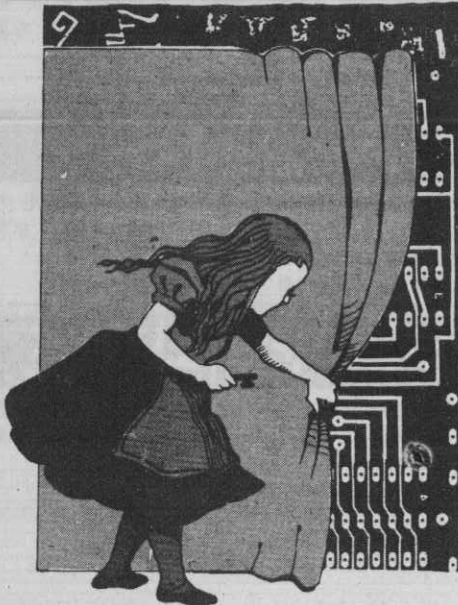


SPC-1000용

## I/O 슬롯의 제작



### 1. 개요

SPC-1000을 가지고 있는 독자들 중에서도, 애플컴퓨터를 사용해 본 경험이 있는 사람들이 상당히 많을 것이다. 각자가 애플컴퓨터를 사용하면서 가장 부러웠던 것은 무엇인가? 그것은 아마도 애플컴퓨터의 뒷뚜껑을 열면 바로 보이는 8개의 I/O 슬롯이 아닌가 생각된다. 애플컴퓨터는 SPC-1000과는 달리 이 8개의 슬롯을 이용하여 8가지의 서로다른 주변기기를 제어할 수 있다. 그런데 이와는 달리 SPC-1000에는 EXPANSION BUS외에는 주변기기를 제어할수 있는 확장용 슬롯이 없어 애플컴퓨터에서와 같이 다양하게 인터페이스 카드를 사용할수 없다.

따라서 SPC-1000을 가지고 있는 독자들은 SPC-1000도 애플컴퓨터와 같이 다양하게 인터페이스 카드를 사용할수 없을가하는 문제를 생각해보았을 것이다. SPC-1000도 애플컴퓨터처럼 다양하게 인터페이스카드를 사용할 수 있다면 거기에 맞는 여러가지 프로그램과 많은 주변기기들이 개발되어 SPC-1000을 사용하는 사용자에게 보다 넓은 환경을 조성하여 줄 것이다.

따라서 여기서는 이러한 여러가지의 인터페이스 카드를 사용할 수 있도록 SPC-1000용 I/O 슬롯을 만들어 보자.

우선 I/O슬롯을 만들기 위해선 EXPANSION BUS에 대하여 사전 지식이 있어야 하므로, 여기에 대한 약간의 설명을 하기로 한다. EXPANSION BUS란 외부 확장용 버스로서 이곳으로는 어드레스 선, 데이터 선, 전원, R/W신호,  $\overline{\text{EXTI}}$ 신호, 클럭(CLK), RESET신호등이 출력된다.

이러한 EXPANSION BUS커넥터(CONNECTOR)의 각핀에 대한 자세한 개요를 표 1에 나타내었다.

표 1에서 가장 주요해야 할 것은  $\overline{\text{EXTI}}$ 이라는 신호이다. 우선 그림 1을 보자. 이것은 SPC-1000 내부의 I/O DECODER 부분의 회로이다.(SPC-1000사용자 설명서 P81, 회로의 IC, F3부분 참조) 여기서 74LS138은 DECODER IC로서 A~C에 들어오는 2진수의 값에 따라  $Y_x$ 가 선택(enable)

되게 된다. (예 : 만약 C, B, A에 각각 1, 0, 0이 입력되었다면 이것은 2진수 100, 즉 10진수로 4, 그러므로 Y<sub>4</sub>가 선택된다. —여기서 “선택” 된다는 말의 뜻은 enable된다는 뜻으로 그 신호가 정논리일 경우에는 1, 또는 HIGH가 되고, 부논리일 경우에는 (회로도상에 '0' 표시, 혹은 그 신호명의 위에 부논리 기호 '—'가 붙어 있을 경우) 0, 또는 LOW가 된다는 뜻이다—). 또한 각자가 SPC-1000 사용자 설명서 P43을 보게되면 I/O MAP 이라는 것이 있는데 여기에는 I/O 어드레스에 대해 자세히 나와 있으므로 참고하기 바란다.

