดงส่วนของคีย์อินและกิสเพลย์ รูทีน

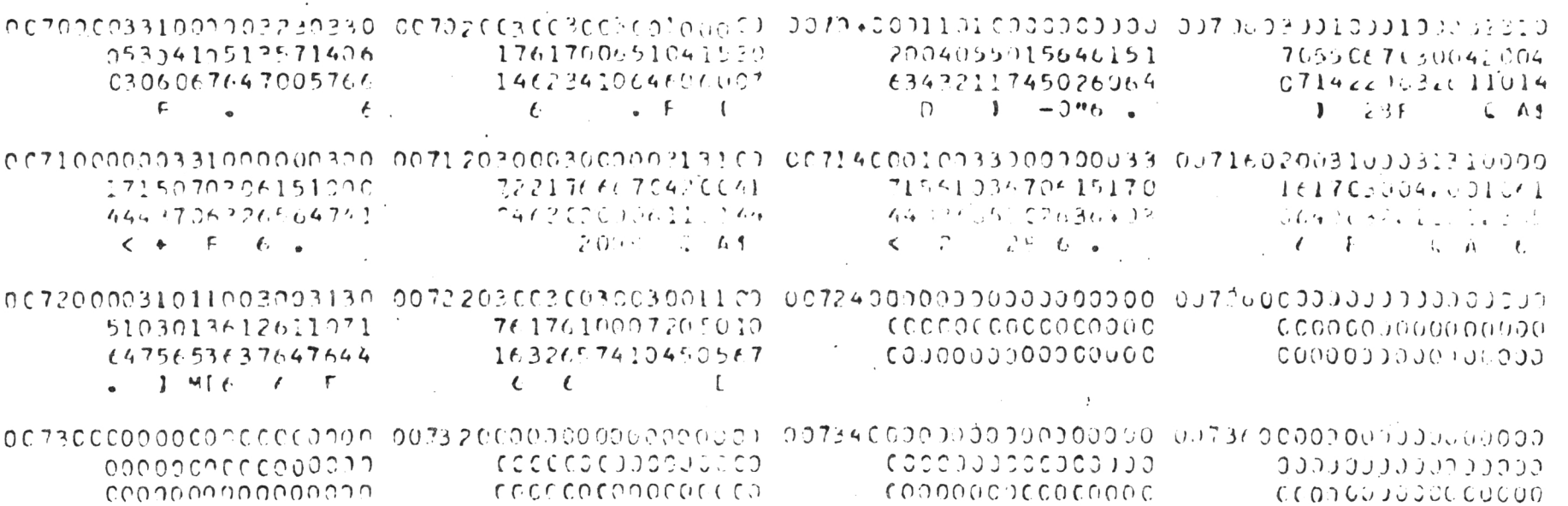
2.2.2.7 DEBUG เป็นโปรแกรมที่ใช้ช่วยในการ debug โปรแกรม ผู้ที่จะใช้โปรแกรมนี้ช่วยค้นหาที่ผิดต้องสามารถจดจำคำสั่งของเครื่องในลักษณะที่เป็นภาษาเครื่อง (Machine Language) ได้ เช่น 307 หมายถึงคำสั่งในภาษาแอสเซมบลี คือ LAM เป็นต้น ความสามารถของโปรแกรมนี้มีหลายอย่าง อาทิเช่น สามารถเรียกใช้แฟ้มข้อมูลในจานแม่เหล็ก โดยใช้เอรีของแฟ้มข้อมูล (FFS) สามารถตรวจสอบข้อมูลที่อยู่ในหน่วยความจำหลัก ณ แอดเดรสที่ต้องการในขณะนั้น พร้อมทั้งสามารถทำการแก้ไขข้อมูลนั้น ๆ ได้ สามารถกำหนดให้โปรแกรมหยุดทำงาน ณ จุดใดจุดหนึ่ง โดยที่ไม่เปลี่ยนแปลงค่าในหน่วยความจำหลักในขณะนั้น ทั้งนี้เพื่อให้ผู้ใช้สามารถใช้งานได้โดยสะดวกและง่ายต่อการใช้ โปรแกรมนี้ใช้เนื้อที่ในหน่วยความจำหลักตั้งแต่แอดเดรสที่ 06000 ฐานแปด ถึง 07377 ฐานแปด ดังรูปที่ 2.22



