

# 5-1

## 8인치와 5인치의 파일 변환

### TF20N으로 파일변환/BIOS의 변경/사용소감

퍼스널 컴퓨터의 파일장치에는, 일반적으로 5 인치의 디스크 드라이브가 사용되고 있는데, 예전의 CP/M머신은 대부분이 8인치 편면 단밀도 디스크 드라이브가 사용되고 있었다.

그 때문에 연구실에 따라서는 8 인치의 파일과 5 인치의 파일이 혼재하여 상당히 어려움을 겪

고 있는 것을 흔히 보게 된다. 그 중에는, 시리얼회로를 써서 8인치 기계와 5인치 기계와의 파일전송을 행하고 있는 예도 있고 시판되는 퍼스널 컴퓨터 중에서 8 인치와 5 인치의 디스크 드라이브 양쪽을 접속할 수 있는 것을 구입하여 파일변환장치로 사용하고 있는 예가 있을 정도이다.

것으로 되어 있기 때문이다. 이것을 거꾸로 생각하면 PPI 경유로 디스크장치 PC8031-2W를 콘트롤할 수 있게 된다.

이번에 TF20N(PC8031-2W와 콤팩티블의 제품 업체사)을 사용해 보는 기회가 있었으므로, 이

표 5-1 M001과 TF20N와의 접속

M001측		TF20N측	
핀번호	신호명	방향	핀번호
1	Reset(NC)	→	34
3	PC <sub>0</sub>	(未使用)	32
5	PC <sub>1</sub>	(DAC)	30
7	PC <sub>2</sub>	(RFD)	28
9	PC <sub>3</sub>	(DAV)	26
11	PC <sub>7</sub>	(ATN)	24
13	PC <sub>6</sub>	(DAC)	22
15	PC <sub>5</sub>	(RFD)	20
17	PC <sub>4</sub>	(DAV)	18
19	PA <sub>7</sub>	←	16
21	PA <sub>6</sub>	←	14
23	PA <sub>5</sub>	←	12
25	PA <sub>4</sub>	←	10
27	PA <sub>3</sub>	←	8
29	PA <sub>2</sub>	←	6
31	PA <sub>1</sub>	←	4
33	PA <sub>0</sub>	←	2
2	GND		33
18	GND		17
20	PB <sub>7</sub>	→	15
22	PB <sub>6</sub>	→	13
24	PB <sub>5</sub>	→	11
26	PB <sub>4</sub>	→	9
28	PB <sub>3</sub>	→	7
30	PB <sub>2</sub>	→	5
32	PB <sub>1</sub>	→	3
34	PB <sub>0</sub>	→	1

## TF20N으로 파일변경

이 시스템에도 8인치와 5 인치를 양쪽 동시에 접속할 수 없는가 여러 가지로 생각한 끝에, 인텔리전트 플로피 디스크장치 PC8031-2W이 있으면 접속할 수 있다는 것을 알게 되었다.

그것은 2-1장에서 소개한 Z80 싱글보드 컴퓨터 M001은 장착 PC8001의 디스크장치로 사용할 것을 생각해 보았기 때문인데, PC8001과 이 시스템 M001을 접속하는 PPI의 핀배치도 그것에 가까운

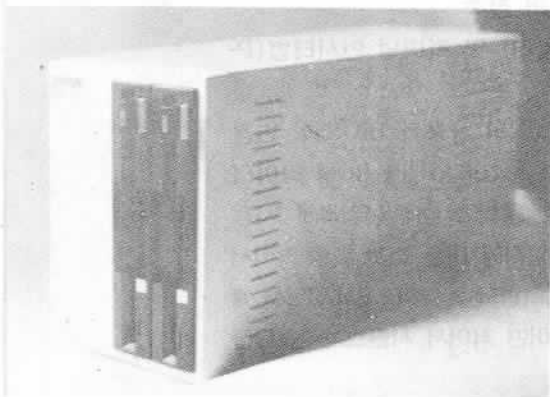
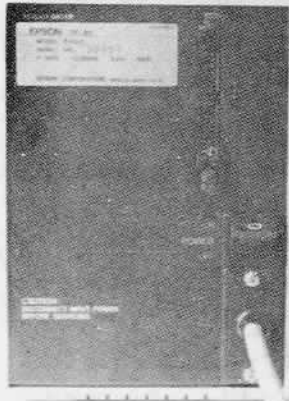


