

부설되, 최광무, 이견영, 이용식, 송상문

한국 과학원 Project Micro Lab.

A 16-bit computer system, called X77, is designed and constructed using bit-slice microprogrammable microprocessor. A cross-assembler/simulator software has been implemented and used to debug the entire system design including actual microcodes prior to hardware realizations. Prototypes of X77 CPU, I/O processing subsystems and their interconnections have been constructed and are in the process of being debugged.

1. 서론

최초의 μP 4004 가 발표된 이래 μP 의 발전 속도는 놀라워서 현재는 여러 종류의 μP 들이 여러 반도체 회사들에 의하여 개발되었으며, 그 응용 범위도 광범위하여 간단한 가계제품에서부터 금전등록기, 기측기기, 통신기기, 혹은 Computer 등의 이르기까지 다양하다. 이렇게 μP 가 전자공학 혹은 computer science 분야의 새로운 총아로 등장하게 된 것은, 반도체 집적 기술의 발달로 인하여 강력한 기능을 가지는 processor 를 한 chip 혹은 몇 개의 chip 속에 충분히 넣을 수 있게 됨으로써 processing power 의 가격이 급격히 하락하여 종전에는 상상할 수도 없던 응용 분야에 μP 를 사용하여 비교적 낮은 가격으로, 제품의 값을 높일 수 있게 된 데에 가장 큰 원인이 있다 하겠다.¹

이러한 μP 응용의 발전과 더불어 computer system 에서도 μP 들을 이용하는 노력이 시도되어 diskette driver, 한글 CRT, Printer 등 각종 Computer 주변 장치에 μP 를 부착하여 주변 장치의 기능을 강화시켰다. 또한 Computer CPU 자체를 μP 로 사용하려는 시도가 이루어지고 있다. 이것은 전자공학의 기본 소자나 진공관에서 Transistor, fixed function IC 를 거쳐서 LSI 로 바뀌어 온 것의 직접적이라 하겠다. 그러나 μP 를 Computer system의 CPU 로 사용하는 것은 data path, execution time 등의 제한으로 Mini-computer 이하의 수준²으로, 비교적 낮은 performance 를 가지고 있었다. 이러한 문제의 한 해결방안으로 Bit-slice microprogrammable μP 들이 등장하여 사용자가 computer CPU 의 data path 와 architecture 를 임의로 정하여 빠른 속도(Bipolar technique) CPU 를

design 하는 방식이다. 본 논문에서는 이러한 방식으로서, Intel 의 Bit-slice microprogrammable microprocessor, Series 3000 을 사용하여 mini-computer 수준 의 computer system 을 설계 제작하려는 한국과학기술원 전산학과 Project Micro Lab.의 Project X77 에 대하여 기술하고자 한다. 또한 본 project 의 목적은 위에서 언급한 general purpose computer system의 설계 제작 이외에도 microprogramming 의 특성을 살려서 language processor 등 high performance dedicated system 제작의 준비단계이기도 하다.

2. 배경

(1) Series 3000

AMD 의 2900 과 비교하여 다음과 같은 장점을 취하여 Series 3000 을 채택하였다.

- ① macro instruction decoding 이 용이하여 일반적인 Computer system 의 CPU 제작에 용이하다.
 - ② micro-code width 가 비교적 작아서 control ROM 의 크기가 줄어든다.
 - ③ Bus 의 종류가 많으므로 (5가지의 Bus) 외부의 circuit 가 줄어든다.
- (2) PACE ⁵

PACE 는 National Semiconductor 회사에서 발표한 16-bit μ P 로서 PMOS 로 되어 있어서 속도는 느리나, CPU architecture 가 강력하고, Instruction Set 에 조직적으로 구성되어 있어서 3000 Series 를 사용한 macro inst. decoding 이 체계적으로 이루어 질 수 있다는 점등의 장점을 취하여서 본 X77 system 의 CPU 의 model 로서 PACE 를 선정하였다.