รูปที่2.22 แสดงส่วนของ คีบัก รูทีน

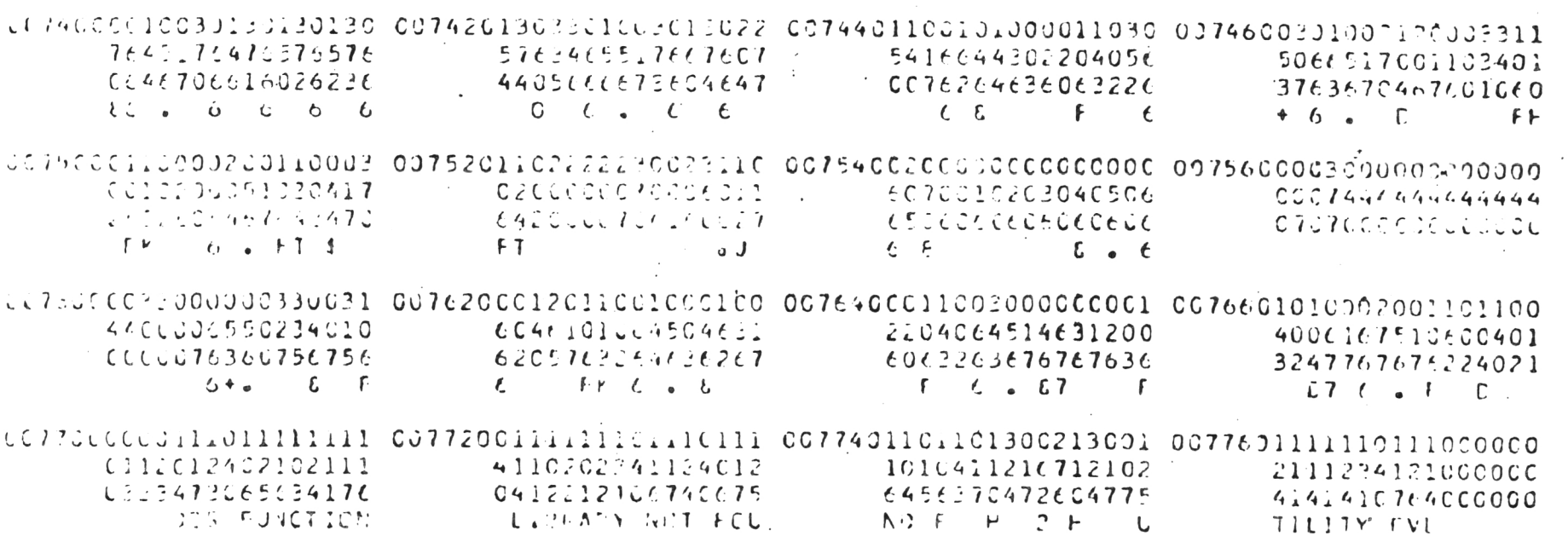


รูปที่ 2.22 (ต่อ) แสดงส่วนของ คีย์บัก รูทีน



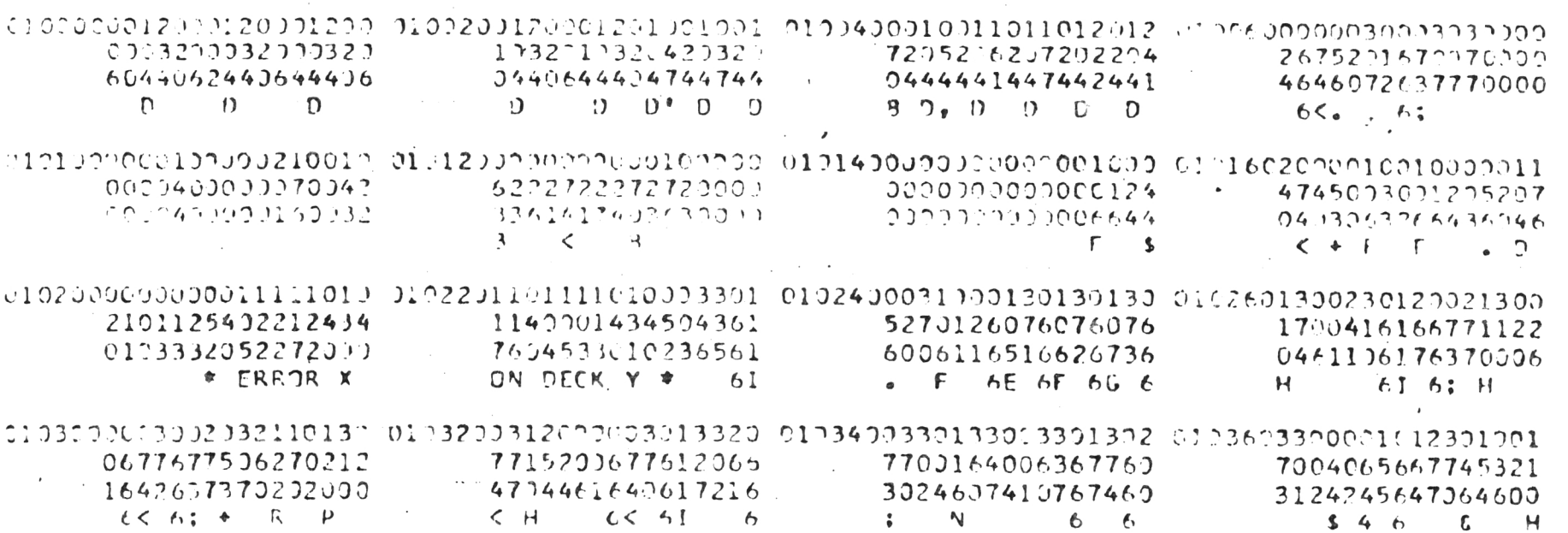
รูปที่ 2.22(ต่อ) แสดงส่วนของ คีบัก รูทีน

2.2.2.8 DOS FUNCTION ใช้เนื้อที่ในหน่วยความจำหลักตั้งแต่แอดเดรสที่ 07400 ฐานแปด ถึง 07777 ฐานแปด โดยแบ่งเนื้อที่ในการใช้งานออกเป็น 2 ส่วน ส่วนต้นของเนื้อที่นี้ใช้เป็นที่สำหรับเก็บคำสั่งในการนำส่วนของโปรแกรมที่ใช้ควบคุมการทำงานเกี่ยวกับจานแม่เหล็กจากแฟ้มข้อมูล SISTEM7/SIS ซึ่งอยู่ในจานแม่เหล็กเข้ามาเก็บไว้ในหน่วยความจำส่วนที่สองของเนื้อที่นี้เฉพาะโปรแกรมที่ต้องการใช้งานในขณะนั้น โปรแกรมที่ใช้ในการควบคุมเกี่ยวกับจานแม่เหล็กนี้มีได้มากที่สุด 255 โปรแกรมแล้วแต่ว่าเป็น ดี โอ เอส รุ่นใด แต่ละโปรแกรมมีความยาวได้ไม่เกิน 124 ไบท์ ถ้ามีการเรียกใช้โปรแกรมซึ่งมีเอรที่เกินจากที่มีอยู่จริง เครื่องจะไม่ยอมทำงานและจะแสดงข้อความบอกถึงข้อผิดพลาดออกมาทางจอภาพแสดงข้อมูล เพื่อให้ผู้ใช้ได้ทราบ

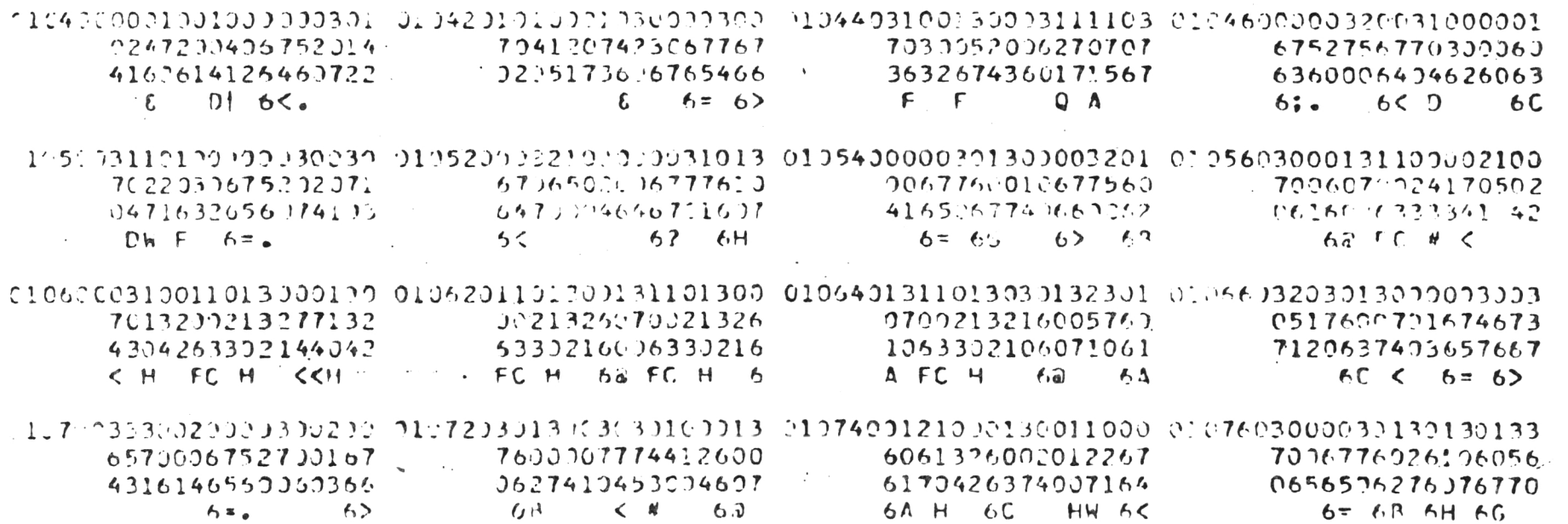


รูปที่ 2.23 แสดงส่วนของคอล์ ฟังก์ชั่น รูทีน

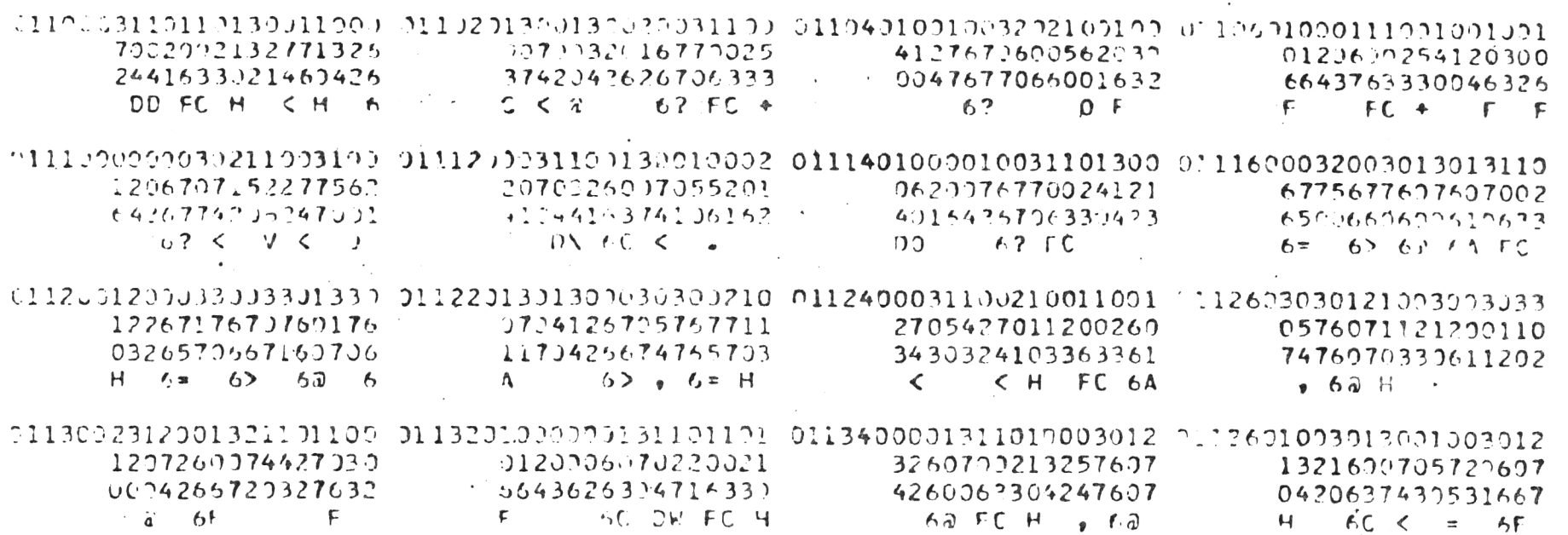
2.2.2.9 CASSETTE DRIVER เป็นส่วนของโปรแกรมที่ทำหน้าที่เกี่ยวกับการอ่านและบันทึกข้อมูลของเทปคลับ (Cassette Tape) ประกอบด้วยคำสั่งในการจัดรูปแบบของแฟ้มข้อมูลในเทปคลับ การให้ชื่อแฟ้มข้อมูลและการสร้างรหัสในการบอกการสิ้นสุดของแฟ้มข้อมูล ส่วนของโปรแกรมนี้อยู่ในหน่วยความจำหลัก ตั้งแต่แอดเดรสที่ 010000 ฐานแปดถึง 012377 ฐานแปด



