

※ 본 논문은 1984년 하반기 학술진흥재단의 연구비 지원에 의하여 연구되었음.

마이크로 컴퓨터를 이용한 차량용 엔진 제어에 관한 기초 연구 (A Study on Engine Control System Using Micro-Computer)

강 기 문,* 전 병 실,* 황 준 택*
Ki Moon Kang, Byoung Sil Chon, Jun Taek Hwang.

Abstract

In order to control ignition advance angle, this system is designed with Z-80 CPU, CTC (Counter Timer Circuit), PIO (Parallel Input Output), A/D Converter and Memory, etc.

Serial pulses from speed sensor and analog voltage from pressure sensor are converted to digital data.

In order to reduce the error of ignition advance angle output, the reference of ignition advance angle output is set 56.25 before TDC (Top Dead Center).

The table of ignition advance angle and program which have a main routine and subroutines are written into ROM (1 K-byte).

The experimental result of this system is correspondent to the theoretical values of proposed ignition advance angle table.

This system can be utilized to any other type of 4 cylinder vehicles for advance angle control by changing software.

1. 서 론

1974년에 미국의 Intel회사는 4 bit 마이크로프로세서로 시판하기 시작하자 마자 General Motors(GM)에서는 "Computer on a Chip" 이라는 Alpha IV Vehicle System⁽¹⁾ 계획을 추진하였으며, 1977년 GM은 Microprocessed Sensing and Automatic Regulation(MISAR)⁽²⁾ spark timing system을 Oldsmobil Toronado 차종에 처음으로 부착하였다.

1978년에 Ford자동차 회사에서 Electronic Engine Control (ECC)⁽³⁾ 시스템을 Lincoln Versailles 차종에 부착하였으며, 같은 해에 Chrysler 자동차회사에서는 Spark Control Computer (SCC)⁽⁴⁾ 시스템을 부착하였다.

1980년 경에는 세계 자동차들의 25%, 1983년 경에는 50% 정도가 전자식 엔진제어⁽⁵⁾를 행하고 있다. 그 후 선진국에서 엔진성능 향상을 위한 많은 연구^(1~20)가 진행되고 있으며 그 기능 역시 다양화 되어가고 있다.

마이크로 컴퓨터를 자동차의 엔진제어에 이용하면 소프트웨어(Software)의 변경으로 제어의 상태를 변경시킬 수 있고, 최적의 소프트웨어로써 종래의 기계적 한계성을 극복할 수 있다. 따라서 선진 각국에서는 자동차의 성능 향상을 위해 마이크로 컴퓨터를 이용하고 있다.

그러나 국내에서는 1980년 부터 마이크로 컴퓨터를 이용한 자동차의 엔진 제어에 관한 연구가 진행되고 있으나,⁽¹⁹⁻²¹⁾ 아직 초기단계로 실용화되지 못하고 있는 실정이다. 따라서 본 논문에서는 자동차의 엔진 점화 진각 제어에 마이크로 컴퓨터를 이용하여 실용화시킬 수 있는 방안으로 Z-80 주변소자인 Z80-CTC (Counter Timer Circuit), Z80-PIO (Parallel Input Output), A/D (Analog/Digital Converter)⁽²²⁻²⁷⁾을 이용하여 자동차의 속도와 흡기 다기관 압력을 센서(Sensor)로 감지하여 SRC-80 마이크로 컴퓨터와 인터페이스시켜 속도와 압력에 따른 최적의 점화 진각 데이터(Data)를 결정하여 정확한 시기에 출력시킬 수 있는 시스템(System)을 구성하였다.

본 연구에서는 최적 점화 진각 데이터(부록 II, III 참조)는 "High-Line" 자동차를 모델⁽⁸⁾로 하여 4실린더 차량으로 고려하여 실험하였다. 여기에서 속도 센서는 Z80-CTC의 채널 3개를 종속으로 연결하여 각각의 채널 데이터를 SRC-80 마이크로 컴퓨터에 입력시키고, 흡기 다기관의 압력은 8비트 A/D 변환하여 Z80-PIO를 이용하여 입력시키고, 점화 진각은 Z80-CTC 채널 2개를 종속으로 연결하여 사용하였다.

