

M10CN 交換機 通話量 데이터 On-line 伝送装置 開發

閔庚善 · 洪起秀 · 俞樂均 / 運用技術室

I. 序 論

本 研究에서는 M10CN 交換機 通話量 데이터 on-line 伝送方案으로 現在까지 開發된 伝送装置 (DTIP : Data Transfer Interface Processor)의 시스템 構成 및 하드웨어와 소프트웨어의 機能을 설명하고, 1602 시스템과의 連結을 위한 connector 製作에 대해 略述한다.

II. DTIP 시스템 概要

1. 用途 및 目的

DTIP는 M10CN의 MTC(Magnetic tape controller)와 Kennedy 9700 MTU(Magnetic tape unit)사이에서 write되는 데이터를 buffering하여 다른 computer로 伝送하는 interface용으로 MTC의 I/O signal timing에 맞추어 병렬 입력 데이터를 memory에 일단 저장한 후, 지정 시간 이내에 on-line network을 통하여 직렬 전송하는데 사용한다. DTIP의 주요 목적은 M10CN 交換機에서 측정된 통화량 데이터를 트래픽 집중관리 시스템의 host computer로 transfer하는 것이다.

2. 設計 指針

DTIP의 설계 지침은 다음과 같다.

- PCB의 크기는 350mm×235mm로 하며 2장으로 구성
- Power는 전화국 전원 -48V를 DC / DC converter에서 공급
- Board간 연결은 72pin edge connector (front측) 2개 사용
- TTC, modem, console I/O는 각각 RS-232 EIA connector 사용
- 4800 baud rate이내의 전송 속도 변경 스위치 자체내 실장
- Memory는 16K ROM과 48K DRAM으로 구성
- Serial 및 데이터 bus의 board간 communication은 74LS24X의 bus transceiver로 구성
- MTC측 connector 연결은 twisted pair cable을 사용하고 impedance matching을 위한 2개의 저항으로 종단

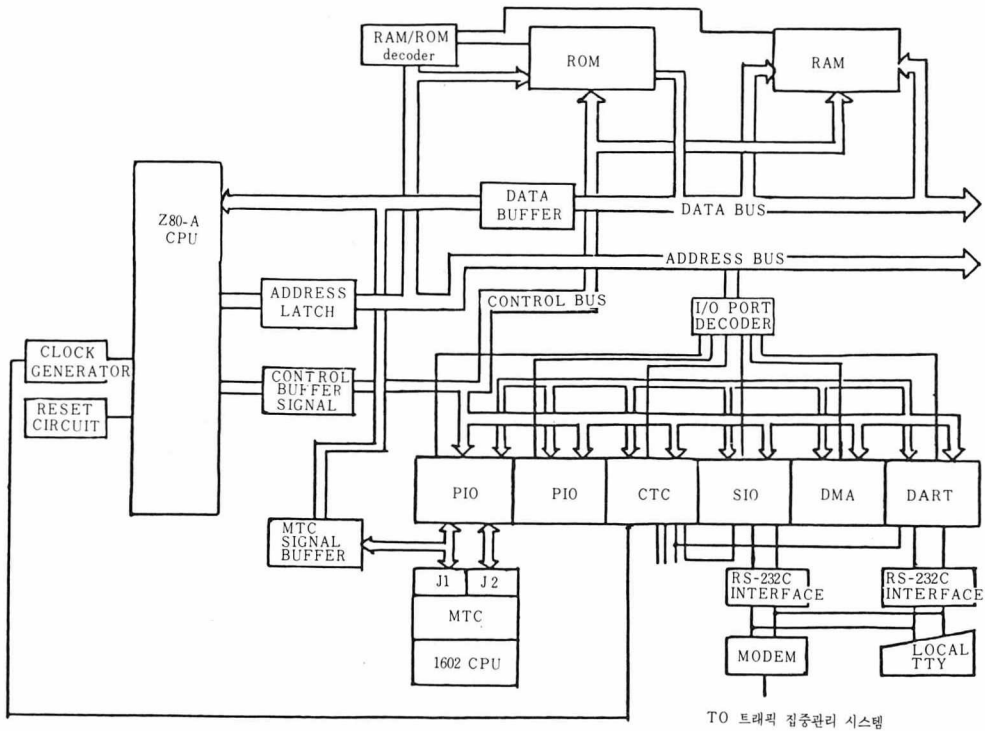
III. DTIP 하드웨어 設計 概要

1. 시스템 構成

DTIP시스템은 CPU, memory 및 decoder로 구성되어 시스템 전체 동작의 timing을 제어하는 제어부와 PIO, SIO, DART등 각 I/O port 및 buffer로 구성되어 1602 MTC와의 정합 및 병렬 출력 데이터를 수신하고, 집중관리 시스템