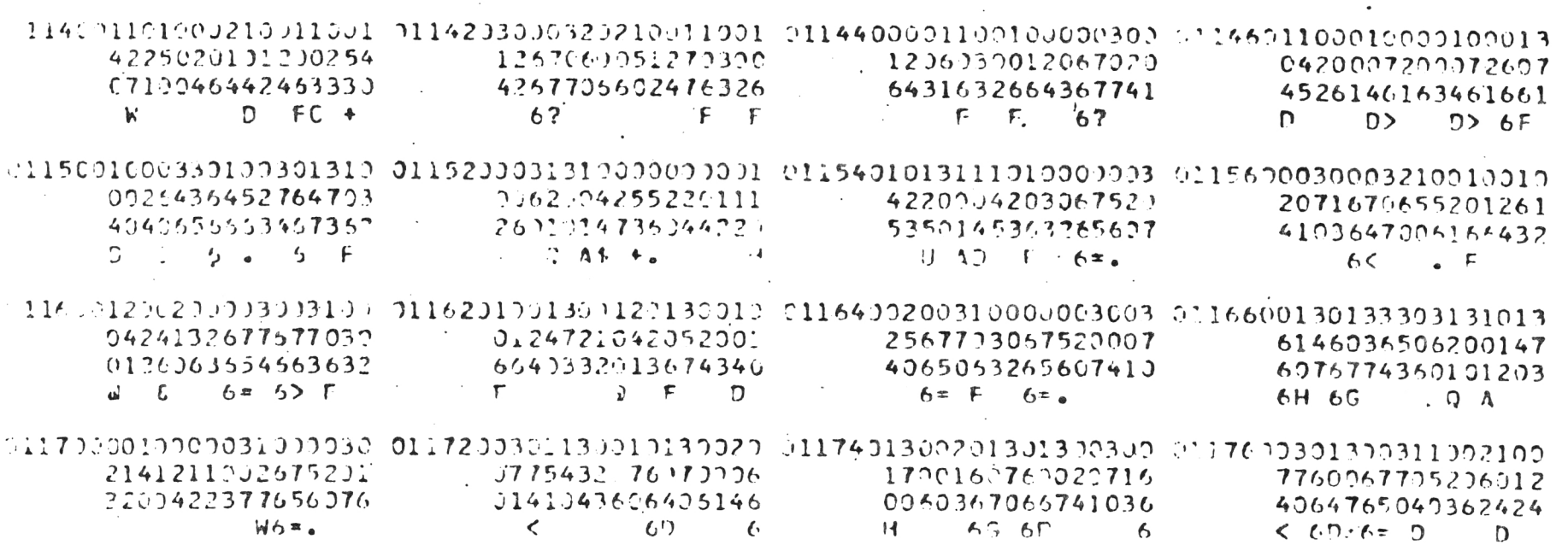
รูปที่ 2.24 แสดงส่วนของแคสเซท ไดร์เวอร์ รูทีน



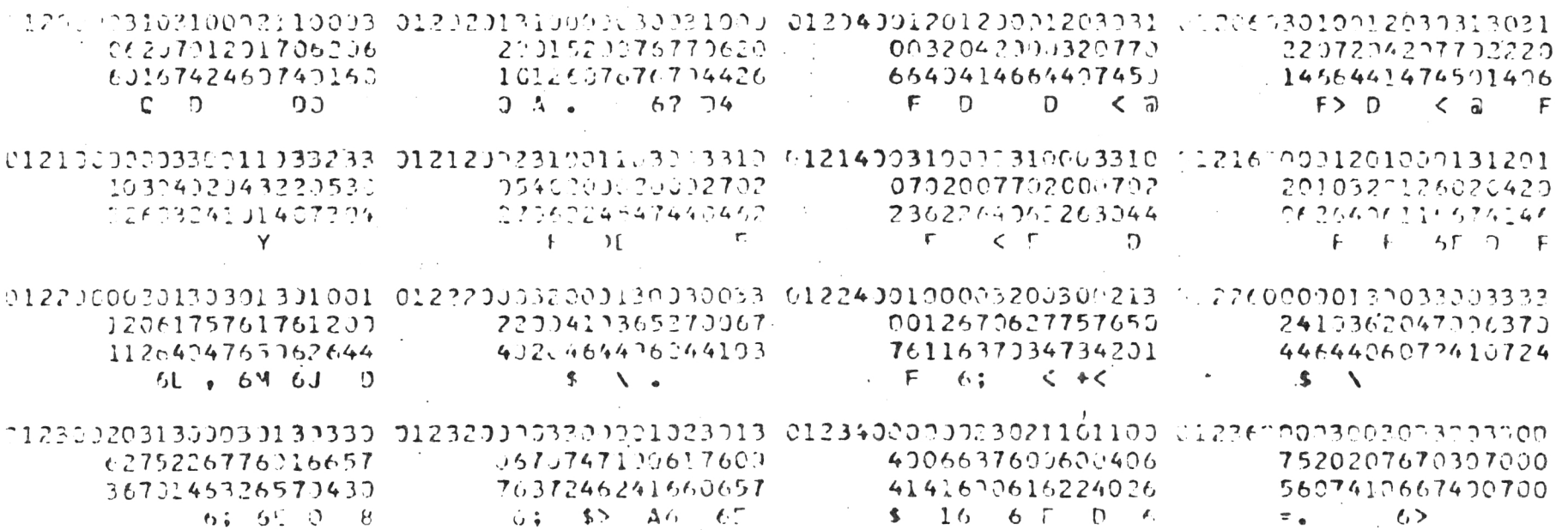
รูปที่ 2.24(ต่อ) แสดงส่วนของแคสเซท ไดร์เวอร์ รูทีน