또한 엔진의 속도 340RPM마다 최적 점화 진각을 ROM에 기입하여 주었다. 따라서 900-6,000RPM 사이에서 15개의 직선식을 이용하여 점화 진각을 구하였기 때문에 메모리 용량을 줄여줄 수 있었다. 각각의 엔진실린더(Cylinder)에 대해 점화 진각 출력의 기준을 TDC(Top Dead Center)전 56.25°에 설정하여 점화 진각 출력오차를 줄여주었다.

이 시스템은 소프트웨어의 변경으로 점화 진각 시기를 바꿀 수 있고, 최적의 점화 진각 데이터가 주어질 경우 엔진의 성능은 향상될 수 있다.

본 연구를 위하여 사용된 SRC-80 마이크로 컴퓨터는 싱글카드(Single card)마이크로 컴퓨터로써 CPU는 Z80-CPU이고, 3K의 ROM (Read Only Memory)과 4K의 RAM (Random Access Memory)을 실장할 수 있는데 본 시스템에서는 1K의 ROM과 약 40byte의 RAM을 이용하고 있다.

2. 점화진각 제어장치의 하드웨어 (Hardware)

그림 2.1은 마이크로 컴퓨터를 이용한 점화 진각 제어 시스템의 전체적인 구성도를 보인다.

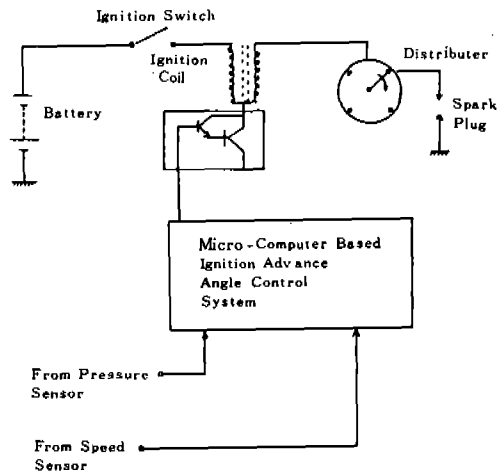


Fig. 2.1. Micro-Computer Based Ignition Advanced Angle Control System.

속도 센서와 압력센서는 Microcomputer based Ignition Advance Angle Control System (이하 MC-IAACS로 약함)으로 연결되고, 이 시스템은 트랜지스터로 구성된 점화 회로에 연결되어 있다. 이 시스템을 동작시키기 위해 점화 스위치를 연결하면 MC-IAACS에서 점화 펄스가 Low로 출력될 때마다 점화 회로가 off되어 1차 점화 코일에서는 자기유도 현상을 일으키고, 이와같은 철심에 감긴 2차점화 코일에서는 상호유도 현상으로 유기 기전력이 발생한다. 이렇게 발생된 유기 기전력은 분배기를 통하여 각각의 실린더의 점화 플러그로 전달되며, 방전 불꽃이 발

생하여 공기와 혼합되어 압축된 연료를 연소시키고 이때 발생하는 폭발 에너지로 엔진의 크랭크축을 회전시킨다.

2.1. 블록다이어그램

그림 2.2는 SRC-80마이크로 컴퓨터를 이용한 시스템 블록 다이어그램이다. 압력센서 인터페이스 장치는 Z80-PIO(이하 PIO로 약함) 와 A/D 변환기로 구성되어 있고, 압력센서로부터 압력변화에 따라 입력되는 아나로그 값을 A/D 변환하여 SRC-80으로 입력시킨다. 속도센서 인

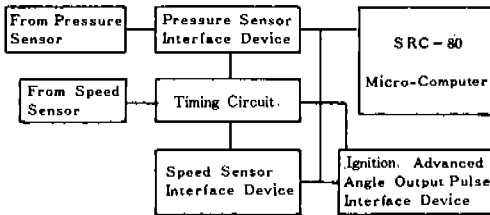


Fig. 2.2. Micro-Computer Based Ignition Advance Angle Control System Block Diagram.