으로 직렬 전송 데이터를 보내거나 집중관리 시스템으로부터 데이터를 받는 기능을 하는 I/O 부, 그리고 power driver, bus transceiver와 같은 기타 logic으로 설계한다.

DTIP의 시스템 구성도는 다음과 같다.



TO 트랙픽 집중관리 시스템

〈그림 1〉 DTIP System Block Diagram

2. CPU, Memory 및 Decoder 設計

CPU는 8bit Z-80A processor를 사용하며, system clock은 8MHz oscillator와 counter (74 LS 74)를 사용한 分周回路를 거쳐 4MHz 를 공급하고 I/O port는 decoder (74 LS 139)를 통해 각 chip이 enable되게 한다. 또한 reset회로를 통하여 수동적으로 시스템을 reset 시킬 수 있도록 한다. Field test를 위해 각 module 별로 회로를 설계하여 module별 시험을 하며, power supply, modem과 함께 이를 실장할 수 있는 rack을 제작하여 사용한다. 이때 각 기관의 연결은 system control signal 및 data driver를 위해 bus transceiver (74 LS 24 X)를 사용한다. Memory는 ROM의 경우 EPROM (273 2) 4개를 사용하며, RAM은 시험용으로 static

RAM (6116) 16개를 사용하여 32K로 구성하고, PCB 제작시에는 dynamic RAM으로 구성한다. (Memory map의 구성은 <表 1>과 같다)

3. I/O Port의 設計 및 機能

가. PIO 및 status buffer

PIO는 1602 MTC로 부터 입력되는 데이터를 수신하여 memory로 load 한다. 즉, MTC로 부터의 $2\mu\text{sec}$ 의 WDS 신호를 PIO B-port의 \overline{B} STB에 연결하여 WDS 신호가 입력되면 \overline{INT} 가 active 되게 한다. Interrupt가 感知되면 CPU는 데이터를 수신하고 이 데이터를 memory로 load한다. PIO A-port의 \overline{ASTB} 는 MTC로 부터의 SRC 신호를 연결하여 error가 발생할 경우 interrupt를 일으켜 데이터의 수신을 중단하

0000H	DTIP System monitor (16K)
3FFFH	
4000H	DTIP Stack area (256 byte)
40FFH	
4100H	DTIP System RAM (768 byte)
43FFH	
4400H	Spare
47FFH	
4800H	DATA Buffer Area (14K)
7FFFH	
8000H	Spare Data Buffer area (16K)
BFFFH	

〈表 1〉 Memory Map

게 하고 error를 처리하게 한다.

또한 MTC로 부터의 다른 신호들(SEL, SFC, SWS, WARS등)은 status buffer로 수신하여 CPU에 의해 각 status를 check 하도록 한다.

나. SIO, DMA의 機能

SIO는 PIO에 의해 memory buffer로 load된 data를 집중관리 시스템으로 전송하고 집중관리 시스템에서 출력된 입력 command를 수신한다. 데이터의 전송시 DMA와 함께 동작하여 bl-

ock 단위 전송을 가능하게 하고, duplex asynchronous 방식으로 송·수신 하도록 SIO, DMA를 set시킨다. 또한 RS 232C interface 회로에서 modem을 제어할 수 있도록 하며, ±13V power driver를 만들어 1488, 1489로 공급한다.