3. 설계 과정

본 X77 의 중요한 의의 중의 하나는 설계 과정에서 사용된 공학적인 접근 방식에 있다. 즉, 급변하는 현대 공학기술에 발 맞추기 위하여는, 설계 제작 과정에서 소요되는 시간과 노력을 최소한으로 줄일 필요가 있다. 그러나 본 X77 과 같은 상당한 복잡도를 가진 system 의 경우에는 전 system 의 설계 신뢰도가 급격히 떨어지게 되므로 설계 과정에서 미리 수정과정에서 파생될 과오를 최소한으로 줄이는 노력이 요구된다.⁶ 이를 위하여서는 강력한 Design tool 을 사용하는 것이 바람직하며 본 X77 project 에서는 다음과 같은 다양한 Software 및 Hardware Design tool 들을 사용하여 설계 및 수정 과정에서 파생될 과오를 최소한으로 줄이는데 노력하였다.

(1) Software 보조 기구

- ① S3076

3000 Series LSI 등과, micro instruction pipelining, user defined micro instruction, macro instruction decoding 을 위한 보코 회로등 X77 CPU 의 hardware 및 firmware 전 system 에 대한 simulator 이다. 본 S3076 을 사용하여 X77 CPU 의 wire-wrapping 이 이루어지기 이전에 이미 X77 CPU 의 logic design 및 micro program 이 올다는 것을 simulate 할 수 있었다. 매 micro instruction 마다 각종 Bus, Pipe line reg., CPE 내부 register 등의 내용을 확인 할 수 있어서 복잡한 logic 의 설계 과오를 찾는 노력을 줄일 수 있었다.⁷

② A3076, M3076, I3076

A3076은 Intel CROMIS Package⁸ 의 XMAS 과 같은 기능을 수행하는 micro code programming cross assembler 이며, M3076 은 CROMIS Package 의 XMAP 과 같은 기능을 수행하는 Micro code 의 CROM mapper 이고, I3076 은 micro code 의 branch table 을 만들어 주는 program 이다. 이 이외에 PACE cross assembler, wire-wrapping list 를 위한 program 들이 있다.

(2) Hardware 보조 기구

① Intel MDS-ICE-30 3000 Series In-Circuit Emulator.⁹

② Intel SIM-104 ROM Simulator.¹⁰

Intel 의 MDS 에 부착해서 사용하는 MCU⁴ 의 "in-circuit" emulator 및 Control ROM Simulator 이다. (1)의 Software 보조 기구를 사용하여 설계 에 세심한 주의를 기울였으나, 실제 Hardware 에서 일어나는 문제들을 debug 하는데 In Circuit Emulator 와 ROM Simulator 가 매우 간편한 보조 기구로 사용되었다.

그림 1 에 Software 및 Hardware 보조 기구간의 관계가 도시되어 있다.

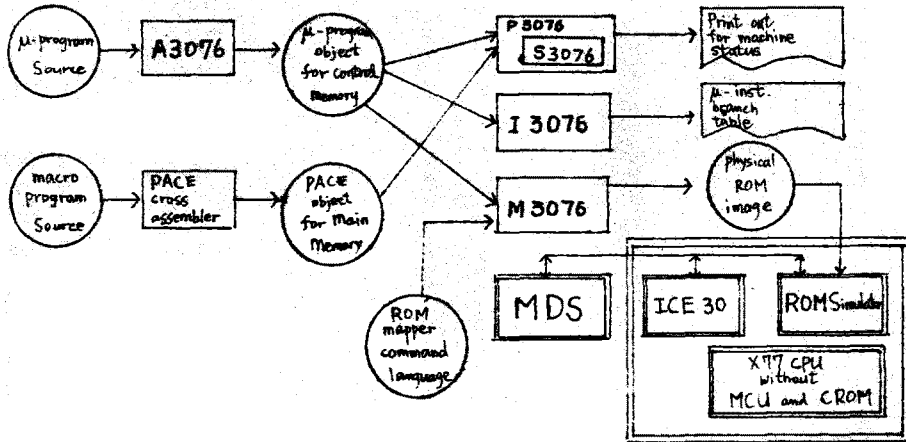
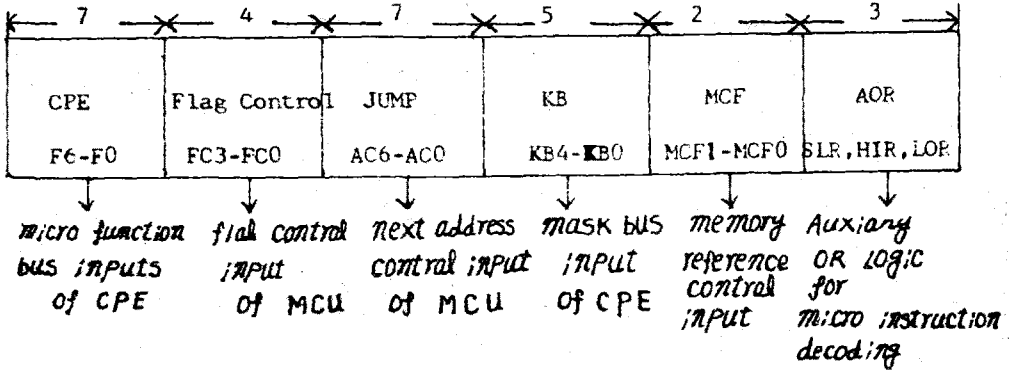


