

# 마이크로프로세서 데이터 처리 시험을 위한 最適試驗命令語

(Optimal Test Instruction Set for Microprocessor  
Data Processing Testing)

安 光 善\*

(Gwang Seon Ahn)

## 要 約

本 論文에서는 마이크로프로세서의 데이터 처리부 試験을 위한 最適試驗命令語 셉을 구하는 方法을 提示하였다. 基本 資料로는 user's manual로부터 얻을 수 있는 命令語 셉을 이용하였다.

最適試驗命令語 選定 過程은 3 단계로 설명된다: 1) 機能的으로 分離된 명령어에 대한 試験處理線圖의構成, 2) 必須線圖, 除去線圖 및 適任線圖의決定, 3) 最適試驗命令語 셉의確定

INTEL 8048인 경우, 最適의 試験命令語는 명령어 레퍼터리(96개의 명령어) 중 50개이다.

## Abstract

This paper deals with the selection of minimal test instruction set for microprocessor data processing test. This test method is based on a function description of the instructions which are obtained from the data given by the user's manual.

Selecting procedure is done in 3 steps: 1) a test execution graphs are represented on the instructions which are grouped functionally, 2) the essential graphs, the eliminable graphs, the eliminable graphs, and the eligible graphs are built, 3) optimal test instruction set from the essential graphs and the eligible graphs is defined.

In the case of INTEL 8048, 50 test instructions can be selected optimally from 8048 instruction repertoires (96 instructions)

## I. 序 論

일반적으로 마이크로프로세서의 試験方法은 機能試驗(functional testing)과 構造試驗(structural testing)으로 나누고 있다.<sup>[1]</sup> 기능시험은 마이크로프로세서를 하나의 시스템으로 여겨, 이 自体를 機能的으로 分리

한 演算裝置, 記憶裝置, PC(program counter) 등의 機械的인 部分으로 나누어 주어진 데이터로 處理하는 方法이며, 構造試驗은 시스템의 回路에 대한 知識을 가지고 D-알고리즘 등을 利用하여 實在回路의 試験을 하는 것이다.<sup>[2, 3]</sup> 마이크로프로세서와 같은 超高集成回路들은 内部의 複雜性과 内部構造에 대한 產業上의 秘密로 정확한 論理設計圖를 알 수 없기 때문에 구조 시험보다는 機能試驗을 主로 하고 있다.<sup>[4, 5, 6]</sup>

마이크로프로세서 試験을 위한 일반적인 모델은 크게 데이터 처리부와 제어부로 兩分시킬 수 있다.<sup>[7]</sup> 데이터 처리부에는 데이터 저장요소(버퍼, 레지스터, 화

\*正會員, 慶北大學校 工科大學 電子工學科

(Dept. of Electron. Eng., Kyung Pook National Univ.)

接受日字 : 1983年 11月 28日

일 등), 데이터 처리요소(ALU, 쉬프터, 비교기 등), 및 데이터 전달요소(버스, MUX, DEMUX)들이 서로 회로網을構成하고 있다. 制御部는 命令語를 解讀하여 수행에 필요한 일련의 制御信號를 發生시킨다.

本論文에서는 마이크로프로세서 제어부를 고장이 없는(fault free) 상태로假定한 후, 데이터 처리부의 고장검출을 위한試験을 위한 시험명령어 셉을 구하는過程을提示하였다. 이를 위한基本資料는 user's maunal로부터 얻을 수 있는 마이크로프로세서의機械的構成圖(functional block diagram) 및命令語(instruction set)를利用하였다.

最適試験命令語 셱은 각 명령어에 대한試験處理線圖(test execution graph)를構成한 후, 이중必須線圖(essential graph) 및適任線圖(eligible graph)를選擇하여 구성시킬 수 있었다.

本研究의對象 마이크로프로세서는 INTEL 8048이며 이에 대한 최적시험명령어 셱을구하는과정을설명하고있다. 本論文에서 提案한方法은 일반적인마이크로프로세서에適用할 수 있다.