다. CTC, DART의 機能

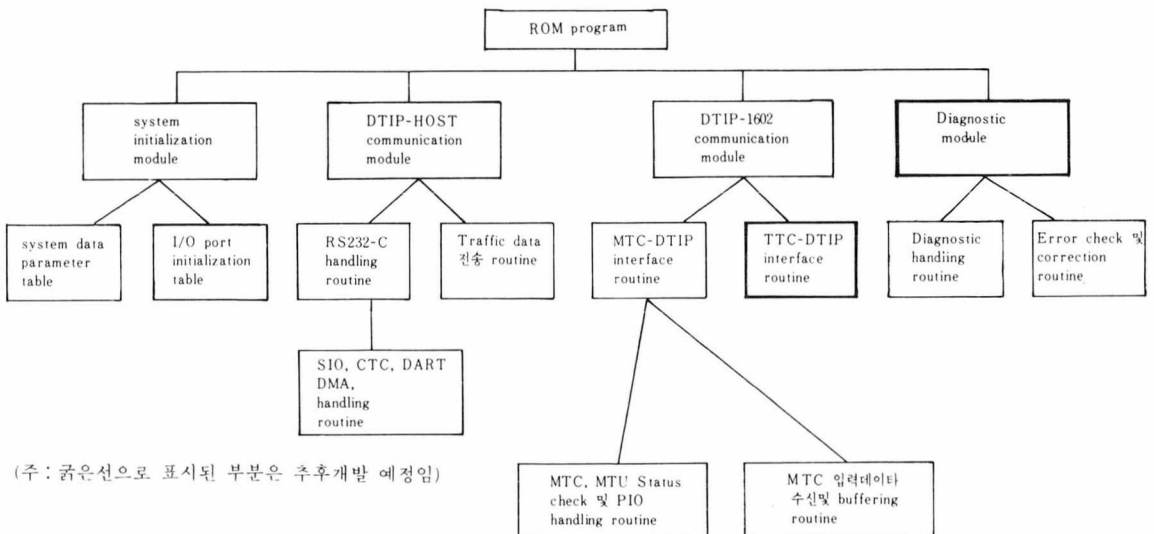
DART의 A port는 1602컴퓨터로 전송하며, B port는 local terminal에 연결하여 DTIP의 상태를 제어하게 한다.

CTC는 SIO와 DART의 송·수신 clock을 공급하기 위해 사용된다. 또한 스위치를 사용, 사용자가 임의로 스위치를 선택하게 함으로써 baud rate를 변경 가능하게 한다.

IV. DTIP 소프트웨어 設計

1. Module별 소프트웨어 構成

DTIP 소프트웨어의 module별 構成은 다음과 같은 hierachy를 가지며 시스템의 parameter 및 data를 定義하고 각 port의 初期化(Initialize)는 system initialization module, DTIP와 host computer와의 communication을 위한 DTIP-host communication module, 그리고 DTIP-1602의 연결 및 데이터의 송·수신을 위한 DTIP-1602 communication module 및 DTIP의 機能 진단을 위한 diagnostic module의 4가지로 나뉘어진다. (〈그림 2〉 참조)



(주: 굵은선으로 표시된 부분은 추후개발 예정임)

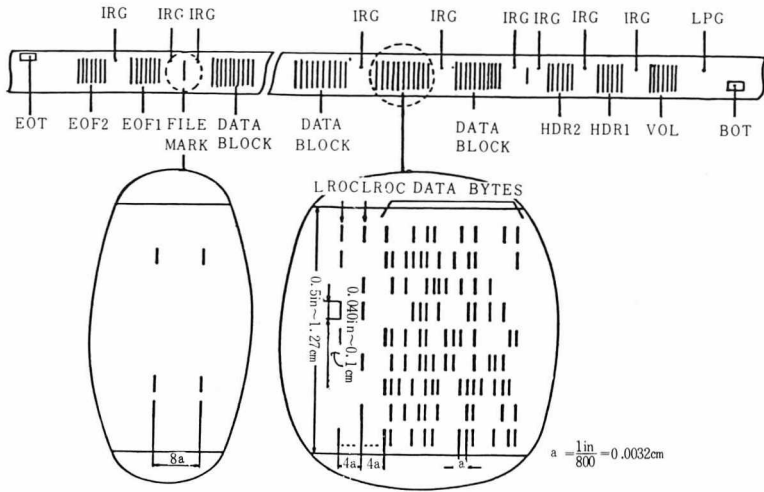
〈그림 2〉 DTIP S/W Structure

2. 입력단 소프트웨어 設計

M10CN file, label의 構造는 ECMA-13 stan-

dard를 사용하며, tape format은 다음과 같다.