그림 1. 보조 기구 작동도

4. X77 CPU

(1) Micro-instruction Format

X77 CPU 의 micro-instruction format 은 그림 2와 같이 6개의 field 28 bit 으로 구성되어 있으며, CPE, Flag control, Jump 등 3개의 field 는 3000 Series standard ⁴ 이므로 KB, MCF, AOR 등 나머지 field 들에 대하여만 간단히 설명 하겠다.



1 링 2 MICRO instruction Format

① KB field

CPE 의 mask field 인 K-Bus 의 내용을 지정해 주는 micro instruction field 로서 그림 3과 같이 5bit 의 출력에 16 bit 의 입력에 연결되어 있다.

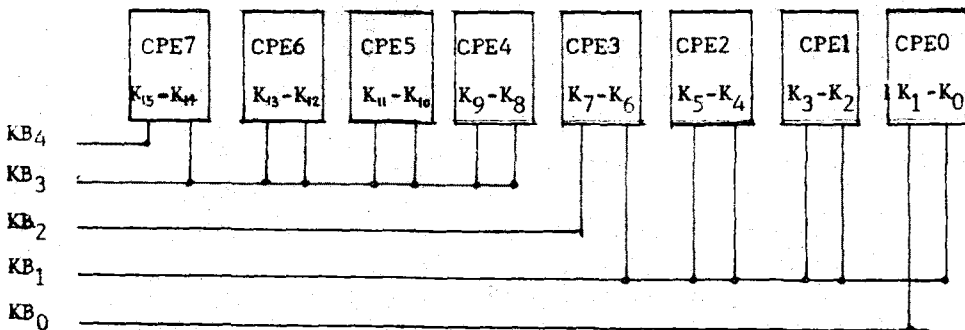


fig.3 K-Bus connecting

② MCF field

memory 의 read/write control 과 CPE 의 no load test (inhibit storing result) 를 위한 micro-instruction field 로서 그 내용이

표 1에 기술되어 있다.

MCF field	Numeric	Purpose
00	NFN	No Function(default)
01	INH	Inhibit CPE Clock
10	RRM	Request Read Memory
11	RWM	Request Write Memory

Table 1. MCF field

③ AOR field

SLR, HIR, LOR 의 3개의 bit 으로 구성되어 있으며 macro instruction decoding 을 위한 micro instruction field 로서 그 동작 원리는 뒤에서 설명하겠다.

(2) Macro CPU Architecture

Macro CPU 의 architecture 는 PACE 의 architecture 와 대부분이 비슷하며, 별도로 필요한 몇 개의 instruction 들이 추가되고, 별로 필요치 않은 몇 개의 instruction 은 삭제되었다. ¹¹ 따라서 본 X77 CPU의 macro architecture 는 PACE 와 거의 비슷하며, 자세한 내용은 참고문헌, 5등 관련 manual 을 참조하기 바라며, 본 논문에는 간단한 instruction format 만을 표2에 정리하였다.