รูปที่ 2.24 (ต่อ) แสดงส่วนของแคสเซท ไดร์เวอร์ รูทีน

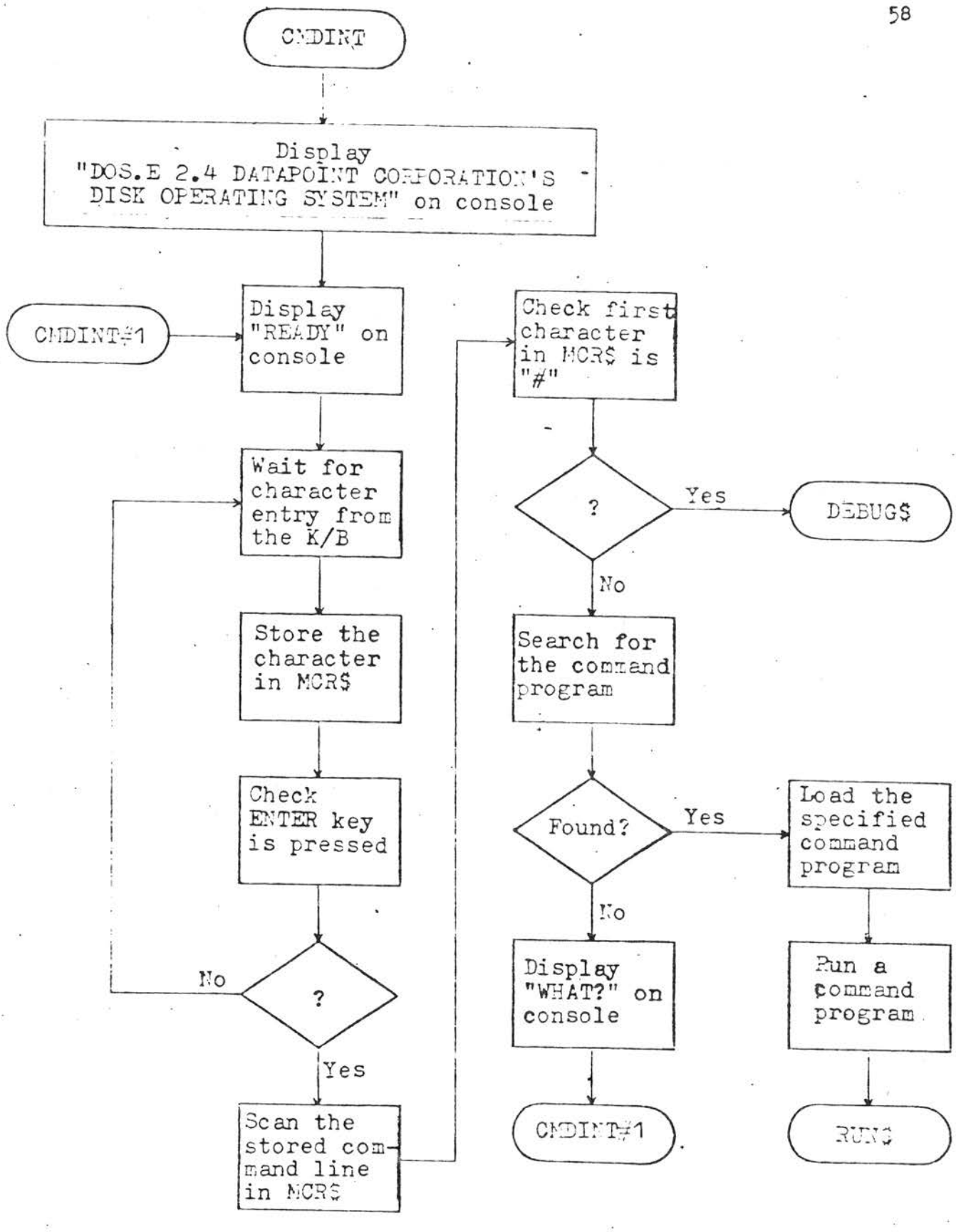


รูปที่ 2.24 (ต่อ) แสดงส่วนของแคสเซท ไดร์เวอร์ รูทีน

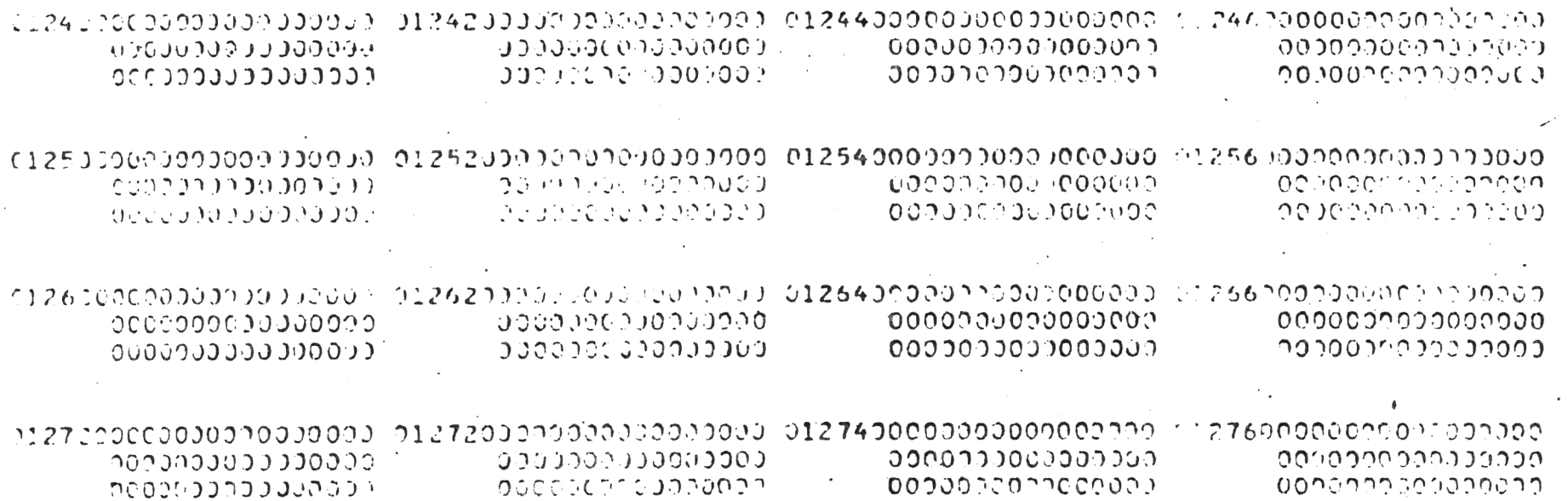


รูปที่2.24 (ต่อ) แสดงส่วนของแคสเซท ไดร์เวอร์ รูทีน

2.2.2.10 COMMAND INTERPRETER - เป็นส่วนของโปรแกรมที่ทำหน้าที่ในการตรวจสอบและแปรคำสั่งที่ได้รับจากแป้นสำหรับรับข้อมูล ซึ่งคำสั่งเหล่านั้นจะถูกเก็บอยู่ในส่วนของหน่วยความจำที่ 01400 ฐานแปด ถึง 01543 ฐานแปด ซึ่งเรียกว่า Monitor Communication Region (MCRS) เมื่อตรวจสอบถูกต้องแล้วจะทำการดึงโปรแกรมที่สอดคล้องกับคำสั่งนั้นจากแฟ้มข้อมูลซึ่งอยู่ในจานแม่เหล็กขึ้นมาเก็บไว้ในหน่วยความจำหลัก ณ แอดเดรสที่โปรแกรมนั้นกำหนดและกระโดดไปยังแอดเดรสที่เป็นจุดเริ่มต้นการทำงานของโปรแกรมนั้น เพื่อทำงานตามโปรแกรมนั้นต่อไป ใช้เนื้อที่ในหน่วยความจำหลักตั้งแต่แอดเดรสที่ 012400 ฐานแปด ถึง 016777 ฐานแปด