(〈그림 3〉 参照)



〈그림 3〉 Tape Format

위의 그림에서 나타난 바대로 tape density가 800 bpi이므로 $a = 1/800$ inch가 되고 tape speed는 12.5ips이기 때문에 a의 간격 즉, 데이터 사이의 간격은 $1/800 \times 1/12.5 = 100 \mu\text{sec}$ 가 된다. 참고로 1602 시스템의 memory에 할당되어 있는 statistics buffer (2048byte) 내의 data block의 구성을 살펴보면 다음과 같다.

Address	내	용
'371722	Block identifier (MEAS : 4byte	Block Header부분
'371724	Block number : "	
'371726	Issue number : "	
'371730	Day of the year : "	
'371732	Record count : "	
'371734	Asterisks : 20byte	
'371746	Data : 2000 "	Data 부분
'373716	Asterisks : 8 "	

따라서 입력단 소프트웨어는 VOL-HDR label을 수신하는 routine과 data block을 수신하는 routine, 그리고 EOF label을 수신하는 3개의 routine으로 구성되며, 15분 단위 time out check routine을 설정하여 지정된 시간에도 달하면 지금까지 수신된 데이터를 host computer로 전송하기 위해 DMA-SIO 전송 routine

으로 enter될 수 있게 한다.

Data block receive routine의 경우, 1602 MTC로 부터의 데이터는 $2 \mu\text{sec}$ WDS pulse에 따라 $100 \mu\text{sec}$ 간격으로 출력되기 때문에 PIO는 input mode로 set시키고, 이를 받기 위하여 interrupt로 처리한다. 즉, WDS에 따라 interrupt를 발생하여 interrupt service routine으로 jump하게 하고, 이 routine에서 1byte의 data (WD₀ - WD₇)을 수신하여 memory에 load하고 다시 return하여 마지막 데이터를 수신하게 한다. 이때 마지막 data block의 검출은 MTC에서 나오는 file mark (17H)를 사용하며, 매 data block의 수신 끝에서 이 mark를 check한다. 데이터를 수신할 때 한 데이터의 수신에서 다음 데이터의 수신까지 $100 \mu\text{sec}$ 동안에 이루어져야 하므로 interrupt service routine이 끝나서 다음 데이터를 받기까지의 수행 시간을 계산해 보면,

Interrupt service routine

```

EI ; 4T
IN A, (PIO1DB) ; 11T
LD (HL), A ; 7T
INC HL ; 6T
    
```