1	A 6	18	D5
2	A 12	19	D2
3	A 5	20	D4
4	A 7	21	D3
5	A 4	22	INT
6	A 8	23	RESET
7	A 3	24	$\overline{\text{IORQ}}$
8	A 9	25	$\overline{\text{MRIQ}}$
9	A 2	26	$\overline{\text{EXTI}}$
10	A 11	27	SYSCLK
11	A 1	28	$\overline{\text{BUSRQ}}$
12	A 10	29	$\overline{\text{BUSAk}}$
13	A 0	30	RD
14	D 7	31	$\overline{\text{WR}}$
15	D 0	32	5V
16	D 6	33	5V
17	D 1	34	GND

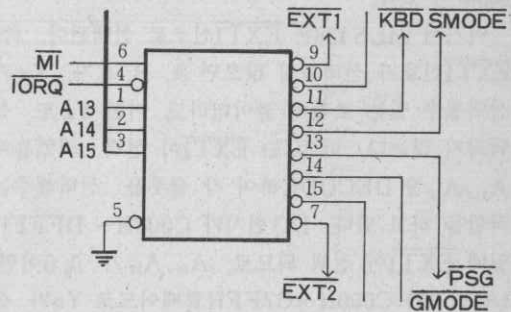
※ IC의 +5V는 제일 마지막 번호의 핀, GND는 1번 핀쪽의 가장 큰 번호의 핀에 연결.

ex) 74LS244 10번핀 → GND 20번핀 → +5V

[표 1]

Z80CPU가 IN, OUT명령으로 그 MAP의 I/O 어드레스 중 어느 한번지를 액세스(ACCESS) 하면, 그림1의 IC, 74LS 138이 A13~A15를 디코드(DECODE)하여 그 번지에 해당하는 신호(VRAM EXTI등)을 선택해준다. 그러므로 여기서 EXTI이라는 신호는 Z80에서 IN, OUT명령으로 C000

H~DFFFH번지 사이를 액세스할때 선택되는 신호이다. 따라서 I/O MAP에서 보는 바와 같이 I/O번지 C000H~DFFFH까지는 EXTERNAL I/O CHIP SELECT로서, 이 EXTI이라는 신호가 바로 주변기기 인터페이스 카드를 선택해주는 신호임을 알 수 있다. E000H~FFFFH 번지의 EXT2도 같은 종류의 것이나 뒤의 EXPANSION BUS로 출력되지 않으므로 생략하기로 한다.



[그림 1] I/O디코더 부분의 회로

그런데 애플컴퓨터에서와 같이 여러개의 확장슬롯을 만들게 될 경우 1개의 슬롯에는 하나의 인터페이스 카드만이 꽂혀짐으로, EXPANSION BUS에서 나오는 신호만으로는 하나의 SLOT밖에 사용할 수 없다는 결론이 나온다. 여기에 앞으로 제작할 I/O 슬롯의 중요한 문제점이 발생하게 된다.

만약 SLOT0부터 SLOT4까지 5개의 확장슬롯을 제작한다면 I/O 슬롯은 I/O번지 C000H~DFFFH까지의 EXTI을 SLOT1~SLOT4까지로 4등분하여, I/O번지 C000H~C7FFFH까지가 액세스(ACCESS)되면 SLOT1을 선택하는 SLOT1신호가 선택되고, C800H~CFFFFH까지는 SLOT2를 선택하는 SLOT2신호, D000H~D7FFFH까지는 SLOT3를 선택하는 SLOT3신호, D800H~DFFFH까지는 SLOT4를 선택하는 SLOT4신호가 각각 선택되도록 구성하여야 한다. 그리고 SLOT0은 좀 특별한 것으로 이 슬롯의 선택신호는 EXTI이 곳바로 들어가고 모든 선이 버퍼를 거치지 않고 직결되도록 한다. 즉 EXPANSION BUS와 똑같은 조건이 되도록 구성한다. 따라서 이 슬롯은 FDD와 같이 DMA(Direct Memory Access)전송을 필요로 하는 경우에 사용된다.