터페이스 장치는 Z80-CTC(이하 CTC로 약함)로 구성되어 있고, 속도센서로부터 입력되는 직렬펄스를 이용하여 CTC를 동작시켜서 정해진 기간동안에 카운트한 값을 SRC-80으로 입력시킨다. 또한 점화 진각출력 펄스 인터페이스 장치는 압력 및 속도센서 인터페이스 장치로부터 입력된 데이터를 이용하여 SRC-80 마이크로 컴퓨터는 점화 진각 데이터를 결정하여 점화 출력장치의 타이머에 로드(load)시켜 점화펄스를 출력시킨다.

타이밍 회로에서는 속도센서로부터 입력되는 직렬펄스를 이용하여 각각의 인터페이스장치에 필요한 펄스를 만들고 있다.

2.2. Timing Diagram

점화 시기제어 시스템에 이용되는 timing diagram을 그림 2.3에 보인다.

파형 1은 속도 센서로부터 입력되는 직렬 속도 펄스로서 캐旭 1회전당 16개의 펄스가 입력될 정도로 하며, 파형 2,3,4,5,6은 파형 7,9,10,11,12를 만들기 위해 파형 1을 각각 2,4, 8,1632분

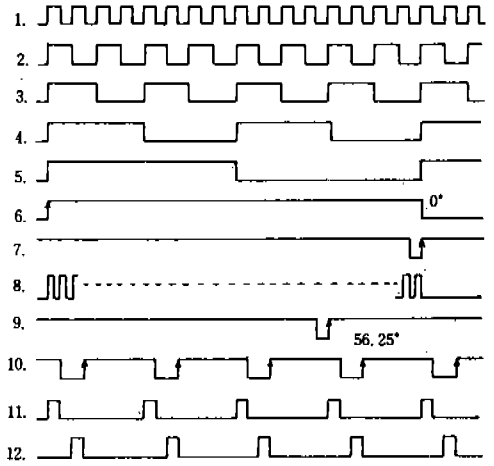


Fig. 2.3. Timing diagram

주한 파형이다.

파형 6은 low에서 high로 상승시 속도센서 인터페이스 장치의 카운터를 동작시켜 주고, 파형 7은 PIO $\overline{B}STB$ 를 이용하여 low에서 high로 상승시 CPU에 인터럽트를 요구하여 속도센서 인터페이스 장치의 카운터에서 카운트한 값을 SRC-80으로 입력시킨다.

파형 8은 속도센서 인터페이스 장치에서 카운트한 파형이다.

파형 9는 A/D 변환기의 \overline{SS} 단자에 가하여 주는 펄스로서 low에서 high로 상승시 압력센서로부터 입력되는 아나로그 값을 A/D 변환하여 주고, A/D 변환 완료시 Z80-PIO의 \overline{ASTB} 에 연결된 \overline{CC} 단자를 이용하여 CPU에 인터럽트를 요구하여 압력 데이터를 SRC-80으로 입력시킨다.

파형10은 점화 진각을 출력시키기 위해 점화 진각 출력장치의 타이머를 동작시키기 위한 파형으로 low에서 high로 상승시 타이머를 트리거 한다.

파형11은 영(zero)도에서 점화 진각을 출력시키기 위한 파형이다.

파형12는 점화 진각 출력장치의 래치 회로에 사용되는데 low에서 high로 상승시 래치를 풀어준다.

2.3. 압력센서 인터페이스 장치

압력센서 인터페이스 장치는 흡기 다기관

력변화를 압력센서를 이용하여 아나로그 전압으로 변환하고, 변환된 아나로그 전압을 A/D 변환하여 마이크로 컴퓨터에 입력시키는 장치이다.

그림 2.4는 본 시스템에서 이용된 압력센서 인터페이스 장치이다.

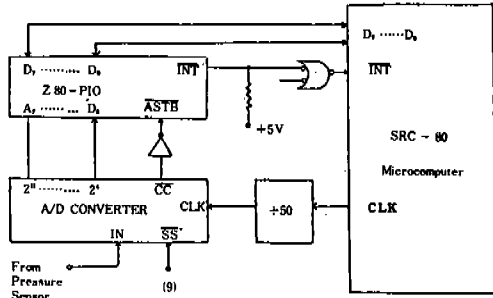


Fig. 2.4. Pressure sensor interface device.