## II. 마이크로프로세서의 記述

마이크로프로세서의試験方法을 일반화하기 위하여는 마이크로프로세서의 기술(description)을表準化하여야 하며, 이를 위하여 마이크로프로세서의 기계적 구성도와 명령어 셱을利用할 수 있다. 命令語의 수행은 실제 하드웨어로處理되기 때문에 명령어만을 가지고機能試験을 위한試験處理線圖를그릴 수 있다.

[定義1] 試験處理線圖는 아래의 요소로構成된다.

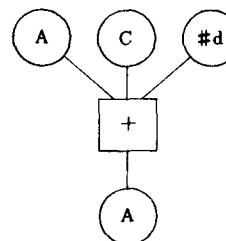
- i) 命令語에 의해 사용되어지는記憶素子는 1型頂點(vertex)으로 나타낸다(圓形으로 표현).
- ii) 命令實行에 수행되는 마이크로 오퍼레이션은 2型頂點으로 나타낸다(矩形으로 표현).
- iii) 1型頂點이 2型頂點에 의해處理될 때는 그 사이에弧(arc)가 존재한다.
- iv) 1型頂點中 入力위치에 있는頂點은 소스(source)라하고 出力위치에 있는정점을 싱크(sink)라 한다.
- v) 시험처리선도가 한 개의完全連結成分으로 구성되어 있으면單數(simple) 試験處理線圖이고, 그렇지 않으면複數(multiple) 試験處理線圖라 한다.

例)

시험처리선도는 INTEL 8048의 명령어들을 이용하여 다음과 같이 설명되어진다.

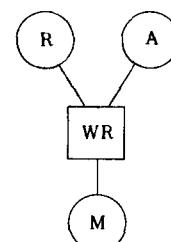
### 1) 단수시험처리선도

ADDC A, #data



(a)

MOVX @R, A



캐리 비트가 누적기에 더해진 후 다시 적치 데 이터 값이 더해지며 결과는 누적기에 남는다.

\* 외부 기억장소에 데이터를 넣거나 읽어내는 명령은 write 및 read로설명되며, 이때 기억번지는 왼쪽 정점에 넣는다.

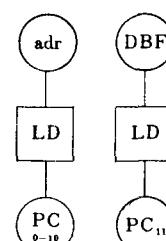
(b)

그림 1. 단수시험처리선도의 예

Fig. 1. Example of simple test execution graph.

### 2) 복수시험처리선도

JMP addr

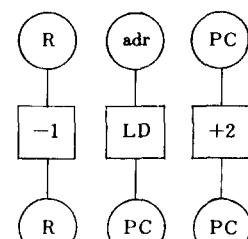


PC<sub>0~10</sub>에 어드레스 값이 저장되고 PC<sub>11</sub>에는 DBF 값이 저장된다.

\* DBF는 memory bank flip-flop이다.

(a)

DJNZ R, addr



1만큼 감소된 레지스터 값이 '0'이면 다음 명령어를 수행하고, '0'이 아니면 어드레스가 가리키는 명령어를 수행한다.

\* 조건 명령어인 경우 제어부는 fault free이므로 두 경우의 데이터 처리를 동시에 고려한다.

(b)

그림 2. 복수시험처리선도의 예

Fig. 2. Example of multiple test execution graph.

### III. 試験命令語의 選定

각 명령어에 대한 試験處理線圖는 각 명령어가 가지 고 있는 構造 및 機能에 따라서 必須線圖(essential graph), 適任線圖(eligible graph) 및 除去線圖(eliminable graph)로 분류된다.

[定義 2] 必須線圖란 같은 機能을 가진 명령어의 線圖에서 構造의으로나 機能의으로 도미넌트(dominant) 관계에 있는 線圖를 말한다. 다시 말하여 같은 기능을 가진 명령어의 線圖  $C_0, C_1, \dots, C_i \dots$ 가 있을 때  $C_i$  는  $\{C_0 \cup C_1 \cup \dots \cup C_{i-1}\}$ 의 線圖들에 대하여 구조적으로나 기능적으로 포함관계가 없을 때 必須線圖가 된다.