사진 5-1 여기서 사용한 미니 플로피 디스크 장치 TF20N (PC8001, 8801용)



· 사진 5-2 TF20N의 뒷면

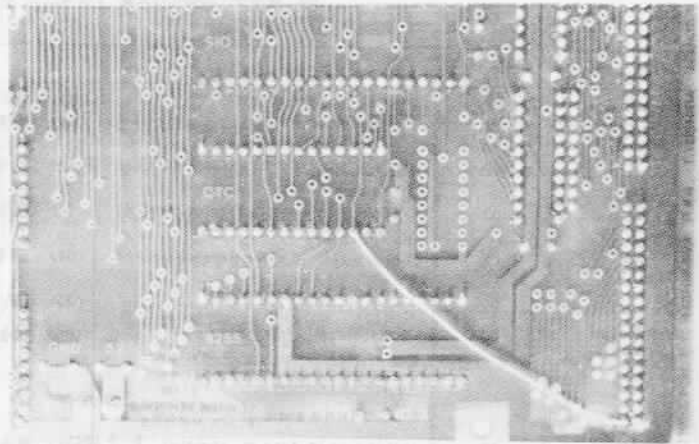


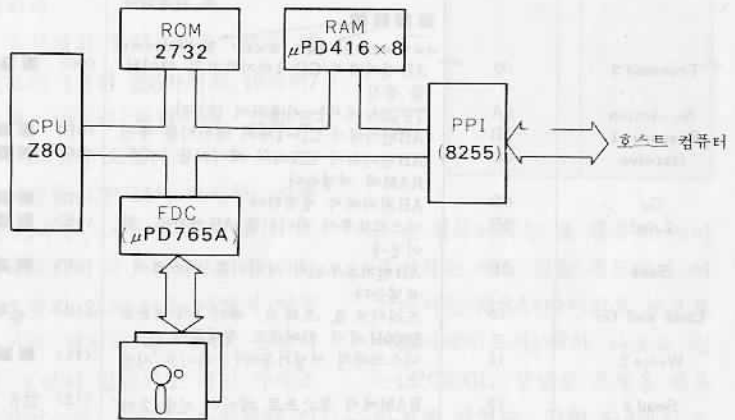
사진 5-3 리셋의 점퍼 부근(Reset 신호를 CTC의 17번핀에서 잡고, PPI 코넥터의 NC(1번핀)에 접속)

아이디어를 실현해볼 수 있었다.

먼저 물리적으로 TF20N과 Z80 싱글보드 컴퓨터 M001을 접속한다. 이것은 표 5-1에 보인 것과 같은 접속과 PPI의 NC핀에 Reset 신호를 내는 점퍼처리를 행한다. 이 리셋신호는 Z80 싱글보드 컴퓨터 M001에 리셋을 걸었을 때, 동시에 TF20N의 내부회로도 리셋하기 위해 필요하게 된다. 물리적인 접속은 이것으로 끝난다.

다음에 TF20N의 콘트롤방법에 대해서 설명한다. TF20N은 Z80 CPU 내장형의 인텔리전트한 디스크장치(그림 5-1)로 코멘트를 가하는 것만으로 디스크의 리드/

그림 5-1 TF20N의 구성



라이트처리를 하게 할 수 있다.

코멘드나 데이터의 전송타이밍은 그림 5-2에 보인 것과 같은 GPIB식인 핸드셰이킹을 행한다.