압력센서로부터 압력변화에 따라 입력되는 0 - 5V의 값을 그림 2.3에서 보인 파형 9를 이용하여 low에서 high로 상승시 A/D 변환을 시작하고, A/D 변환이 완료되면 \overline{CC} 단자는 high에서 low로 하강하므로, \overline{ASTB} 단자는 low에서 high로 되어 인터럽트를 요구하여 A/D 변환된 데이터값을 PIOA포트를 통하여 SRC-80에 입력시킨다.

여기에서 사용된 A/D 변환기는 12비트로 구성되어 있어 있는데 2^{12} 단자로부터 8 bit를 이용하고 있다. 그 이유는 Z-80 CPU가 8비트 데이터를 처리하기 때문이다. A/D 변환기의 CLK 단자에 입력되는 클럭은 SRC-80 마이크로 컴퓨터의 클럭 2.5MHz를 50분주하여 사용하였다.

2.4. 속도센서 인터페이스 장치

속도센서 인터페이스 장치는 엔진의 캠축의 1회전동안 카운터를 동작시켜서 1회전동안 카운트된 값을 SRC-80마이크로 컴퓨터에 입력시키는 장치이다. 그림 2.5는 본 시스템에서 이용된 속도센서 인터페이스 장치이다.

속도센서로부터 캠축 1회전당 16개의 펄스가 입력된다. 입력된 펄스를 16분주하면 캠축 1회전당 1주기를 갖는 펄스가 된다. 이 펄스는 그림 2.3에서 보인 파형 5에 해당되고, 파형 6이

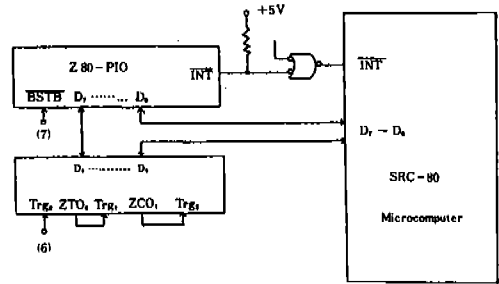


Fig. 2.5. Speed sensor interface device.

low에서 high로 상승시 속도센서 인터페이스 장치에 이용된 CTC 1의 채널의 타이머를 트리거하게 된다. CTC 1은 3개의 채널을 이용하는데 채널 0는 타이머로 사용되고, 채널 1과 채널 2는 카운터로 이용되며, 이들은 서로 종속으로 연결한다. 파형 5가 다시 1주기가 되는 지점, 즉 그림 2.3에서 보인 파형 7은 PIO \overline{BSTB} 단자를 low에서 high로 상승시켜 CPU에 인터럽트를 요구한다. 여기에서 PIO포트는 출력모드로 설정했다. CPU가 인터럽트를 받아들여서 수행하면 채널 0,1,2의 값을 읽어서 RAM영역에 보관하고 CTC 1을 리셋시킨다. 여기에서 엔진의 회전수와 CTC 1의 카운트값과의 관계를 알아보면 다음과 같다.

엔진의 회전수와 캠의 회전수를 각각 N_e (r.p.m), N_c (r.p.m)라 하고 캠의 주기와 마이크로 컴퓨터에서 발생하는 clock ($f = 2.5 \text{ MHz}$)의 주기를 T_{cam} , T_{clock} 라 하면,

$$N_c = N_e / 2$$

가 되며 r.p.m을 r.p.s (r.p.m - 60 r.p.s)로 고쳐쓰면 캠의 주기와 클럭의 주기는,

$$T_{cam} = 1 / N_c = 120 / N_e \text{ (sec)}$$

$$T_{clock} = 1 / f = 1 / 2.5 \times 10^6 \text{ (sec)}$$

이므로 캠의 1주기 동안, CTC 1에서 클럭주파수를 카운트한 값 C_p 을 구하기 위하여 각각의 주기를 계산하면 되므로,

$$C_p = T_{cam} / T_{clock} = (120 / N_e) / (1 / 2.5 \times 10^6) = (3.0 \times 10^8) / N_e \text{(1)}$$

이다.