[定義 3] 適任線圖란 같은 기능을 가진 명령어들의 線圖中 기능은 같지만 구조가 다른 선도중에서 最小試験命令語의 構成에 適任인 線圖를 말한다.

適任線圖에 대한 우선순위는 구조가 복잡한 順, 사이클 시간이 짧은 順으로 한다.

[定義 4] 除去線圖란 같은 기능을 가진 명령어들의 線圖中에서 그 구조와 기능이 必須線圖 및 適任線圖 들에 완전포함 관계에 있는 線圖를 말한다.

例)

1) 필수선도 및 제거선도

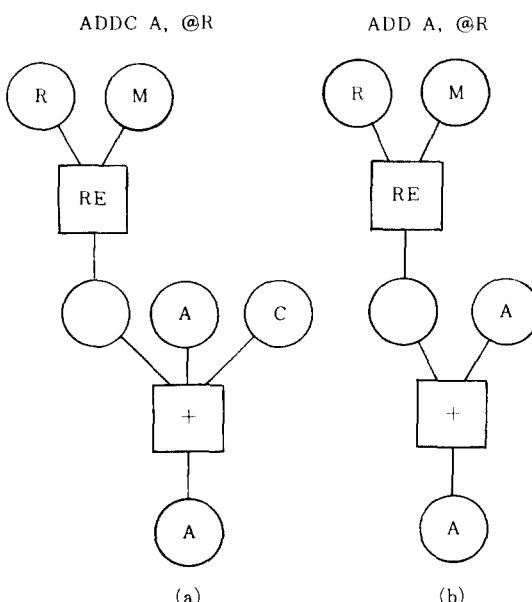


그림 3. 필수선도(a) 및 제거선도(b)

Fig. 3. Essential graph (a) and eliminable graph (b).

(a) 線圖는 (b) 線圖의 기능 및 구조를 포함하므로 필수선도이며 (b) 線圖는 제거선도이다.

### 2) 적임선도

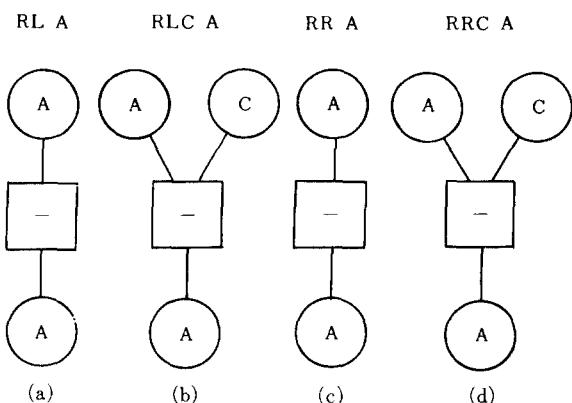


그림 4. 적임선도(b, c 또는 a, d)

Fig. 4. Eligible graph(b, c or a, d).

Shift left 및 shift right의 명령어는 같은 구조를 가지고 있으므로 適任線圖로 RLC A, RR A(또는 RL A, RRC A)가 選定된다.

最適試験命令語 셉은 아래의 3 단계 과정을 통해서 구해진다.

- 1) 마이크로프로세서 명령어를 기능적으로 분리하여 그룹별 試験處理線圖를 作成한다.
- 2) 각 그룹별 線圖로부터 必須線圖, 適任線圖 및 除去線圖를 구한다.
- 3) 全体 그룹에 대한 필수선도, 적임선도들로부터 最適의 試験命令語 셉을 구한다.

위의 과정에 의해서 최적시험명령어 셉이 결정되면 이를 이용하여 線圖上의 頂點들과 마이크로 오퍼레이션을 試験하는 試験發生 과정이 必要하며, 本研究에서는 이 과정을 省略하지 않았다.