DEC HL ; 6T
 RETI ; 14T

수행시간 = (4 + 11 + 7 + 6 + 6 + 14) × 0.25μs/T
 (4MHZ 기준)
 = 12μsec

Main program

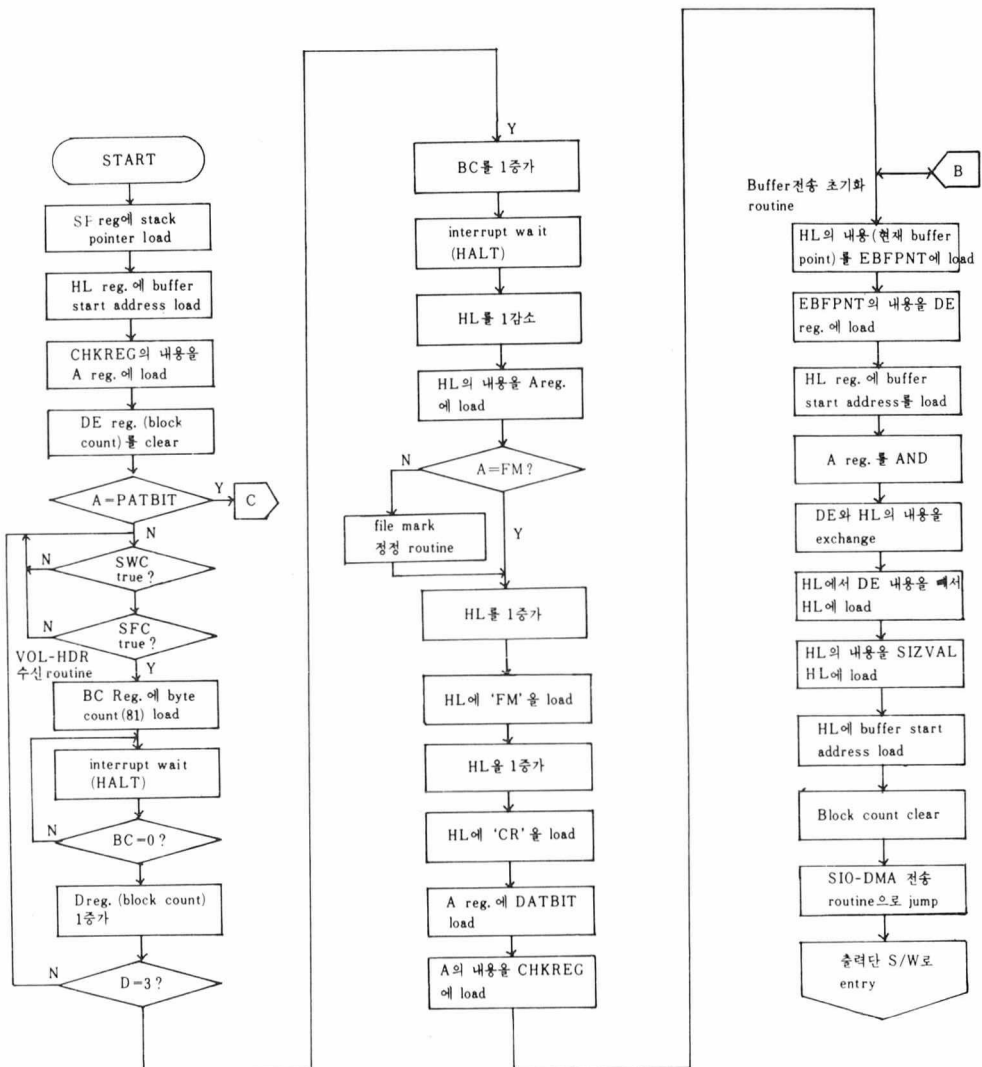
LD A, C ; 4T
 CP 0 ; 7T