2.5. 점화진각출력 인터페이스 장치

점화출력 장치는 압력센서 인터페이스 장치와 속도센서 인터페이스 장치에서 입력한 홀기 다기관 입력과 속도에 따른 데이터(부록Ⅱ, Ⅲ참조)로부터 정해진 점화진각 값은 점화출력 장치의 타이머에 입력하고, time zero시에 ZTO₀ 단자에서 출력되는 펄스를 단안정 멀티바이브레이터에 입력하여 점화출력 펄스를 얻는 장치이다. 그림 2.6은 본 시스템에서 이용된 점화진각 출력장치이다.

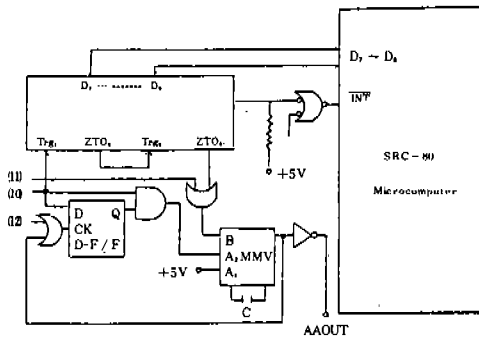


Fig. 2. 6. Ignition advanced angle output interface device.

CTC 2의 채널 0,1은 타이머로 이용되고 있고, 이들은 서로 중속으로 연결되어져 있다.

이들 타이머는 그림 2.3에서 보인 파형 11과 OR GATE를 통하여 단안정 멀티바이브레이터 B 단자에 입력되는데 A 2가 low이고 B 단자가 high로 상승시에 Q 단자에서 C값으로 결정된 duty값의 펄스가 출력된다. 그런데 그림 2.3에서 보인 파형 12로 항상 D-FF의 Q를 low로 하여주는데 D-FF의 D가 high상태(56.25도-11.25도) 일때 점화펄스가 출력(50도-0도)되며 D-FF의 Q는 high로 되어 단안정 멀티바이브레이터는 D 단자가 low로 될때까지 동작하지 않는다. 만일 CTC 2로부터 점화펄스가 출력되지 않으면 파형 11에 의해 영(zero)도에서 점화펄스가 출력되고 CTC 2에서 점화펄스가 출력되면 A 2는 high로 되어 영(zero)도에서 점화 펄스가 출력되는 것을 방지한다. 점화펄스의 폭은 C의 값으로 결정된다. 엔진의 RPM과 점화출력장치에서 카운트한

값과의 관계는 다음과 같다.

CTC 2의 타이머에서 카운트한 값을 C_A라 하고, 점화진각을 A도라 할때 CTC 2의 타이머는 TDC전 56.25도에서 동작하므로 다음과 같은 식을 얻는다.

$$C_p : C_A = 360^\circ : (56.25^\circ - A^\circ)$$

의 관계가 있으므로

$$C_A = C_p (56.25^\circ - A^\circ) / 360^\circ$$

이다. 뒷 식에 식(1)을 대입하면,

$$C_A = 833.333 (56.25^\circ - A^\circ) / N_e \dots\dots\dots (2)$$

가 된다.

3. 점화진각 제어장치의 소프트웨어 (Soft ware)

MC-IAACS에서는 1개의 메인루틴과 2개의 서브루틴, 3개의 인터럽트 서브루틴이 이용되고 있다.

3.1. 메인루틴

그림 3.1은 MC-IAACS에서 사용된 메인루틴의 흐름도이다. MC-IAACS에서는 I/O 장치에서 인터럽트 모드 2를 이용하고 있으므로 I 레지스터에 인터럽트 상위바이트 벡터값을 지정하고, 하위바이트 벡터값은 I/O장치에서 지정하여서 상위와 하위바이트 벡터를 조합하여 인터럽트 서브루틴의 시작번지가 들어있는 번지를 지정한다. 또한 압력데이터를 SRC-80 마이크로 컴퓨터에 입력하기 위해서 PIO를 이용하는데 필요한 동작모드를 설정한다. 여기에서는 인터럽트, A포트 입력, B포트 출력으로 설정하였고, 인터럽트 요구시 하위 벡터 값을 지정하기 위해 하위 벡터를 설정하여 주었다.