여기서는 이 타이밍을 PPI의 포트의 조작에 의해 행한다. TF20N에 가할 수 있는 코멘트는 표 5-2에 보인 것과 같은 33종류가

그림 5-2 데이터 전송

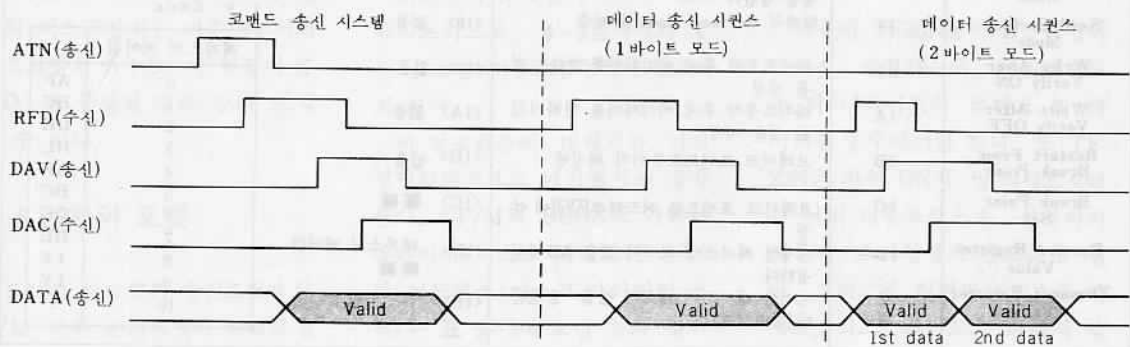


표 5-2 TF20N의 코맨드표

[N] : 블록수 (256바이트를 1 블록으로 한다)

[T] : 트랙번호 (0~79)

[D] : 드라이브번호 (0~2)

[S] : 섹터번호 (1~16)

코맨드명	코맨드 번호	기능	파라미터의 주는법
Initialize	00	FDC의 초기설정을 한다	<00> 없음
Write	01	디스크에의 새넣기동작(1바이트 시퀀스)	<01> <b>N D T S DATA</b>
Read	02	디스크에서 내부의 RAM에 데이터를 읽어낸다	<02> <b>N D T S</b>
Send Data	03	RAM에서 호스트로 데이터전송(1바이트 시퀀스)	<03> 없음
Copy	04	드라이브간의 코피를 실행한다	<04> <b>N D T S D T S</b> 소스      타스 타미션
Format	05	디스켓의 포맷을 실행한다	<05> <b>D</b>
Send Result Status	06	스테이터스를 전송한다	<06> 없음
		<div> <div>MSB</div> <div>1 0 0 0 0 0</div> <div>LSB</div> </div> <div> <div>1: 데이터 어레이를</div> <div>0: 노드 어레이를</div> </div> <div> <div>1: 섹터</div> <div>0: OK</div> </div>	
Send Drive Status	07	드라이브 스테이터스를 전송한다	<07> 없음
		<div> <div>MSB</div> <div>0 0 1 1 1 1</div> <div>Drive 1 Drive 0</div> </div> <div> <div>1: 동작가</div> <div>0: 동작불가</div> </div>	
RAM Test	08	내부의 RAM의 테스트를 실행한 결과를 되돌린다	<08> 없음
Transmit 2	09	내부의 어드레스 세팅은 데이터 읽어내기 데이터 AD번지에서 CD-1번지까지의 데이터를 송신	<09> <b>A B C D</b>
No Action	0A	TF20N에서는 사용되지 않는다.	<0B> <b>A B C D</b>
Transmit 1	0B	AB번지에서 CD-1개의 데이터를 송신	<0C> <b>A B C D DATA</b>
Receive	0C	AB번지위서 CD개의 데이터를 내부 RAM에 새넣는다	
Go	0D	AB번지에서 실행한다	<0D> <b>A B</b>
Load	0E	디스크로부터 데이터를 AB번지에 읽어낸다	<0E> <b>N D T S A B</b>
Save	0F	AB번지로부터의 데이터를 디스크에 새넣는다	<0F> <b>N D T S A B</b>
Load and Go	10	드라이브 0, 트랙 0, 섹터 1의 내용을 5000H에서 읽어내고 실행한다	<10> 없음
Write 2	11	디스크에의 새넣기동작(2바이트 시퀀스)	<11> <b>N D T S DATA</b>
Send 2	12	RAM에서 호스트로 데이터 전송(2바이트 시퀀스)	<12> 없음
Send FDC Status	13	FDC(μPD765)의 상세한 스테이터스를 송신. 스테이터스는 8바이트 송출	<13> 없음
Send FDC Sens Device	14	FDC가 지정한 드라이브를 센스하여 현재의 스테이터스(1바이트)를 송신	<div> <div>1바이트</div> <div> <div>X X X X X</div> <div>HD ↑</div> <div>US1 US0</div> </div> <div>HD 1클럭 제드번호</div> <div>US0</div> <div>US1 유닛드번호</div> </div>
Status N Transmit 1	15	AB번지에서 CD-1개의 데이터를 2바이트 시퀀스를 사용하여 송신	<15> <b>A B C D</b>
N Receive	16	AB번지에서 CD개의 데이터를 2바이트 시퀀스를 만들어 전송하여 내부RAM에 새넣는다. 켄	<16> <b>A B C D DATA</b>
Set Surface Mode	17	드라이브마다의 양면/전면 사용의 지정을 행한다	<div> <div>0: 전면</div> <div>1: 양면</div> </div> <div> <div>X X X X X X X</div> <div>1 0 드라이브</div> </div>
Send Surface Mode	18	현재의 Surface Mode를 전송	<18> 없음
Write After Verify ON	19	라이트동작 후에 베리파이를 행하는 것을 지정	<19> 없음
Write After Verify OFF	1A	라이트동작 후에 베리파이를 행하지 않는 것을 지정	<1A> 없음
Restart From Break Point	1B	브레이크 포인트로부터의 재실행	<1B> 없음
Break Point Set	1C	브레이크 포인트를 어드레스 [A][B]에 설정	<1C> <b>A B</b>
Receive Register Value	1D	지정한 레지스터 제어의 값을 AB에 설정한다	<1D> 레지스터 제어값 <b>A B</b>
Transmit Register Value	1E	지정한 레지스터 제어의 값을 송신한다(2바이트)	<1E> 레지스터 제어값