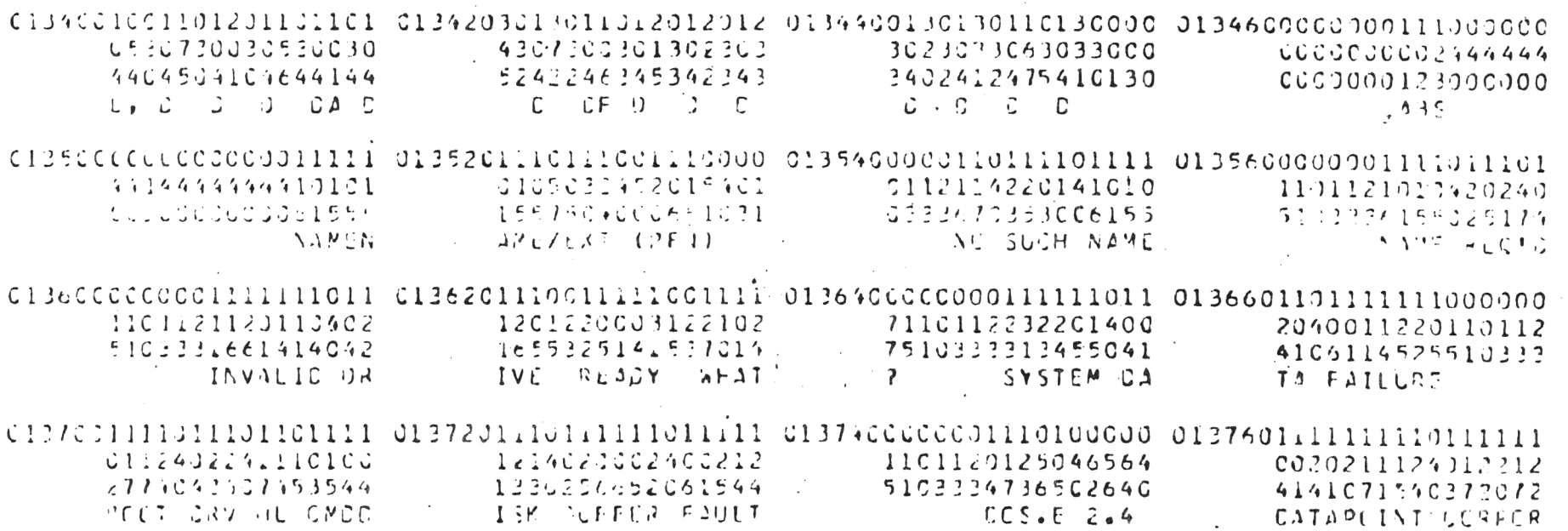


รูปที่ 2.25 แสดงขั้นตอนการทำงานของคอมมานด์ อินเตอร์พรีเตอร์

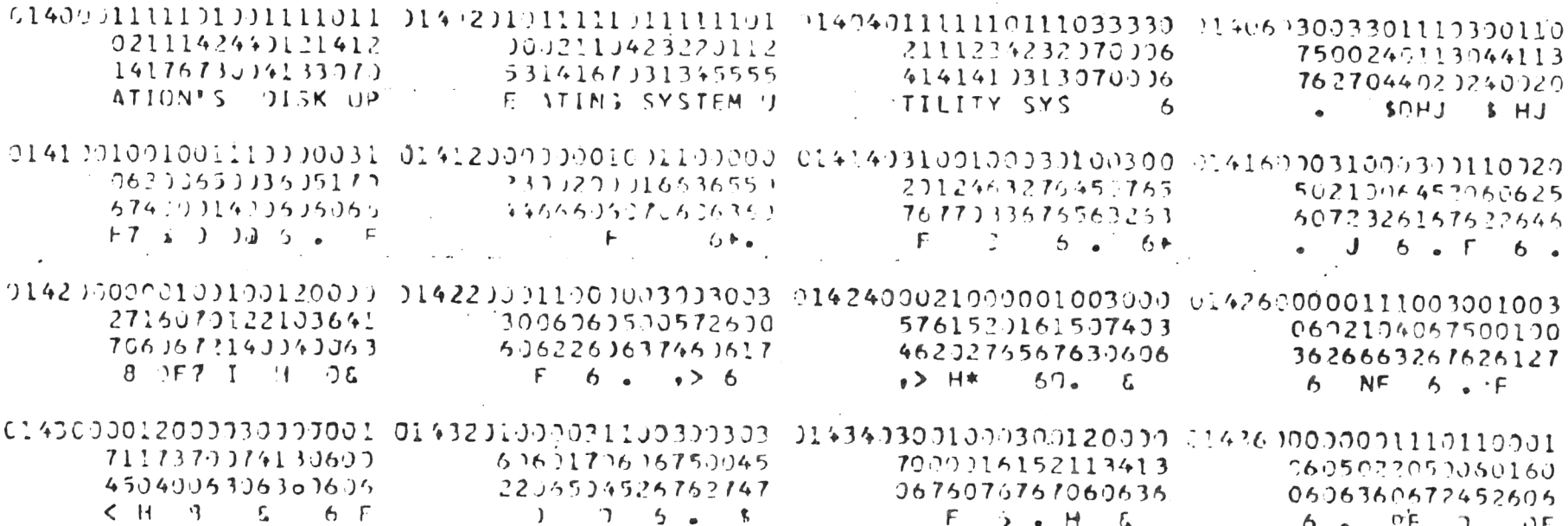


รูปที่ 2.26 แสดงส่วนของคอมมานด์ อินเตอร์พรีเตอร์

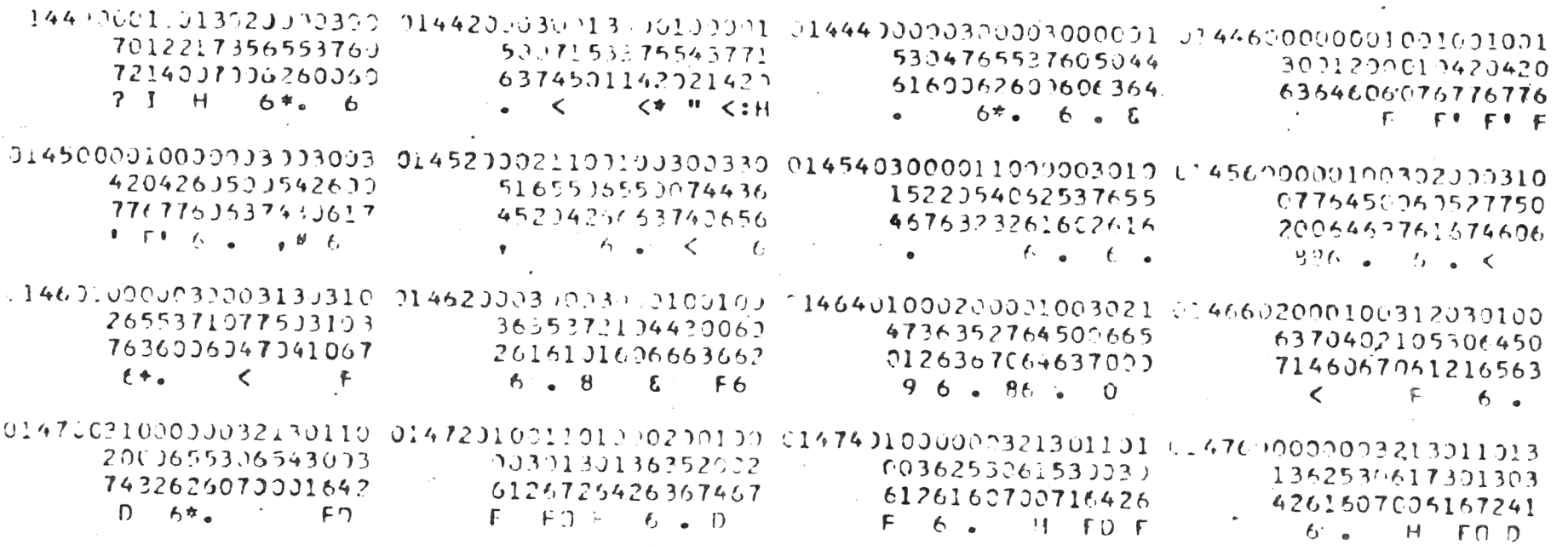
รูปที่ 2.26(ต่อ) แสดงส่วนของคอมมานด์ อินเตอร์พริเตอร์



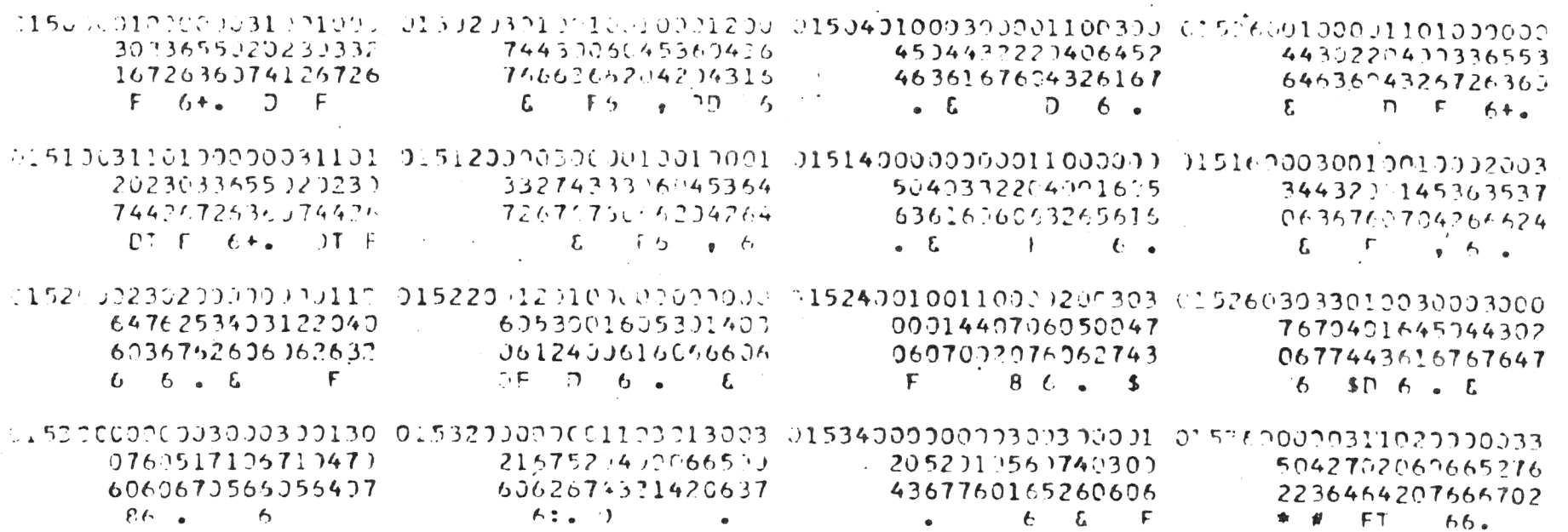
รูปที่ 2.26 (ต่อ) แสดงส่วนของคอมมานด์ อินเตอร์พริเตอร์