## 2. 회로설명

이와같이하여 회로를 구성하면 그림 2와 같이 나타낼 수 있다. 여기서는 그다지 긴설명을 필요로 하는, 특별한 것은 없다. 제작에 사용되는 IC로는 왼쪽 상단의 DECODER IC, 74LS138, 그리고 중간쯤의 버퍼IC, 74LS244, 쌍방향 버퍼인 74LS245등이 있다.

여기서 74LS138은  $\overline{\text{EXTI}}$ 신호로 선택된다. 즉,  $\overline{\text{EXTI}}$ 신호가 선택되지 않으면 A, B, C가,  $Y_x$ 가 선택될수 있는 조건에 놓이더라도 어떤  $Y_x$ 도 선택되지 않는다. 이 IC는  $\overline{\text{EXTI}}$ 이 선택 되었을때  $A_{11}, A_{12}$ 를 DECODE하여 각 슬롯을 선택해주는 역할을 하고 있다. I/O번지가 C000H~DFFFH 일때  $\overline{\text{EXTI}}$ 이 선택 되므로,  $A_{11}, A_{12}$ 가 0, 0이면, I/O번지가 C000H~C7FFH일때이므로  $Y_0$ 가 선택되고, 이  $Y_0$ 은 SLOT1을 선택해주는 신호인  $\overline{\text{SLOT1}}$ 에 연결되도록 구성되어 있다. 이런 식으로 C800H~CFFFH일때는  $\overline{\text{SLOT2}}$  선택되고 D000H~D7FFH일때는  $\overline{\text{SLOT3}}$  선택되며 D800~DFFFH일때는  $\overline{\text{SLOT4}}$ 가 선택된다. 따라서 SLOT3에 있는 인터페이스 카드를 쓰고 싶다면, D000H~D7FFH사이의 어드레스를 IN, OUT명령으로 액세스하여 3번 SLOT을 선택해주면 SLOT3에 있는 인터페이스카드를 사용할 수 있다. 그 다음, 74LS244는 3-STATE버퍼로서  $\overline{\text{EXTI}}$ 이 선택될때만 어드레스 라인을 LOW상태로 만들고, 그 외에는 하이임피던스 상태로 만든다. SLOT0의 어드레스라인은 이 버퍼를 통하지 않고 직결되도록 구성되어 있다. 74LS245도 마찬가지로 데이터 라인을 LOW상태로 만들며 이것은 쌍방향성이다. 그외  $\overline{\text{B}}\overline{\text{USRQ}}$ 와  $\overline{\text{BUSAK}}$ , 이 두신호는 DMA전송이나, 다른 CPU카드를 꽂을때만 사용되는 신호이므로 SLOT0에만 연결한다.

## 3. 부품의 구입

이 I/O 슬롯을 제작하기 위해서는 표 2에 있는 부품들이 필요하다. IC는 보통형과 LS형이 있는

데 반드시 LS형으로 구입하여야 한다. 그렇지 않으면 특성이 다르므로 동작시 문제점이 발생하게 된다. 34선 플랫 케이블은 34핀 소켓 구입시, 그 상점에서 직접 연결을 한다. 그러면 기계로 깨끗이 연결을 할 수 있다. 기계로 연결하지 않고 손으로 할 경우는 상당히 연결하기가 어렵다. 그 외에 납땀기와 납이 조금 필요한데, 요즘은 IC전용으로, 세라믹 히터를 사용한 납땀기가 나와 있으므로 약간 비싸긴 하지만 여유가 있는 사람들은 한번 사용해 보는 것도 괜찮을 것이다.