레지스터 제어값	
0	AF
1	BC
2	DE
3	HL
4	AF'
5	BC'
6	DE'
7	HL'
8	IX
9	IY
10	I

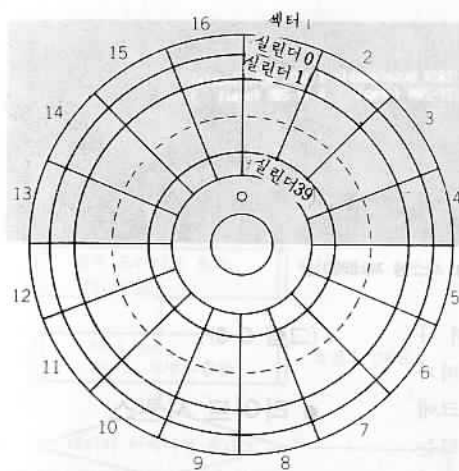


그림 5-3 5인치 디스켓의 물리 포맷

1 섹터당 256바이트 사이드 0  
사이드 1과 같다 포맷

그림 5-4  
PC8001 CP/M 5인치  
양면 롱포맷(MSA)

실린더	사이드 0	사이드 1
0	부트/시스템 트랙	미사용
1	시스템 트랙	미사용
2	디렉토리	CP/M 데이터 트랙
...		
39	CP/M 데이터 트랙	CP/M 데이터 트랙

준비되어 있다.

## ● 소프트웨어의 설계

TF20N에 준비되어 있는 각종 코맨드를 사용하여 파일전송을 행하는 소프트웨어에 대하여 설명한다.

파일전송용 소프트웨어에는 2개의 방법이 생각된다. 하나는 전용 파일전송용 프로그램을 만들고, 이것을 COM파일로서 실행하는 방법, 다른 하나는 BIOS 내부에 TF20N용의 드라이버를 조립하는 방법이다. 여기에서는 파일전송 외에 각종 유틸리티를 5인치 디스크쪽에서 움직여 보고 싶었으므로, 나중 방법을 채용하기로 하였다. 따라서 접속한 5인치의 디스크장치는 기존 8인치의 디스크장치 A:B:에 덧붙여 C:D:가 증설된 것과 같이 조작할 수 있다.

## ● 5인치의 포맷

5인치 디스켓의 물리포맷과 CP/M 상의 논리포맷에 대해서 설

명한다.

5인치의 물리포맷은 양면 배밀도의 1섹터 256바이트, 16섹터/트랙, 40트랙/편면이다. 그림 5-3에 그것을 보였다.

이것을 CP/M의 파일로 사용할 경우는, 시스템 트랙의 위치, 디렉토리의 수, 디렉토리의 위치, 디스켓의 앞과 뒤를 어떻게 다루는가의 결정이 필요하다. 표준이 될 포맷이 없는가도 여러 가지 조사하였는데, 어쩔지 통일되어 있지 않은 것 같다. 같은 제품인 퍼

스널 컴퓨터에는 몇 종류의 것이 존재하는 예도 있을 정도이다. 여기에서는 MSA(마이크로 소프트웨어소사이어티사)에서 나오고 있는 PC8001, 양면용 포맷을 채용하기로 하였다. 그림 5-4에 이 포맷의 사방을 보였다.