 JP NZ, WDINT 2 ; 10T
 LD A, B ; 4T
 CP 0 ; 7T
 JP NZ, WDINT 2 ; 10T

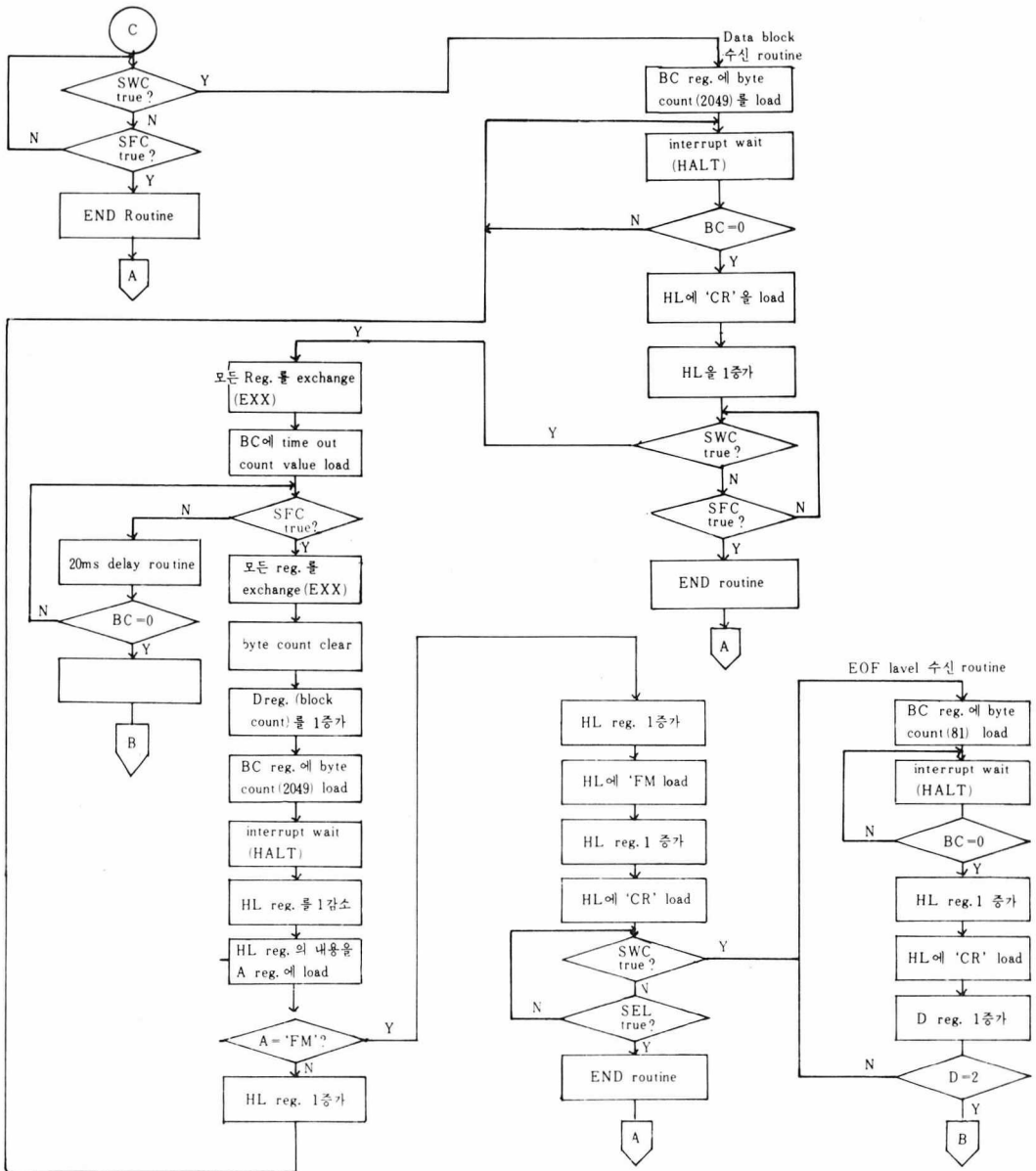
수행시간 = (4 + 7 + 10 + 4 + 7 + 10) × 0.25μs/T
 = 10.5μsec