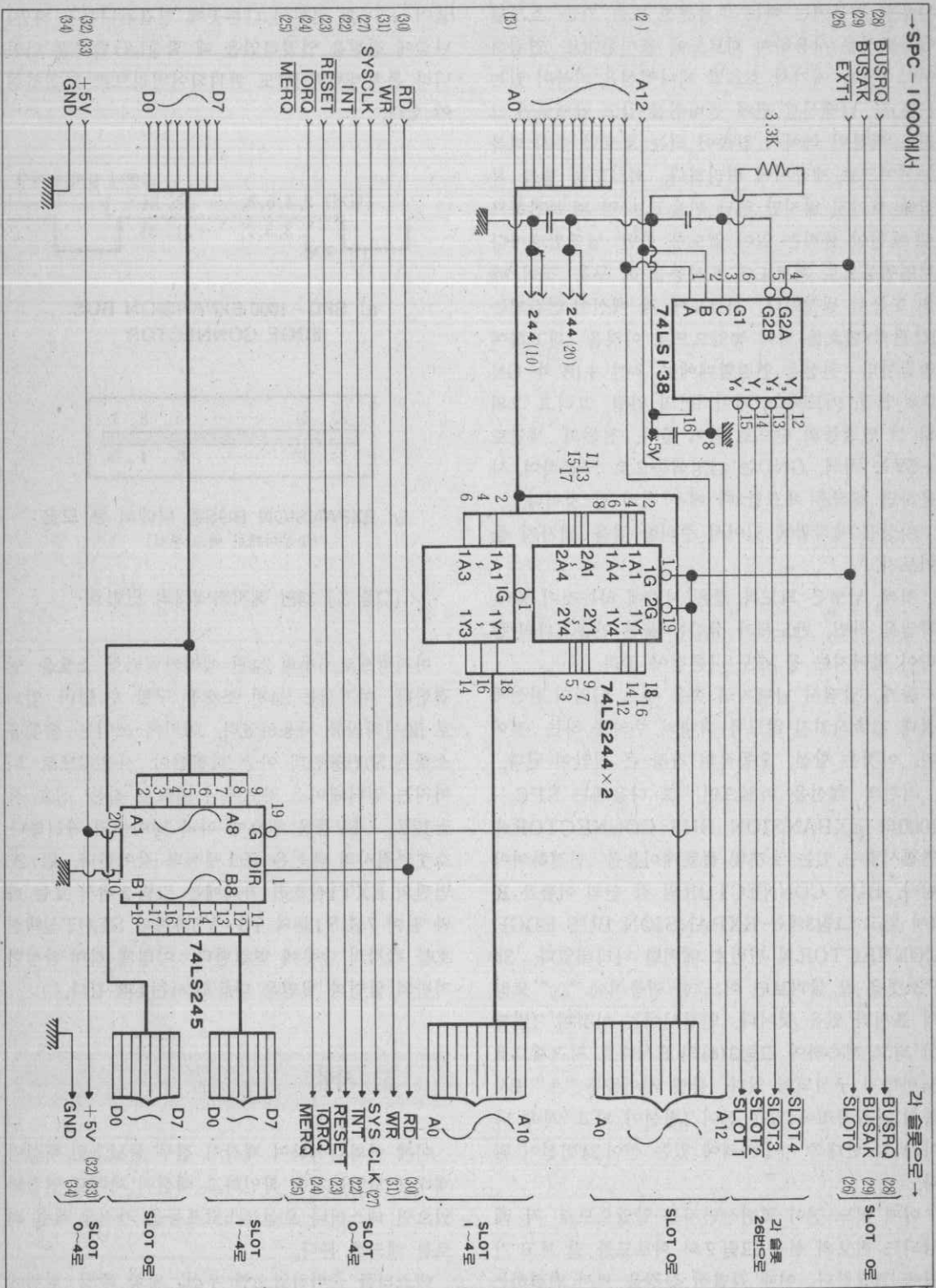
부 품	가 격
IC용 만능 기관	600
소켓34P HIF5-34D-R	2800
플랫 케이블 34P (30cm)	600
소켓50P PC-50A×5	700×5
IC 소켓 20P×2	120×3
IC 소켓 16P	80
74LS 244×2	800×2
74LS 138	500
74 LS 245	800
저항 3.3KΩ	20
세라믹 콘덴서 100P×3	8×3
그외 배선용 전선 약간 (CFC-34로 함이 좋다)	1900
납약간	