## ■ BIOS의 변경

디스켓의 포맷은 1섹터가 256바이트이므로, 4-3장에서와 같은 블로킹/디블로킹동작이 필요하게 된다.

이 알고리즘에 대해서는 이미 설명하였으므로 여기에서는 생략한다. CP/M의 BDOS에 가하는 파라미터(디스크 파라미터 블록) 및 어드레스(디스크 파라미터 헤더)는 표 5-3에 보인 것과 같이

된다.

다음에 TF20N용의 I/O 방법에 대해서 설명하는데, 기본적으로는 리드/라이트의 코맨드 및 데이터를 송수하기만 한다. 또 TF20N은 파워 ON시 및 리세츠시에 어떤 디폴트값으로 설정되기 때문에 초기설정이 필요하게 된다. 그러므로 일련의 시퀀스를 리세트, 리드, 라이트로 나뉘어서 설

## ● 리셋 시퀀스

먼저 M001의 PPI를 이니셜라이즈한 다음, TF20N에 리셋 코멘드를 가한다. 다음에 센드 드라이브 스테이더스 코멘드(코멘드 4)를 보내고, 현재 액티브한 디스크 드라이브의 수를 체크한다.

액티브한 드라이브가 만일 존재하면, 다음에 드라이브의 양면을 사용하는 것을 지정하는 코멘드(코멘드 23)를 가한다. 이들의 일련의 시퀀스는 BIOS의 BOOT(콜드 부트) 루틴에 있어서 1회만 처리된다(그림 5-5).

## ● 리드 시퀀스

리드 데이터 코멘드(코멘드 2)에 읽어내기 블록수, 드라이브번호, 트랙번호, 섹터번호를 붙여서 TF20N으로 보낸다. 블록수는 1블록(256 바이트)으로 고정한다.

이 코멘드로 지정섹터가 TF

20N의 내부 RAM에 읽여져 나오기 때문에, 이 디스크처리가 정상적으로 행해지고 디스크에서 읽어낸 데이터가 존재하는가 어떤가를 조사하기 위해, 센드 리서트 스테이더스 코멘드(코멘드 6)를 보낸다. 받아들인 스테이더스 중에서, 데이터 어레이블 플래그(비트 6)가 액티브로 되어 있는 것을 확인한다. 다음, 센드 데이터 코멘드(코멘드 18)를 보내어 데이터를 받아들이는다.

이 때 보내오는 데이터의 사이즈는 리드 데이터 코멘드로 지정한 것이다. 데이터를 완전하게 받아들인 다음, 다시 센드 리서트 스테이더스 코멘드를 보내어 에러의 유무를 확인한다.

(그림 5-6).

## ● 라이트 시퀀스

라이트 코멘드(코멘드 17)의 다음에, 써넣기 블록수, 드라이브번호, 트랙번호, 섹터번호, 써넣기 데이터(블록수×256 바이트)를 연속적으로 TF20N에 보낸다. 이것으로 데이터가 디스켓에 써넣어지므로, 그 후에 센드 리서트 스테이더스 코멘드를 보내어 받아들인 스테이더스를 체크하여 에러의 유무를 확인한다(그림 5-7).

표 5-3 TF20N용의 디스크 파라미터 헤더 및 디스크 파라미터 블록

DPBASEC:				
DW	0,0,0,0	;FOR DRIVE C:TF-20N FORMAT(256)		
DW	DIRBF,DPBLKC			
DW	CHK02,ALLO2			
DPBASED:				
DW	0,0,0,0	;FOR DRIVE D:TF-20N FORMAT(256)		
DW	DIRBF,DPBLKD			
DW	CHK03,ALLO3			

(a) TF20N의 디스크 파라미터 헤더

DPBLKC:				
DW	64	;FOR DRIVE C:TF-20N FORMAT		
DB	4	;CP/M SEC/TRK (SPT)		
DB	15	;BLOCK=2048 (BSH)		
DB	1	;128 BYTE/SEC (BLM)		
DB	151	;EXM)		
DW	127	;256*32*38/2048-1(DSM)		
DB	192	;NO. OF DIR (DRM)		
DB	0	;1 BLOCK FOR DIR (ALO)		
DW	32	;ALL)		
DW	2	;CHECK=(DIR+1)/4 (CKS)		
DPBLKD	DPBLKC	;TRACK OFFSET (OFF)		
		;FOR DRIVE D:TF-20N FORMAT		