표2. Instruction Format

Type	Remark	Machine Format	Instructions
1	Memory reference	$\begin{array}{ c c c } \hline \text{op} & \text{XR}_i & \text{disp} \\ \hline 15 & 9 \ 8 \ 7 & 0 \end{array}$	JMP JMP@ JSR JSR@ SKG SKAZ ISZ DSZ LD@ ST@ LSEX AND OR SUBB DECA
2	Immediate	$\begin{array}{ c c c } \hline \text{op} & \text{r}_i & \text{disp} \\ \hline \end{array}$	AISZ LI CAI
3	Memory data xfer and skip	$\begin{array}{ c c c } \hline \text{op}, \text{r}, \text{XR}_i & & \text{disp} \\ \hline \end{array}$	LD ST ADD SKNE
4	Register to register	$\begin{array}{ c c c c } \hline \text{op} & \text{dr} & \text{sr} & \text{unused} \\ \hline \end{array}$	RCPY EXCH RADD RADC RAND RXOR
5	Register with stack or FR	$\begin{array}{ c c c } \hline \text{op} & \text{r}_i & \text{unused} \\ \hline \end{array}$	XCHRS CFR CRF PUSH POP
6		$\begin{array}{ c c c } \hline \text{op} & \text{XR}_i & \text{unused} \\ \hline \end{array}$	PUSHF POPF HALT
7	Return	$\begin{array}{ c c c } \hline \text{op} & \text{00} & \text{disp} \\ \hline \end{array}$	RTS RTI
8	Shift and rotate	$\begin{array}{ c c c c } \hline \text{op} & \text{r}_i & \text{n} & \text{1} \\ \hline \end{array}$	SHL SHR ROL ROR
9	Branch on conditions	$\begin{array}{ c c c } \hline \text{op}, \text{cc} & & \text{disp} \\ \hline \end{array}$	BOC
10	FR flag	$\begin{array}{ c c c } \hline \text{op}, \text{fc}^* & & \text{unused} \\ \hline \end{array}$	SFIG (* is 1) PFIG (* is 0)

X: don't care, cc: branch conditions, fc: FR flag name. * stands for bit 7.

(3) Macro Instruction Fetch 와 Decoding 13,14,15

PACE 의 macro-instruction 의 대부분이 low byte 를 effective address 를 계산할때에 displacement 도 사용하므로 16 bit arithmetic 을 수행하기 위하여는 low byte 를 16 bit expansion¹² 하여야 한다. Fetch Sequence 는 모든 micro instruction 에서 공동으로 사용하는 부분이므로 이 부분의 micro code 를 efficient 하게 사용하기 위하여 X77 예서는 사용하지 않는 I Bus 를 사용하여 16 bit expansion 을 1 micro instruction 으로 수행하였다. 또한 PACE 의 instruction 의 addressing mode 는 표 3에 표시된 것과 같은 4가지 경우가 있는데, xr (macro instruction 의 bit9,8) 의 내용에 따라 R4,R5,R6,R7 을 각각 선정하여야 한다. 이것은 다음절에서 설명할 AOR 를 이용하여 간단히 1 micro instruction 으로 수행할 수 있다.

Xr	Addressing Mode	Effective Addr.	CPE E.A.
00	Base page	disp	(R4)+disp
01	PC relative	(PC)+disp	(R5)+disp
10	AC2 indexed	(AC2)+disp	(R6)+disp
11	AC3 indexed	(AC3)+disp	(R7)+disp

table 3. Addressing Mode

대부분의 PACE instruction 이 high order 4 bit 혹은 6 bit 의 op code 를 가지고 있으므로 Instruction 의 high order 4 bit를 MCU 의 Primary Instruction Bus(PX7-PX4)에 연결하여 JPX jump 를 사용하여 16 way branching⁴ 을 1 instruction 으로 수행할 수 있었다. index reg. 결정과 16 way branching 은 서로 다른 micro instruction field 를 사용하므로 1 micro-instruction 으로 가능하다. 따라서 instruction fetch 와 8 bit displacement 의 16 bit expansion 그리고 index register selection 과 16 way branching 등 복잡한 fetch sequence를 3 micro-instruction 으로 수행하여 X77 CPU 의 performance 를 크게 향상시켰다.

(4) register 배정 및 AOR field

index register 나, operand 의 source 혹은 destination register 를 결정하는 데 각각의 경우마다 4 way branch를 수행한다면¹⁶ micro instruction 의 길이는 매우 길어지게 될 것이다. 이것을 방지하기 위하여 CPE register 가 CPE function bus input 의 bit F2-F0 에 의하여 결정되는 것을 이용하여 macro instruction의 register selection 와 micro instruction 의 AOR field 와 CPE function field 들로써 적당한 combinational circuit¹⁷를 구성하여 그 output 을 CPE function

에 연결하였다. 이 경우 combinational circuit 를 간단히 하기 위하여 register 배정을 표 4 와 같이 하였다. AOR field 는 SLR,HIR,LOR 의 3 가지 경우가 있는데, 이것은 macro instruction 의 register selection field 의 위치가 표 2 에서 보여 지듯이 서로 다르므로, 각각의 경우를 기본하여 SLR 은 macro instruction 의 Bit 9,8 을 HIR 을 Bit 10,11 을 LOR 은 Bit 7,6 을 각각 사용하였다.