[표 2] 부품표

## 4. 제작

자, 이제 모든 부품이 전부 준비되었으면 제작에 들어 가보자, 우선 IC용 만능기관의 적당한 위치에 각 IC와 소켓을 꼽는다. 몇개 되지는 않으나 다리의 수가 많으므로 약간의 주의를 요한다. 각자가 편리한, 간편한 위치에 IC, 소켓을 꽂았으면 점선으로 연결을 시작한다. “래핑”이라는 방법을 사용하면 좀 더 쉬워지나 래퍼, 언래퍼등의 기구를 사용하는등 사전준비가 필요하므로 여기서는 직접 전선을 연결하여 납땀하는 방법을 사용한다. 전선을





사용할 경우에는 가는 피복전선 또는 가는 호르말 에나멜선을 사용하여 회로도에 표시한대로 연결하여야 한다. 여기서 호르말 에나멜선은 피복이 있는 그대로, 납땜시킬 핀에 한바퀴를 감고 납땜하면 그 열로 피복이 녹아서 접속이 되는 호르말 수지 피복 동선이므로 제작시에 편리하다. 회로도는 별로 복잡해 보이지 않지만 일단 만들고 나면 꽤 복잡하므로 배선이 틀리는 일이 없도록 한번 납땜할때마다 색연필등으로 회로도에 표시등을 해 두는 것이 매우 도움이 될것이다. 회로도에 각 배선이 연결되는 IC 핀의 번호를 적어 놓았으므로 이것을 참고하여 연결한다. 전선을 연결할때에는 우선 +5V와 GND의 연결, 어드레스, 데이터선의 연결, 그리고 그외의 선 연결등의 순으로 함이 좋다. 전선의 색깔도 +5V는 적색, GND는 검정색등으로 구분하여 사용하면 제작후 체크할 때 매우 편리 할 것이다.

이상을 제작함에 있어서 주의할 점을 몇가지 들 어보자.

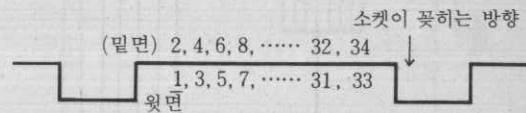
첫째, 납땜은 되도록 짧은 시간에 하는것이 좋다. 과열을 하면, 반도체가 상함은 물론 만능기판의 동 박이 떨어지는 등 매우 나쁜점이 많다.

둘째, 납땜시 납땜기의 촉을 필요 이외의 전선에 절대 접촉시키지 않도록 최대의 주의를 하는 것이다. 이것은 합선, 오동작의 가장 큰 원인이 된다.

기판의 배선을 마쳤으면, 그 다음에는 SPC-1000의 EXPANSION BUS CONNECTOR와 연결시킬수 있는 소켓및 플랫케이블을 연결하여야 된다. BUS CONNECTOR의 각 핀의 이름은 표 1에 있고 그림3에는 EXPANSION BUS EDGE CONNECTOR의 핀번호 배치를 나타내었다. 34P 소켓을 잘 살펴보면 어느 한 귀퉁이에 “△” 모양의 표시가 있을 것이다. 연결시에는 이것이 1번핀 이 되고 계속하여 그림3(a)의 표시대로 치그재그로 34번까지 구성되어 있다. 플랫 케이블은 “△” 표시 쪽의 가장자리에 있는 선이 1번선이 되고 계속 나 아가 그 반대쪽 가장자리에 있는 선이 34번선이 된 다.

이제 어느 선이 몇번선인지를 알았으므로 각 해당되는 번호의 선을 그림 2의 회로도를 잘 보고 기 판에 연결한다. 이때 각별히 신경을 써서 연결하는 것이 좋을 것이다. 정확히 연결하지 않으면 (예를

들어 +5V의 전원이 다른곳에 연결된다든지 하면) 나중에 전원을 연결하였을 때 확장I/O 슬롯뿐 만아 니라 본체자체의 IC도 파괴되어버리므로 조심하여 야 한다.



a) SPC-1000 EXPANSION BUS  
EDGE CONNECTOR

33, 31	.....	5, 3, 1
34, 32	.....	6, 4, 2

b) EXPANSION BUS를 뒤에서 본 모습  
(고무마개를 빼고 본것)