(b) TF20N의 디스크 파라미터 블록

사진 5-4 시스템 제네레이션

그림 5-5 리셋 시퀀스

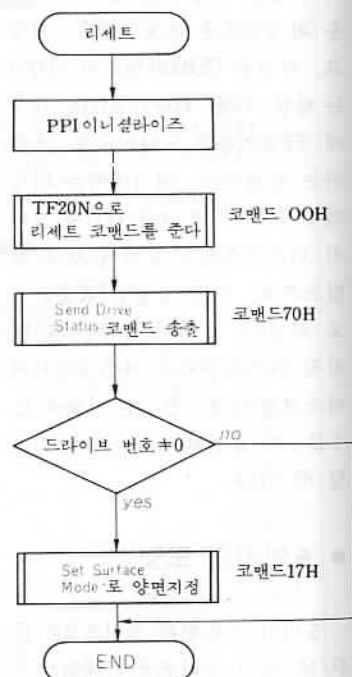


그림 5-6 리드 시퀀스

그림 5-7 라이트 시퀀스

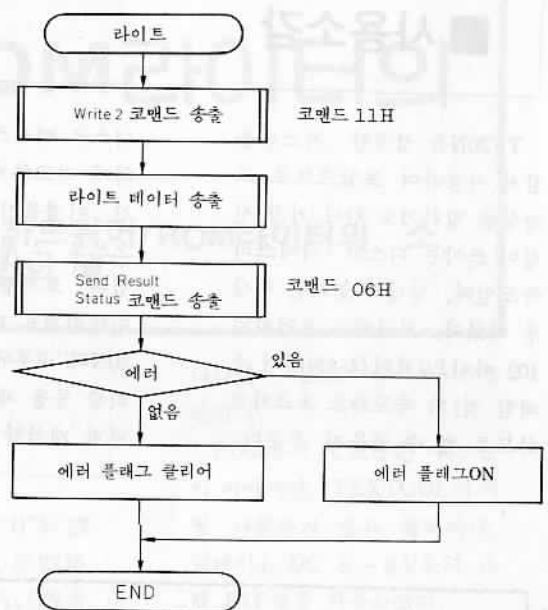
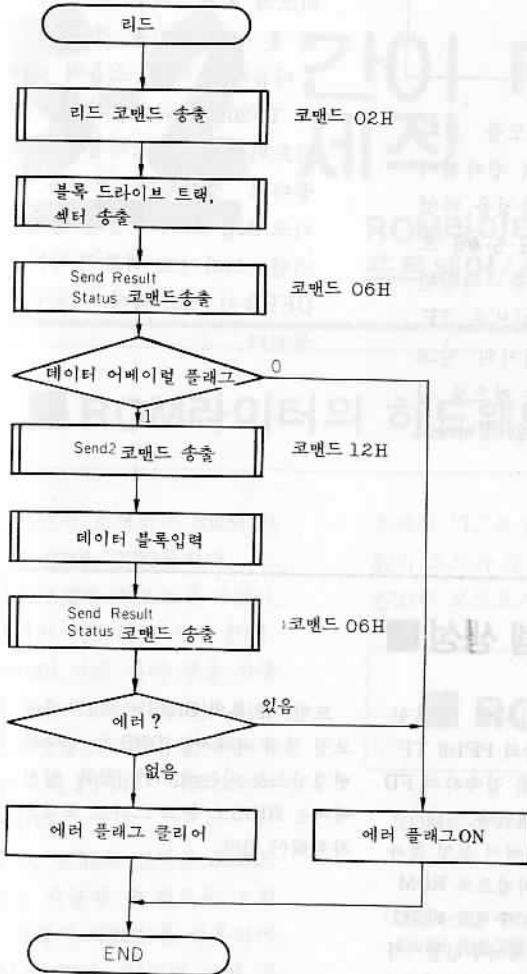


표 5-4 디스크 액세스 시간의 비교

단위 : sec

디스크	마이크로컴퓨터 (MAC) 로 출력파일이 있는 때 주1)	마이크로컴퓨터 (MAC) 로 출력파일이 있는 때 주2)	C컴파일러 (BDSC) 의 컴파일시간 주3)	C의 정크시간 주4)
TF20N	187	579	94	46
8인치 양면배럴 1024바이트/섹터	93	166	43	14
8인치 양면배럴 256바이트/섹터	110	230	47	17
8인치 전면단일 128바이트/섹터	129	253	60	25