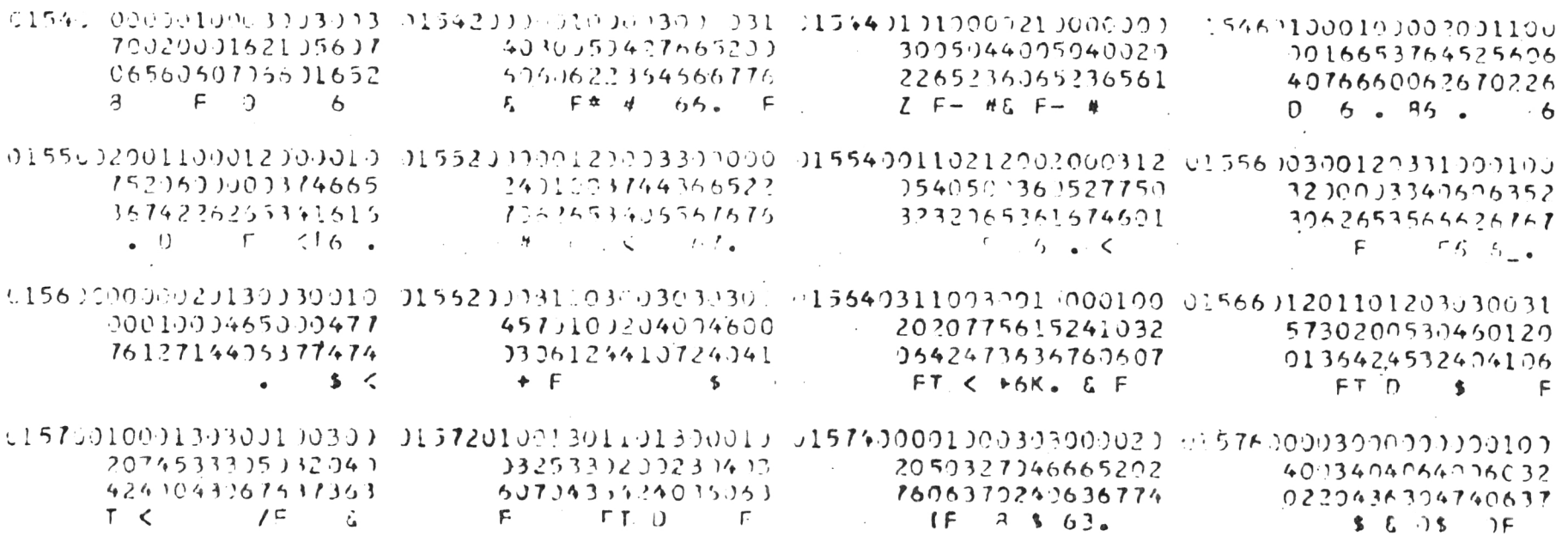
รูปที่ 2.26(ต่อ) แสดงส่วนของคอมมานด์ อินเตอร์พรีเตอร์



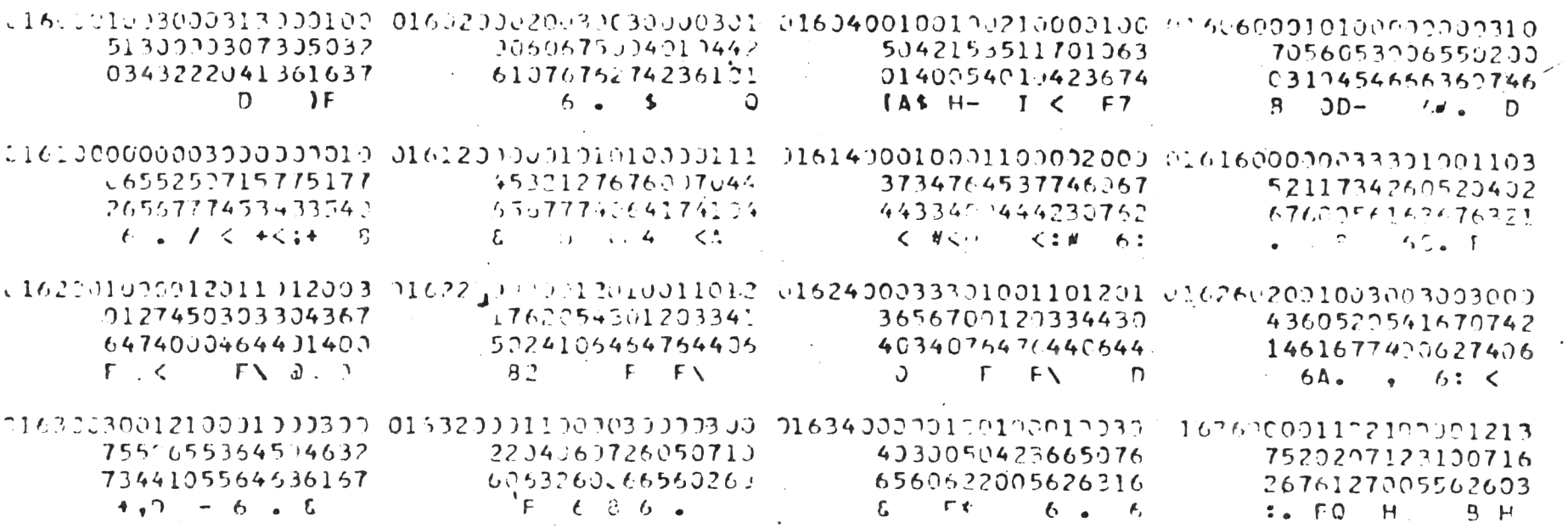
รูปที่ 2.26(ต่อ) แสดงส่วนของคอมมานด์ อินเตอร์พริเตอร์



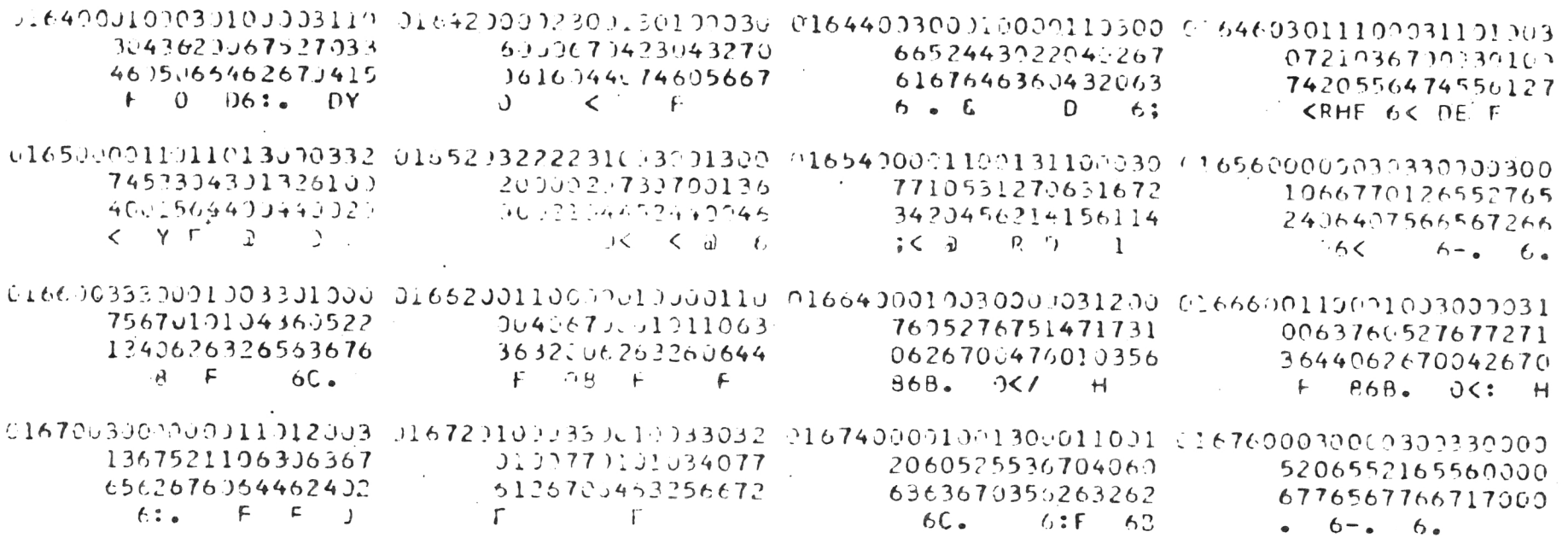
รูปที่ 2.26(ต่อ) แสดงส่วนของคอมมานด์ อินเตอร์พรีเตอร์