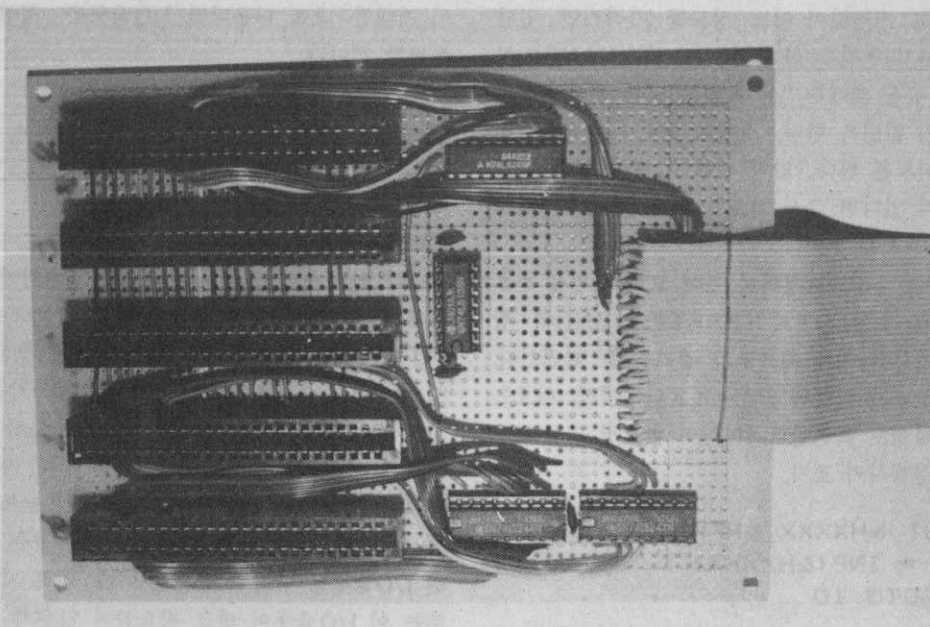
[그림 3] 34핀 에지커넥터의 핀번호

마지막으로 5개의 34핀 에지커넥터형 소켓을 연 결한다. 여기서는 34핀 소켓을 구할 수 없어 임의 로 50핀짜리를 사용하였다. 여기에 쓰이는 슬롯용 소켓은 50핀용인데 이중 34핀만이 사용되므로 나 머지는 인터페이스 카드에서 필요로 하는 신호 즉  $\pm 12V$ ,  $-5V$ 등을 필요에 따라 넣어주면 편리하다. 소켓연결시의 배선은 표 1에서와 같이한다. 단 26 번핀의 EXT1 신호핀 자리에는 그림 2에서 보는 바 와 같이 74LS138의 Y0~Y3까지의 SLOT선택신 호를 각각의 슬롯에 연결한다. 이렇게 하여 완성한 기판의 앞면과 뒷면은 다음의 사진1과 같다.