CPE scratchpad register	X77 CPU register
R0	ACO
R1	AC1
R2	FR(Flag Register)
R3	SP(Stack Pointer)
R4	always contains 0
R5	PC(Program Counter)
R6	AC2
R7	AC3
R8	IR(Interrup Register)
R9	not used

Table 4. CPE register allocation

5. I/O Processor^{19,20,21}

X77 Project 는 general purpose computer system 이라는 개념에 기초하여, I/O 의 문제가 중요한 해결 과제이다. 종래의 computer system 들과 같이 각 주변 장치들에 대한 controller 및 interface 들을 X77 CPU 에 맞게 직접 설계 제작하는 방식도 가정될 수 있으나, CPU 의 overhead 를 줄이고 현재 Project Micro lab. 에 설치된 micro computer system 들의 주변 장치를 그대로 사용하기 위하여¹⁸ 기존의 micro computer system 을 I/O processor 로 하고 X77 CPU 를 main processor 로 하는 "Concurrent" processing system 을 구성하였다. Main processor 와 I/O processor 상호 연결 방식은, 각 processor 의 main memory 의 일부를 두 processor 가 서로 공유하여 그 공유된 Memory(Common memory) 를 통하여 control 와 data 를 주고 받는 방식을 택하였다.^{19,20,21} 이러한 방식은 단순히 Main processor 와 I/O processor 가 Master 와 Slave 의 관계로서 유지되는 Centralized Processing 의 개념에서 확장하여 2 개의 Processor 가 Common Memory 를 통하여 서로 control message 와 data 를 주고 받으며 대등한 입장에서 운영되는 Distributed Processing 개념으로까지 확장시킬 수 있으므로, 앞으로의 X77 project 는 이 부분에 깊은 관심을 보일 예정이다. Common memory 에 적당한 message 나 data 를 쓴 후 상대방

processor 에게 그 Message 나 data 를 처리 할 것을 요구 하기 위하여는 서로 상대방 processor 에게 interrupt 를 걸 수 있어야 한다. Main processor 는 microprogram 되어 있으므로 interrupt service routine 을 microprogram 하여 firmware를 처리하고, Return from Interrupt (RII) instruction 을 새로 만들어서 Interrupt Service 의 performance 를 향상 시켰다.

6. 결과 및 결론

Microprogrammable μP 를 사용한 general purpose computer system 을 설계 하였다. Software 의 설계 및 제작이 있어서 다양한 Software 및 Hardware 보크 기구들을 사용하여 설계 및 점검과 검역이 소요될 시간과 노력을 최소한으로 줄여서 새로운 양산의 연구 개발 방법을 채택하였다. 제작된 X77 CPU 의 속도를 PACE 와 비교한 표가 아래 표 5에 표시되어 있다.

Execution time	PACE (2M hertz max. clock)	Series 5000	
		(200 n-sec u-cycle)	
INSTRUCTION			
RADD	$8.0 + E$	$1.4 + E$	
ADD	$3.0 + 2E$	$1.8 + E$	
CAI	$10.0 + E$	$1.2 + E$	
LI	$8.0 + E$	$1.2 + E$	
LD	$6.0 + 2E$	$1.4 + 2E$	
ST	$8.0 + E + E'$	$1.2 + E + E'$	
LDD	$8.0 + 3E$	$1.6 + 3E$	
STO	$8.0 + 2E + E'$	$1.4 + 2E + E'$	
JMP	$8.0 + E$	$1.0 + E$	
JSR	$10.0 + E$	$1.6 + E + E'$	
RTS	$10.0 + E$	$1.8 + 2E$	
XCHIS	$12.0 + E$	$1.0 + 2E + E'$	
AISZ	$12.0 + E$	$1.6 + E$	
ISZ	$15.0 + 2E + E'$	$2.0 + 2E + E'$	
SKG	$10.0 + 2E$	$2.2 + 2E$	
SKAG	$10.0 + 2E$	$1.0 + 2E + E'$	
ROL	$10.0 + 0.6(N-1) + E$	$2.2 + 0.6(N-1) + E$	
RHR	$10.0 + 0.6(N-1) + E$	$2.0 + 0.6(N-1) + E$	
PUSH?	$8.0 + E$	$1.0 + E + E'$	

표 5. PACE 와 X77 의 비교
 E: main memory access time
 E': main memory write time

본 X77 project 는 최선의 Software 와 Hardware 기술을 사용하여 General Purpose Computer System 을 설계 제작하는 작업으로서, microprocessor application 에서 Computer Architecture, Microprogramming, System Program 에 이르기 까지 다양한 spectrum 을 갖는 종합적인 장기 대형 연구 과제이다.