### IV. INTEL 8048을 위한 試験命令語

INTEL 8048의 모든 명령어는 1개 혹은 2개의 바이트(byte)로 되어 있으며, 70% 이상이 1개 바이트로 구성되고 있다.

또한 모든 명령어는 1개 혹은 2개의 사이클로 처리되며 6 MHz XTAL을 使用할 경우  $2.5\mu s$  혹은  $5\mu s$ 로 命令이 遂行된다.

이들 명령어는 총 96개로서 處理機能에 의해 accumulator, input/output, register, branch, subroutine, flag, data move, timer/count 및 nop의 10가지로 分類시킬 수 있다. 最適의 試験命令語 셉은 10가지로 기능분류된 명령어들의 시험처리선도로부터 구하게 되며 각 그룹에 대한 必須線圖, 適任線圖 및 除去線圖를 구하면 第 1과 같다.

## 五 1. INTEL 8048 命令語의 分類

Table 1. Classification of INTEL 8048 instruction set.

| 區分            | 必須線圖命令語  | 適任線圖命令語  | 除去線圖命令語  |
|---------------|--|--|--|
| ACCU-MULAT-OR | ADDC A, @R<br>(INC A)<br>(DEC A)<br>CLR A<br>CPL A<br>DA A<br>SWAP A | ADD A, R<br>(ANL A, @R)<br>(ORL A, #data)<br>XRL A, R<br>RLC A<br>RR A   | ADD A, @R<br>ADD A, #data<br>ADDC A, #data<br>ADD A, R<br>ANL A, R<br>ANL A, #data<br>ORL A, R<br>ORL A, @R<br>XRL A, @R<br>XRL A, #data<br>RL A<br>RR C A |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
| INPUT/OUTPUT  | ANL P, #data<br>ORL BUS, #data<br>ANLD P, A                          | iN A, P<br>OUTL P, A<br>ORL P, #data<br>iNS A, BUS<br>OUTL BUS, A<br>ANL BUS, #data<br>ORLD P, A<br>MOVD A, P<br>MOVD P, A                               | iN A, P<br>OUTL P, A<br>ORL P, #data<br>iNS A, BUS<br>OUTL BUS, A<br>ANL BUS, #data<br>ORLD P, A<br>MOVD A, P<br>MOVD P, A                                 |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
| REG-ISTER     | DEC R<br>INC @R  |  | INC R  |
| BRANCH        | JMP addr<br>JMPP @A<br>DJNZ Rr, addr                                 | JC addr<br>JNC addr<br>JZ addr<br>JNZ addr<br>JTO addr<br>JNTO addr<br>JTI addr<br>JNTI addr<br>JFO addr<br>JFl addr<br>JTF addr<br>JNI addr<br>JBb addr | JC addr<br>JNC addr<br>JZ addr<br>JNZ addr<br>JTO addr<br>JNTO addr<br>JTI addr<br>JNTI addr<br>JFO addr<br>JFl addr<br>JTF addr<br>JNI addr<br>JBb addr   |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
|               |  |  |  |
| SUB-ROUTINE   | CALL addr<br>RETR  |  | RET  |
| FLAG          |  | CLR C<br>CPL F0<br>CLR F1  | CPL C<br>CLR F0<br>CPL F1  |
|               | MOV P3 A, @A<br>MOVX A, @R   | MOV A, @R<br>MOV R, #data  | MOV A, #data<br>MOV R, A   |

| 區分          | 必須線圖命令語   | 適任線圖命令語 | 除去線圖命令語       |
|-------------|---|---------|---------------|
| DATA MOVE   | MOVX @R, A<br>XCHD A, @R<br>XCH A, @R<br>XCH A, R<br>MOV A, PSW<br>MOV A, R<br>MOV PSW, A<br>MOV @R, A<br>MOV @R, #data |         | MOV @R, #data |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
| TIMER/COUNT | MOV A, T<br>MOV T, A<br>STRTR R<br>STRTR CNT<br>STOP TCNT<br>EN TCNTI<br>DIS TCNTI                                      |         |               |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
| CONT-ROL    | EN I<br>DIS I<br>SEL MBO<br>SEL MBI<br>SEL RBO<br>SEL RBI<br>ENTO CLK   |         |               |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
|             |   |         |               |
| NOP         | NOP   |         |               |
| 合計          | 41(43)  | 9(11)   | 52            |