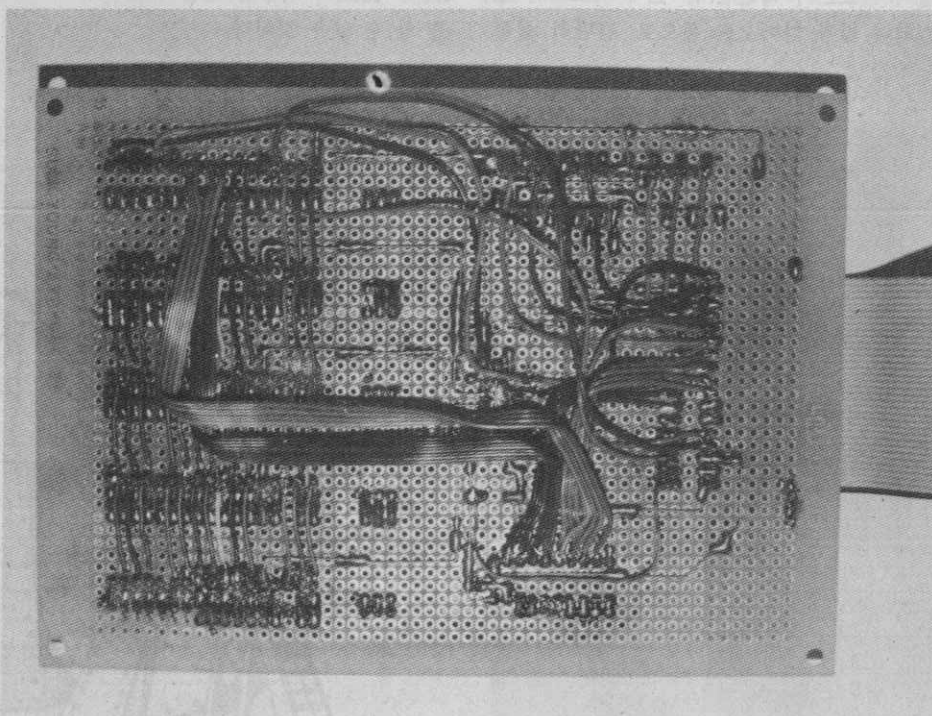
## 5. 시험

이제 이와같이하여 제작이 전부 끝났으면 배선이 제대로 되었는지를 확인하고 배선이 제대로 연결되 었으면 테스터나 오실로스코프등을 가지고 직접 회 로를 체크해 본다.

테스터를 준비하였으면 우선, 저항 측정 범위에 놓고 기판 +와 GND의 도통검사를 한다. 만약 저



[사진 1] 완성된 기판의 앞면과 뒷면





항값이 0에 가깝다면 그것은 +와 GND가 합선이 된 것이다. 저항값이 어느 정도를 가리키면 일단 안심하고 연결을 한다. 소켓을 EXPANSION BUS에 꽂을 때에는 “△”표에 주의를 해서 반대로 꽂는 일이 없도록 한다. 정확히 꽂았다면 컴퓨터의 전원 스위치를 켜다. 만약 이때 IC에서 열이 나거나 이상이 있다면 즉시 끄도록 한다. 이상이 없다면 받은 성공한 것이다.

테스터로 +5V가 정확히 들어 오나 확인을 한후 그 다음에 각 SLOT의 RESET핀을 GND와 합선시켜 컴퓨터 내에 리셋트가 걸리는가 확인한다. 만약 스코프가 있다면 SYSCLOCK단자에서 나오는 신호의 파형도 확인한다. 이상이 없으면 다음 프로그램 실행시켜 보자.

```
10 OUT &HXXXX,&HFF (각 슬롯 해당번지)
20 Y = INP(&HXXXX)
30 GOTO 10
```

그 다음 스코프를 가지고 어드레스 라인, 데이터 라인, IORQ, EXT, RD, WR 등을 측정하여 기대한 파형이 나오는지 확인한다. 만약 스코프가 없다면 테스터나 발광 다이오드 등으로 간단히 실험

해 보아도 괜찮을 것이다. 이렇게 하여 모든 파형이 기대한 대로 나온다면 I/O 슬롯은 정상적으로 동작할 것이다.

## 6. 맺음말

이 I/O 슬롯은 아직까지는 아무런 쓸모가 없다. 그러나 얼마 안있어 SPC-1000용 주변기기 및 인터페이스 카드가 속속 나올 것으로 예상되므로 이 I/O 슬롯은 그 활용범위가 매우 넓어질 것이다. 그리고 이 I/O 슬롯은 각자가 주변기기를 직접 설계 제작하는 데도 사용할 수 있을 것이다. 또한 이 I/O 슬롯은 설계면에 몇가지 불충분 한 점이 있을지도 모르겠다. 하지만 각자가 이 회로를 기초로하여 미비한 점을 연구하여 고쳐서 사용한다면 한층 더 편리한 I/O 슬롯을 만들수 있을 것이다. 비록 지금 현재로는 이 I/O 슬롯이 별로 필요하지 않을지 모르나 가까운 시일안에 이 I/O 슬롯을 이용한 여러가지의 인터페이스카드가 제작되어 나오면 이 I/O 슬롯은 SPC-1000의 기능을 한층더 넓히는데 커다란 몫을 할 수 있을 것이다.

[완성된 I/O 슬롯]