รูปที่ 2.2(ต่อ) แสดงส่วนของคอมมาชัต อินเตอร์พรีเตอร์



รูปที่ 2.26(ต่อ) แสดงส่วนของคอมมานด์ อินเตอร์พรีเตอร์



รูปที่ 2.26(ต่อ)แสดงส่วนของคอมมานด์ อินเตอร์พริเตอร์

2.2.2.11 COMMAND OVERLAYS เป็นส่วนของหน่วยความจำหลักที่ผู้ใช้สามารถนำไปใช้งานได้โดยไม่กระทบกระเทือนกับคำสั่งของ ดี โอ เอส ใด ๆ ผู้ใช้อาจกำหนดให้เป็นที่เก็บตัวโปรแกรมของผู้ใช้หรือใช้เป็นที่เก็บข้อมูลที่ต้องการได้สามารถใช้ได้ตั้งแต่แอดเดรสที่ 017000 ฐานแปด ขึ้นไปเรื่อย ๆ จนถึงค่าสูงสุดของแอดเดรสที่เครื่องสามารถมีได้

2.3 ลักษณะการบันทึกข้อมูลของบนจานแม่เหล็ก

ดี โอ เอส จะไม่ยุ่งเกี่ยวกับข้อมูลของผู้ใช้ที่อยู่ในระดับซึ่งต่ำกว่าระดับของเรคคอร์ด (Record Level) แต่ ดี โอ เอส จะทำหน้าที่คอยติดตามว่าเรคคอร์ดต่าง ๆ นั้นถูกบันทึกไว้ที่ไหนบ้าง ดังนั้น ดี โอ เอส จึงเปิดโอกาสให้ผู้ใช้สามารถที่จะจัดรูปแบบของข้อมูลในลักษณะใดก็ได้ตามที่ผู้ใช้ต้องการ โดยเรคคอร์ดแต่ละเรคคอร์ดจะมีความยาวได้ไม่เกิน 253 ไบท์ ทั้งนี้เนื่องจาก ดี โอ เอส มีความจำเป็นต้องใช้ไบท์ที่ 1 ของแต่ละเซกเตอร์เพื่อใช้สำหรับเก็บค่าของฟิสิกส์คัล ไฟล์ นัมเบอร์ (Physical File Number) และใช้ไบท์ที่ 2 และ 3 ของแต่ละเซกเตอร์สำหรับเก็บค่าของซิสเตม โลจิกคัล เรคคอร์ด นัมเบอร์ (System Logical Record Number) เพื่อใช้ในการอ้างอิงทางระบบ ส่วนอีก 253 ไบท์ ที่เหลือผู้ใช้สามารถนำไปใช้บันทึกข้อมูลได้ ดี โอ เอส ได้แบ่งแฟ้มข้อมูลที่บันทึกอยู่ในจานแม่เหล็กออกเป็น 2 ประเภท คือ แฟ้มข้อมูลที่เป็นออบเจค เรคคอร์ด (Object Record) และแฟ้มข้อมูลที่เป็น ซิมบอลิก คาคา เรคคอร์ด (Symbolic Data)

ออบเจค เรคคอร์ด หมายถึง เรคคอร์ด ชุดอย่างที่สามารถโลดเข้าสู่หน่วยความจำหลักได้โดย ซิสเตมโลดเดอร์ ส่วนซิมบอลิก คาคา เรคคอร์ด หมายถึงเรคคอร์ดทั้งหมดที่สแคนคาร์ด คาคา แอนอิง โปรแกรม สามารถทำการควบคุมดูแลได้

2.3.1 ชื่อแฟ้มข้อมูล (File Name) แฟ้มข้อมูลที่ถูกบันทึกอยู่ในจานแม่เหล็กจะถูกเจาะจงโดยชื่อ (Name) ส่วนขยาย (Extension) และโลจิกคัล ไดร์ว นัมเบอร์ (Logical Drive Number) ชื่อสามารถตั้งได้ตั้งแต่ 1 ถึง 8 ตัวอักษร อัลฟานิวเมอริค (Alphanumeric) ตัวอย่างการตั้งชื่อได้แก่

EDIT	PAYROLL
EMPLOYEE	JUL1075
23NOV76	XYZ

ส่วนขยาย (Extension) ต้องเริ่มต้นด้วยตัวอักษร (Alphabetic) และอาจจะตาม ด้วยตัวอักษรอัลฟานิวเมอริคอีก 1 หรือ 2 ตัว เมื่อมีส่วนขยายปรากฏอยู่ในแฟ้มข้อมูล จะ ต้องแยกออกจากชื่อด้วยเครื่องหมาย "/" ส่วนขยายนอกจากจะใช้บอกถึงแฟ้มข้อมูลที่ต้องการ แล้ว ยังบอกถึงชนิดของข้อมูลที่แฟ้มข้อมูลนั้นบรรจุอยู่ เช่น "TXT" หมายถึง Text ซึ่ง อาจจะเป็นคาดาหรือโปรแกรมที่ผู้ใช้เรียนขึ้นก็ได้ "ABS" หมายถึงตัวโปรแกรมที่ผ่านการ คอมไพล์ (Compile) หรือ สแนป (Snap) แล้ว และซิสเตม โลดเดอร์ สามารถทำ การโลดลงสู่หน่วยความจำเพื่อใช้งานได้ "CMD" หมายถึงโปรแกรมที่สามารถเรียกใช้ได้ จากซิสเตม คอนโซล (System Console) ส่วนขยายอื่น ๆ นอกจากที่กล่าวมาแล้ว ได้ แก่ "REL", "ISI", "DBC", "SYS", "PRT", และ 'OVI"

โลจิกคัล ไดรว์ นัมเบอร์ (Logical Drive Number) ใช้กำหนดว่าแฟ้ม ข้อมูลที่ต้องการนั้นอยู่หรือจะอยู่ที่ไดรว์ นัมเบอร์อะไร การกำหนดไดรว์ มีรูปแบบเป็น ":DRn" หรือ ":Dn" หรือ ": Volid " เมื่อ n เป็นตัวเลขที่บอกถึงโลจิกคัล ไดรว์ นัมเบอร์ ของจานแม่เหล็ก ซึ่งถูกกำหนดขึ้นตอนติดตั้งระบบ (System Installation) ส่วน ": Volid " ประกอบขึ้นด้วยตัวอักษรตั้งแต่ 1 ถึง 8 ตัว เพื่อใช้บอกถึงชื่อของวอลุมที่ ต้องการโดยไม่คำนึงว่าวอลุมที่กำหนดนี้อยู่ที่ไดรว์ นัมเบอร์อะไร

รูปแบบที่สมบูรณ์ของชื่อแฟ้มข้อมูลมีดังนี้

Name/Extension : Drive

เมื่อชื่อแฟ้มข้อมูลถูกใช้เป็นส่วนหนึ่งของคำสั่ง (Command) ส่วนประกอบต่าง ๆ ที่ประกอบ ขึ้นเป็นชื่อแฟ้มข้อมูล อาจไม่จำเป็นต้องมีครบทั้ง 3 ส่วนเสมอไป ทั้งนี้แล้วแต่ความจำเป็น การพิจารณาว่ามีส่วนใดบ้างสามารถพิจารณาได้จาก "/" และ ":" ดังตัวอย่าง

NAME/ABS:DR0	/ABS:DR1
NAME/REL	/TXT
NAME:D0	:D2
NAME	NAME:DOSR1

หากส่วนของชื่อแฟ้มข้อมูลไม่ได้ถูกกำหนดให้ ดี ไอ เอส จะใช้ค่าที่ ดี ไอ เอส กำหนดขึ้น (Default Valve) ซึ่งค่าที่จะใช้นี้ขึ้นอยู่กับตำแหน่งของชื่อที่ปรากฏอยู่บนบรรทัดคำสั่ง (Command line) และขึ้นกับคำสั่งที่ใช้ ตำแหน่งแรกที่ปรากฏอยู่บนบรรทัดคำสั่งจะเป็นชื่อของโปรแกรมคำสั่ง (Command Program) ที่ต้องการจะใช้ทำงาน ตำแหน่งแรกนี้จะเว้นว่างไม่ได้ ต้องเป็นชื่อโปรแกรมเสมอ ถ้าไม่ใส่ส่วนขยาย ดี ไอ เอส จะถือว่าเป็น "CMD" และถ้าไม่กำหนดไดรว์ ดี ไอ เอส จะถือว่าเป็นไดรว์ ใดก็ได้ โดยจะทำการค้นหาจากไดรว์ที่มีอยู่ทั้งหมดในขณะนั้นและอยู่ในลักษณะออนไลน์ (On - line) การค้นหาจะเริ่มจากไดรว์หมายเลข 0 ก่อนเสมอ ส่วนตำแหน่งอื่น ๆ ของบรรทัดคำสั่งจะเป็นไปตามข้อกำหนดของแต่ละคำสั่ง