표에서 보듯이 필수선도는 43개이며, 적임선도 명령어는 11개로試驗命令語는總 54개가 된다. III章의過程3에 의해, 위에서 얻어진 전체 명령어에 대한 두 종류의線圖命令語는 다시 고려되어지며 除去시킬 수 있는 명령어를 찾게 된다.

전체 명령어에 대한 必須線圖命令語와 適任線圖命令語 중 AND, OR, INC, DEC 명령어는 각각 2개씩 있으며, 그중 도미넌트한 선도는 ANL P, #data, ORL BUS, #data, DEC R 및 INC @R이 되어, 누적기를 이용한 명령어의 그룹중 INC A, DEC A, ALN A, @R 및 ORL A, #data가 除去된다. 따라서 最終的인 最適試驗命令語 셉은 50개로決定된다.

## V. 結論

마이크로프로세서의複雜性때문에 시험셋(test set)의選定은使用者가一般的으로 구할 수 있는情報(命令語 및 機械的構成圖)를 利用하였다.

마이크로프로세서의 데이터 처리부의 試驗을 위해서 명령어 셉에 대한 試驗處理線圖를作成할 수 있으며,

機能別로 그룹을 이룬 시험처리선도로부터 必須線圖, 適任線圖 및 除去線圖를 區分하여 最適試驗命令語 셉을 구하는데 利用하였다.

INTEL 8048 마이크로프로세서의 命令語들은 總 96 개로서 機能에 의해 10개의 그룹으로 나뉘며, 54개의 必須線圖 및 適任線圖로부터 最終的으로 選定된 試驗命令語 셉은 50개가 되었다. 선정된 50개의 명령어는 96개의 명령어에 대한 機能을 모두 포함하기 때문에 試驗에 걸리는 시간을 줄일 수 있어 효과적으로 이용될 수 있다.

본 研究에 의하여 선정되어지는 試驗命令語들은 그 수가 적기 때문에 짧은 시간 동안에 칩에 대한 기능시험을 할 수 있다. 따라서 制作中의 칩에 대한 시험은 물론 본 칩을 이용하는 시스템의 시험도 시스템의 여유시간(idle time) 동안에 이를 수 있다.

本 研究에서 提示한 方法은 시스템의 制御부가 고장이 없는(fault free) 상태의前提 아래 데이터 처리부의 機能시험을 위한 시험명령어 셉을 구하고 있으며, 앞으로는 구해진 시험명령어 셉에 대한 記憶素子 및 마이크로 오퍼레이션의 精密試驗을 行하기 위한 試驗發生過程(test generation procedure)에 대한 研究가 必要하리라 생각한다.

## 參 考 文 獻

- [1] W. Barraclogh, "Techniques for testing the microcomputer family," *Proc. IEEE*, vol. 64, pp. 943-950, 1976.
- [2] B. Courtois, "On line oriented functional testing of control sections of integrated CPUs," *Proc. Euromicro-7*, pp. 221-231, 1981.
- [3] 김충기, "MOS technology development," 전자기술연구소 연구보고서, pp. 243-325, 1983.
- [4] M. Annaratone, "An approach to functional testing of microprocessor," *Proc. FTCS-12*, pp. 158-164, 1982.
- [5] C. Robach, "Application oriented microprocessor test method," *Proc. FTCS-10*, pp. 121-125, 1980.
- [6] R. Chantal, "Microprocessor functional testing", *IEEE Test Conference*, pp. 433-443, 1980.
- [7] S. Thatte, "A methodology for functional level testing of microprocessors," *Proc. FTCS-8*, pp. 90-95, 1978.