이상의 계산으로부터 총 遂行時間은 12 + 10.5 μsec = 22.5 μsec가 된다.

따라서 INT를 acknowledge 하는 시간과 ISR로 jump하는데 걸리는 시간을 약 10 μsec로 잡아서 위의 총 遂行時間을 합해도 32.5 μsec정도가 되므로 67.5 μsec의 margin이 있기 때문에 data의 수신 및 처리에 필요한 timing은 충분하다.

Main program 중 이러한 기능을 하기 위한 입력단 소프트웨어를 flow chart로 나타내면 다음과 같다.





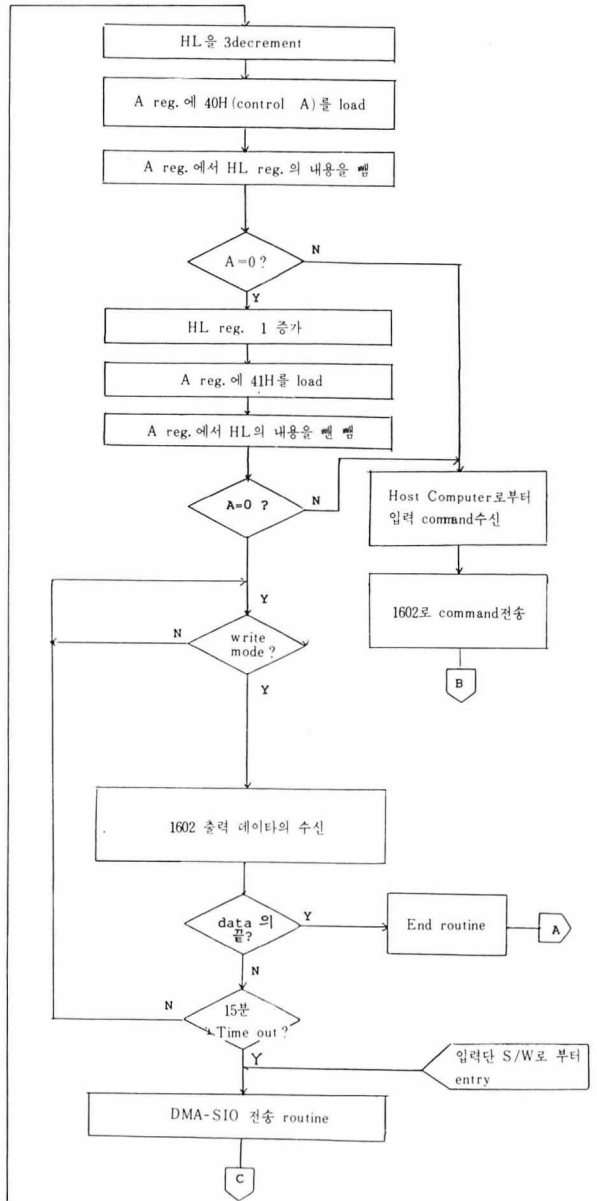
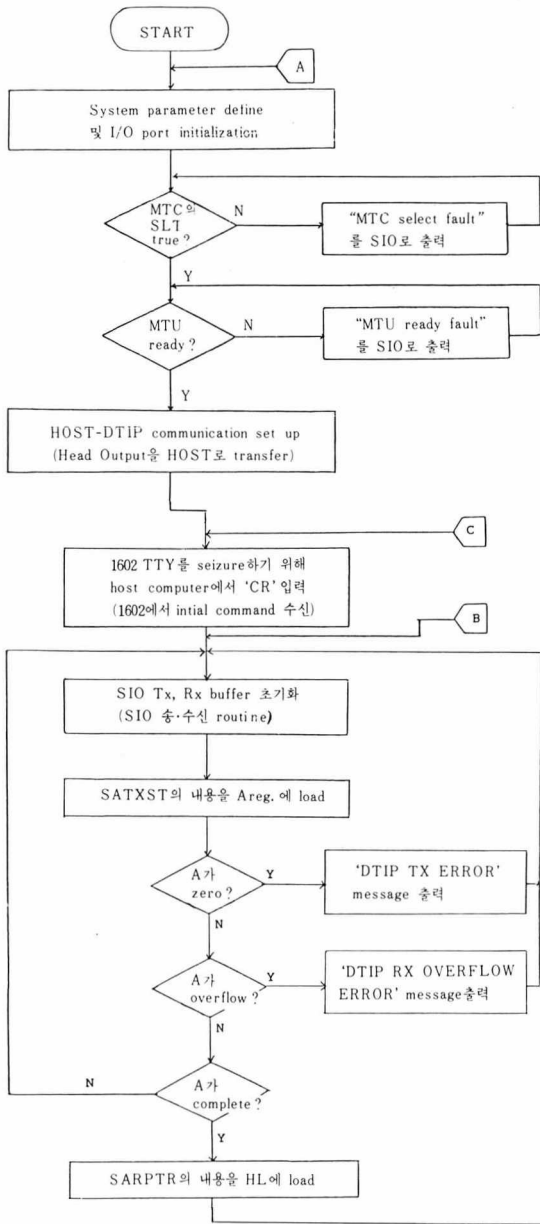
입력단 소프트웨어의 Flow Chart

3. 출력단 소프트웨어 設計

출력단 소프트웨어의 기능은 우선 트래픽 집중관리 시스템의 host computer의 데이터 통신이 이루어지게 함으로써 데이터의 전송이 가능하게 하고, host computer로 부터의 입력 명령을 받아 1602 컴퓨터(M10CN)로 전달하며, 앞서 언급한 입력단 소프트웨어로 pointer를 entry하여 출력되는 데이터를 수신하게 한다. 또

한 입력단 소프트웨어에서 모든 작업이 이루어지면 데이터 전송 routine으로 enter 시킴으로써 15분 측정 명령이 다시 주어지기 전에 일단 memory buffer에 있는 데이터를 host computer로 전송하게 하고, 측정 명령을 받는 routine으로 enter 함으로써 앞의 sequence를 반복하게 한다.

위와 같은 기능의 출력단 소프트웨어를 flow chart로 나타내면 다음과 같다.