2.3.2 การสร้างแฟ้มข้อมูล (File Creation) การสร้างแฟ้มข้อมูลในระบบ ดี ไอ เอส ของเครื่องคาดาพยห์จะเป็นไปอย่างอัตโนมัติ คำสั่งใดก็ตามที่มีการบันทึกเอนัตร์องบนจานแม่เหล็ก เอนัตร์นั้นจะถูกบันทึกลงในแฟ้มข้อมูลที่มีอยู่แล้ว (Existing File) หรือถ้าหากแฟ้มข้อมูลนั้นยังไม่มีการบันทึกอยู่ในจานแม่เหล็ก ดี ไอ เอส ก็จะทำการสร้างแฟ้มข้อมูลขึ้นมาใหม่เพื่อใช้เก็บเอนัตร์หรือข้อมูลชุดนั้น แฟ้มข้อมูลที่ถูกสร้างขึ้นใหม่จะถูกสร้างขึ้นบนไดรว์ที่ผู้ใช้กำหนดหรือจะให้ ดี ไอ เอส ทำการกำหนดให้ก็ได้ ถ้าให้ ดี ไอ เอส กำหนดให้แฟ้มข้อมูลจะถูกสร้างขึ้นบนไดรว์ใดก็ได้ที่มีที่ว่าง โดยการค้นหาที่ว่างจะเริ่มจากไดรว์ 0 เป็นต้นไป ที่ว่างนี้จะหมายถึงที่ว่างที่มีอยู่ในสารบัญ (Directory) ของไดรว์นั้น ซึ่งสามารถที่จะบรรจุชื่อของแฟ้มข้อมูลใหม่ได้ และอย่างน้อยที่สุดจะต้องมีที่ว่าง 1 คลัสเตอร์อยู่บนจานแม่เหล็กไดรว์นั้นด้วย คลัสเตอร์คือหน่วยของเนื้อที่ที่เล็กที่สุดที่สามารถจัดสรรให้แก่แฟ้มข้อมูล

2.3.3 การลบแฟ้มข้อมูล (File Dellion) ผู้ใช้ต้องเป็นผู้กำหนดให้ ดี ไอ เอส ทำการลบแฟ้มข้อมูลที่ไม่ต้องการทิ้งโดยใช้คำสั่ง "Kill" การลบแฟ้มข้อมูลทิ้งเป็นการลบแบบโลจิกคัลคือ จะทำการลบชื่อแฟ้มข้อมูลที่ไม่ต้องการออกจากสารบัญของจานแม่เหล็ก เนื้อที่ของจานแม่เหล็กที่แฟ้มข้อมูลนั้นใช้อยู่ แฟ้มข้อมูลอื่นสามารถที่จะนำไปใช้ได้

2.3.4 การป้องกันแฟ้มข้อมูล(File Protection) แฟ้มข้อมูลในระบบ ดี โอ เอส สามารถมีการป้องกันได้ 3 แบบ ดังนี้

2.3.4.1 การป้องกันการบันทึกซ้ำ(Write Protection) แฟ้มข้อมูลที่มีการป้องกันแบบนี้จะไม่สามารถทำการบันทึกซ้ำหรือลบทิ้งได้

2.3.4.2 การป้องกันการลบทิ้ง(Delete Protection) แฟ้มข้อมูลที่มีการป้องกันแบบนี้จะไม่สามารถทำการลบทิ้งได้แต่สามารถทำการบันทึกซ้ำได้

2.3.4.3 ไม่มีการป้องกัน(No Protection) แฟ้มข้อมูลที่ไม่มีการป้องกันสามารถที่จะลบทิ้งหรือบันทึกซ้ำได้

2.4 ลักษณะแฟ้มข้อมูลแบบอัด(Compressed File)

รูปแบบของแฟ้มข้อมูลแบบอัดเป็นลักษณะพิเศษแบบหนึ่งของรูปแบบแฟ้มข้อมูลแบบข้อความ (Text File) ในระบบ ดี โอ เอส ของเครื่องคาดาพอยต์ รูปแบบของแฟ้มข้อมูลแบบข้อความประกอบขึ้นด้วยตัวอักษรแอสกี(ASCII Characters) และตัวอักษรที่ใช้สำหรับควบคุม(Control Characters) ตัวอักษรที่ใช้สำหรับควบคุมมีค่าอยู่ระหว่าง 000 ฐานแปด และ 037 ฐานแปด แต่ละฟิสิกส์คัล เรคคอร์ด(Physical Record) ของแฟ้มข้อมูลแบบข้อความประกอบขึ้นด้วยตัวอักษร 256ตัว หรืออีกนัยหนึ่งก็คือ หนึ่งโลจิกคัล ดิส เซกเตอร์(Logical Disk Sector) นั่นเอง ตัวอักษร 3 ตัวแรก และ 2 ตัวสุดท้ายถูกสำรองไว้ใช้สำหรับทำหน้าที่ในการควบคุม ดังนั้นจึงเหลือเนื้อที่สำหรับใช้บันทึกข้อมูลอีก 251 ไบท์ รูปแบบของแฟ้มข้อมูลแบบข้อความมีดังต่อไปนี้

ออฟเซท(Offset)	ความยาว(Length)	รายละเอียด(Description)
000	001	ใช้บันทึกค่าของ พี เอฟ เอ็น
001	002	ใช้บันทึกค่าของ แอล อาร์ เอ็น
003	0373	ใช้บันทึกข้อมูล
0376	002	สำรอง(Reserves)

ส่วนที่ใช้สำหรับบันทึกข้อมูล สามารถบันทึกข้ามขอบเขตของเซกเตอร์(Sector Boundary) ได้ เสมือนว่าข้อมูลเหล่านั้นได้ถูกบันทึกติดต่อกัน ทั้งนี้โดยการอาศัยตัวอักษรที่ใช้ในการควบคุม ซึ่งแทรกอยู่ภายในตัวข้อมูลเหล่านั้น ตัวอักษรที่ใช้ในการควบคุมที่จำเป็นมีอยู่ 5 ตัว ดังนี้

000 <NUL>	ใช้แสดงจุดสิ้นสุดของแฟ้มข้อมูล
003 <EM>	ใช้แสดงจุดสิ้นสุดของเซกเตอร์
011 <CMP>	ใช้แสดง Space Compression
015 <ENT>	ใช้แสดงจุดสิ้นสุดของโลจิกคัล เรคคอร์ด
032 <DEL>	ใช้แสดงข้อมูลที่ถูกลบทิ้ง (Deleted Data)

พื้นฐานในการบันทึกแฟ้มข้อมูลแบบนี้ก็คือ โลจิกคัล เรคคอร์ด, โลจิกคัล เรคคอร์ดจะถูกบันทึกติดต่อกันไปเรื่อย ๆ โดยแต่ละโลจิกคัล เรคคอร์ดถูกแบ่งออกจากกันด้วยตัวอักษรที่ใช้ในการควบคุม <ENT> ถ้าในโลจิกคัล เรคคอร์ดมีตัวอักษร <CMP> ปรากฏอยู่ แสดงว่ามีการอัดที่ว่าง (Space - Compressed) เกิดขึ้น ตัวอักษรที่อยู่ถัดมาจากตัวอักษร <CMP> ใช้เป็นตัวนับที่ว่าง (Space Count) แสดงจำนวนที่ว่างที่ถูกแทน (Removed) ถ้าจำนวนโลจิกคัล เรคคอร์ดในแต่ละฟิสิกส์คัล เรคคอร์ดมีจำนวนเท่ากันตลอดทุก ๆ ฟิสิกส์คัล เรคคอร์ด เรียกลักษณะแฟ้มข้อมูลแบบนี้ว่า บลอก (Block) ถ้าหากจำนวนโลจิกคัล เรคคอร์ด ในแต่ละฟิสิกส์คัล เรคคอร์ด มีจำนวนไม่แน่นอน เรียกลักษณะแฟ้มข้อมูลแบบนี้ว่า เรคคอร์ดคอมเพรส (Record Compressed) หรือแฟ้มข้อมูลแบบอัด (Compression File)

จากข้อกำหนดของ ดี โอ เอส ลักษณะรูปแบบของเครื่องหมายที่แสดงการสิ้นสุดของแฟ้มข้อมูล (End of File Mark) มีลักษณะดังนี้

000 000 000 000 000 000 003

เครื่องหมายนี้ต้องเริ่มต้นที่ตำแหน่ง 3 ถึง 9 ของเซกเตอร์ (ตำแหน่งที่ 0 ถึง 2 ใช้บันทึกค่าของ พี เอฟ เอ็น และค่าของ แอล อาร์ เอน)