주

1) 58K의 소스 파일을 MAC로 어셈블한 것(출력파일 없음)

2) 58K의 소스 파일을 MAC에서 어셈블한 것 출력파일은

HEX파일 26K

PRN파일 146K

SYM파일 6K

3) 24K의 C의 소스 파일과 헤더 파일(8K)을 BDSC 컴파일러(5.4장)에서 컴파일하여 12K의 CRL 파일(릴로케이터를 파일)을 출력

4) 주3)의 CRL 파일과 라이브러리를 링크하여 16K의 COM파일을 출력

이상 간단하게 콘트롤방법을 설명하였다. 이것을 실제로 BIOS 속에 조립한 것이 리스트 5.1이다. 이 BIOS는 8인치의 A드라이브, B드라이브가 4·4장에서 소개한 양면 배럴도 102바이트/섹터의 포맷이며, C, D드라이브가 TF20N의 드라이브 0과 1에 할당된 62K CP/M용이다.

따라서, 시스템 제네레이션은 4-4장과 같은 방법으로 할 수 있다. 또 참고로 TF20N을 사용하여 포맷을 행하는 프로그램은 리스트 5.2에 보였다.



## ■ 사용소감

TF20N을 접속한 시스템을 잠시 사용하여 보았으므로 그 감상을 말하기로 한다. 가장 신경이 쓰이는 디스크 액세스의 속도인데, 상당히 늦다는 인상을 받았다. 최근에는 8인치의 102 바이트/섹터(4.44에서 소개한 것)의 속도라도 초조하게 하므로 별 수 없을지 모른다.

디스크 액세스의 속도를 정확하게 비교하기 위해 컴파일시간, 어셈블시간의 측정을 하였으므로 그 결과를 표 5.4에 보았다. 모처럼 CPU를 내장한 인텔리전트 타이프이므로, TF20N의 내부에서 캐시의 알고리즘 등을 채용하는 것으로, 상당히 개선할 수 있을 것이라고

생각된다. 또 포트간의 인터페이스의 시퀀스에도 다시 생각해 볼 여지가 있을 것 같다.

다음에 조작성, 소음에 대해서 TF20N의 디스켓의 세트, 이젝트의 기능의 조작성은 아주 좋다고 생각된다. 모터음, 시이크 스텝 모터의 음은 상관없지만, 드라이브 섹트의 ON/OFF음이 다소 귀에 거슬리게 들린다.

## ■ 5인치용 CP/M으로부터의 시스템 생성

이 장의 예와 같이 8인치와 5인치의 드라이브가 양쪽 접속할 수 있으면, 5인치 시스템용의 CP/M을 구하여 상충하는 것도 가능하게 된다. NEC나 카시소 계산기 등 디지털 리서치사와 OEM계약을 체결하여 나오고 있는 CP/M로써의 기능은 같기 때문에 이용할 수 있다. 시스템의 기동에는 다음 두 가지 경우를 생각할 수 있다.

① 8인치와 5인치가 접속되어 이미 변환용 CP/M(본문 참조)이 동작하고 있는 경우.

이 경우는 PIP 등을 사용하여 파일 모듈을 전송하여 제4부에서 보인 방법에 기초로 하여 행한다.

② 변환용 CP/M을 이용할 수 있을 때.

이것은 하드웨어적으로 8인치와 5인치, 다시 5인치만이 접속되어 있는데, CP/M의 서포트를 받아들이 수 없는 경우. 이 때의 모든 경우를 생각하면, 그림 5-A와 같이 될 것으로 생각된다. (1)은 본문에서 설명한 8인치와 5인치의 접속법, (2)는 보

드의 FDC에 직접 5인치를 접속한 경우, (3)은 보드 상의 PPI에 TF20N 타이프의 5인치를 접속하여 FDC는 사용하지 않는 예이다.

(2)의 경우는 제4부에서 보인 것과 같은 제1단계의 패치법으로 ROM 프로그램만으로 행할 수 있는데, (3)에서는 TF20N의 제어루틴을 자기가 그려 ROM 등에 고정시켜 섹터 리드/라이트로 행하여 패치를 진행하게 된다.

또한, (2)용의 BIOS는 제4부에서 보인 것에 대해서는 DPB 등 상수의 변경만으로 되는데, (1), (3)의 방법에서는 BIOS도 쿨드 스타트 로더도 자작해야 한다.