따라서 본 연구 과정에 대한 연구는 계속적으로 수행될 것이며, 그 계속적인 결과에 대한 보고서도 계속 발표할 예정이다.

참고 문헌 및 색인

1. 박철희, "Microprocessors", 한국 전기공학회지, 25.535 (1976)
2. IBM 의 E-Series 등이 개발된 이후 종전의 mini-computer 의 개념이 점차 모호해져서 종전의 mini computer 회사들은 Super mini 에서 새로운 활로를 찾고 있다.
3. 박철희 외 다수, "X77 - Design and Construction of a 16-bit Microcomputer System using Bit Slice Microprogrammable Microprocessor", 한국정보과학회지, Vol 4, No. 2, (1977) 본 논문은 참고 문헌 3에 계속되는 논문이다. 따라서 중복되는 부분은 가급적이면 간 설명을 피하였다.
4. "Intel Series 3000 Reference Manual", Intel Corp. (1976)
5. "PACE Technical Description", National Semiconductor Corp. (1975)
6. 이러한 장점은 Computer Science 의 Software Engineering 에서 Software 들 대상으로 현재 활발히 논의되고 있다.
7. Hardware debug 시에는 CPE 내부 register 들의 내용을 직접 볼 수가 없으므로 micro program debug 에 매우 편리 하였다.
8. "Intel Series 3000 Microprogramming Manual", Intel Corp. (1976)
9. "ICE-30 In-Circuit Emulator Reference Manual", Intel Corp. (1975)
10. "ROM-Simulator Reference Manual", Intel Corp. (1975)
11. 추가된 instruction 들은 Interrupt Service 를 위한 몇 개의 instruction 그리고, stack pointer 를 set 시키는 instruction, multiply 와 divide instruction 등이며 삭제된 instruction은 Branch On Condition 과 Flag Register 에서 별로 사용되지 않거나 PACE 에서만 쓰이는 독특한 몇 가지 condition 들이다.
12. low byte 의 sign bit(bit 7) 을 high byte 로 expansion 한다.
13. 안일사, "Emulation with the Microprogrammable Microprocessor", 한국 과학원, 석사학위 논문 (1976)
14. 송상훈, "Experiment of a Bit-Slice Microprocessor" 한국 과학원 석사학위 논문 (1978)
15. 이용석, "Experiment of Microprogramming with Microprogrammable Microprocessor", 한국 과학원 석사학위 논문 (1978)
16. MCU 의 JLL 혹은 JRL jump instruction
17. 실제로는 open collector 를 사용하여 Dot OR 하였다.
18. 현재 Project Micro Lab.에는 Intel 의 MDS 이외에도, Cromemco, IMSAI, ALTAIR 등 각종 microcomputer system 과 CRT, Line Printer, ITY, IV monitor, TV graphic terminal, Casette Driver, Diskette Driver 등 다양한 주변장치들이 설치되어 있어서 기존의 주변 장치를 그대로 사용할 수 있다.
19. A.J. Flavell, F. Gandini, J. Park, R. Ruedieger, R. Schilling, L. Schnupp,

"BRUSH Status Report", Max-Planck-Institut für Physik und Astrophysik, Abt. Numerische Rechenmaschinen, Munich, Germany (1972)

20. 손상혁, "Experiment on Concurrent Internal and External Processor," 한국과학기술원 석사학위 논문(1977)

21. 이천영, "I/O Processor Interface Implementation in X77 Multiprocessor Computing System" 한국과학기술원 석사학위 논문(1979)

22. 본 연구의 몇 가지 단계들이 한국과학기술원 석사학위 논문으로 수행되었다. 초기에 본 연구에 공헌했던 사람들은 안일수¹³, 이찬훈, 손상혁²⁰ 등이다.