출력단 소프트웨어의 Flow Chart

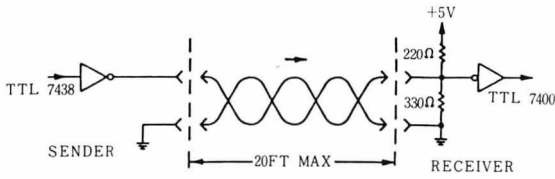
V. MTC-DTIP Connector 設計

1. Interface Signal 특성

1602 MTC와 Kennedy 9700 MTU는 twist된 pair cable로 연결되며, 각각의 receiver는 impedance matching이 이루어지도록 2개의 저항 (220Ω, 330Ω)으로 종단되고, sender는 open

collector driver (TTC 7438)에 의해 drive 된다. Twist된 pair cable은 cable간 누화를 감소시키기 위하여,

- 절연 두께 0.01 inch 이상
- 24 AWG (0.05mm 이상)
- 1 inch당 1회 이상 twist
- 길이 20 feet 이내의 규격으로 설계되었다.

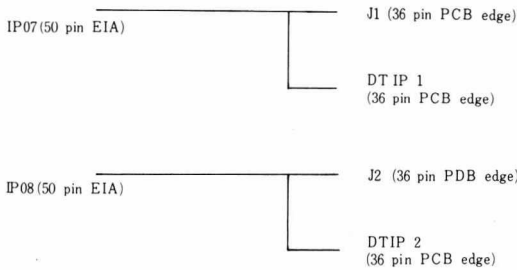


〈그림 4〉 Interface Configuration

2. Connector 設計

본 연구의 단계적 추진을 위해 첫번째 connector 설계는 common 방식으로 현 MTC와 MTU의 연결 cable에서 분기된 cable을 DTIP에 연결하고, MTC의 출력 데이터를 MTU, DTIP 양쪽에서 각각 수신하여 M. T와 DTIP memory 내용을 비교 검토한다. 이때 backward 신호는 현 MTU에게 보내도록 한다.

Interface connector는 3개의 36pin edge connector (J1 : control, J2 : write, J3 : read)로 MTU에 연결되어 있고 반대쪽은 50pin EIA connector로 바로 윗 rack의 IP07, IP08, IP09에 각각 별도로 접속된다. 데이터의 read 동작은 필요하지 않으므로 다음 connector 만을 설계한다.



〈그림 5〉 Common Connector

36pin edge connector는 pin 2개가 한조로 되어 한 pin은 active, 다른 한 pin은 ground에 연결된다.

Edge connector와 EIA connector의 pin연결과 signal은 다음과 같다.

가. IP07 - J1 (control)

(EIA)	(Edge)	(EIA)	(Endge)	(Signal)
1	A	34	→	+5V
2	B	35	←	OVW
3	C	36	←	SFC
4	D	37		

5	E	38	←	5	SRC
6	F	39		6	
7	H	40	←	7	RWC
8	J	41	←	8	SLT
9	K	42	←	9	SWS
10	L	43	←	10	OFFC
11	M	44	→	11	ONL
12	N	45	→	12	RWD
13	P	46	→	13	FPT
14	R	47	→	14	LP
15	S	48	→	15	WEN
16	T	49	→	16	RDY
17	U	50	→	17	EOT
18	V	19	→	18	RNG

(active) (ground)

나. IP08 - J2 (write)

(EIA)	(Edge)	(EIA)	(Edge)	(Signal)
1, 2	A	34, 35	1	WDS
3, 4	C	36, 37	3	WARS
5, 6	L	38, 39	10	WDP
7, 8	M	40, 41	11	D 0
9, 10	N	42, 43	12	D 1
11, 12	P	44, 45	13	D 2
13, 14	R	46, 47	14	D 3
15, 16	S	48, 49	15	D 4
17, 18	T	19, 50	16	D 5
20, 22	U	21, 23	17	D 6
24, 26	V	25, 27	18	D 7

(active) (ground)

VI. 結 論

이상과 같이 通話量 데이터를 on-line으로 전송하기 위해 設計된 DTIP의 機能을 說明하였으며, 해당 데이터 受信 routine이 interrupt를 처리하기에 充分한가를 CTC에 의한 펄스 發生에 의하여 實驗적으로 證明하였다. 그리고 1602 시스템의 signal interface가 common 방식의 connection으로 構成하였을 때 信号 伝送에 문제점이 없음을 밝혔다.

DTIP의 試驗은 해당분국에 S/W center에 設置된 M10CN MODEX에 連結하여 試驗하고, 데이터 出力의 確認은 일단 CRT의 display를 통해 하기로 하였으며, 추후 host computer에 連結하여 試驗하기로 하였다.