CALL INTLSP는 속도 데이터를 SRC-80으로 입력하기 위해 사용한 CTC를 초기화하여 속도 데이터를 받을 준비를 하여 준다.

점화진각의 계산은 엔진의 속도가 900rpm이상에서만 행하도록 하였다.

만일 엔진속도가 900rpm이하 일때는 CTC를 리셋(reset)시키고, 엔진속도가 900rpm 이상이

되기를 기다린다.

엔진속도가 900rpm 이상이 되면 압력 값과 엔진의 속도값을 계산하여 주어진 진각 테이블에서 점화 진각을 찾고, 이 진각을 이용하여 직선식을 만들어서 엔진의 속도에 따른 점화진각을 계산하게 된다.

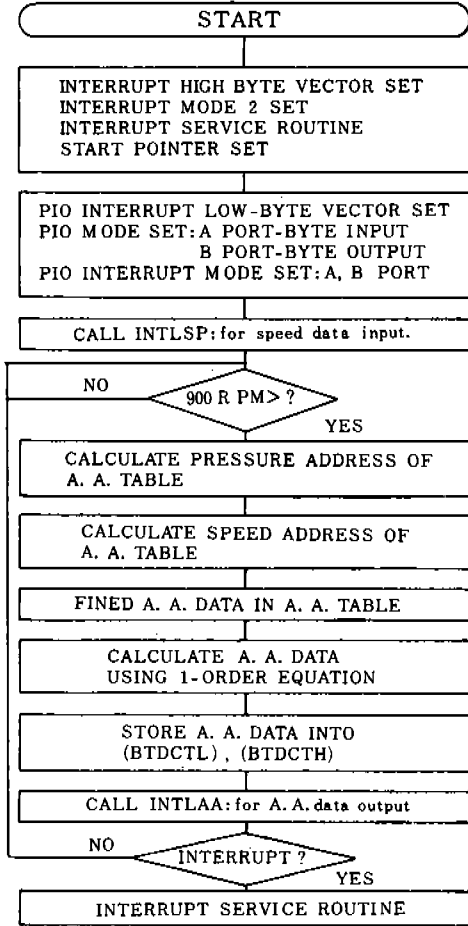


Fig. 3.1. Main Routine

계산된 진각은 각각(BTDCTL), (BTDCTH)의 RAM 영역에 저장하고 점화 출력을 하기 위해 CALL INTLAA을 행하게 된다.

이 루틴이 수행될때 인터럽트가 요구되면 프로그램 카운터는 인터럽트 서비스루틴으로 점프하고, 그렇지 않으면 계속하여 점화진각을 계산하는 루틴을 반복한다.

3.2. 서브루틴

3.2.1. 서브루틴

그림3.2는 MC-IAACS에서 사용된 서브루틴의 흐름도이다. INTLSP는 속도 데이터를 SRC-80으로 입력하기 위해 사용된 CTC1을 초기화하여 주기 위한 것으로 채널0의 타이머 시정수는 81트리거, CLK/16모드로, 이용되고 채널1, 채널2는 카운터로 사용되고, 시정수는 256으로 설정하여 이용하였는데 이때 CTC 1에서 카운트한 값은 식(1)로부터 구할 수 있다.

여기에서 채널 2의 카운터는 엔진속도가 900 rpm 이상 일때만 카운트된다.

INTLAA는 점화진각을 출력하기 위해 사용된 CTC 2를 초기화하여 주기 위한 것으로 채널 0에서는 인터럽트를 요구하기 때문에 하위바이트 벡터 값을 로드하여 주고, 인터럽트, 타이머, 시정수는(BTDCTL)에 있는 값으로 결정하여지며 TRG, CLK/16으로 설정하였다.

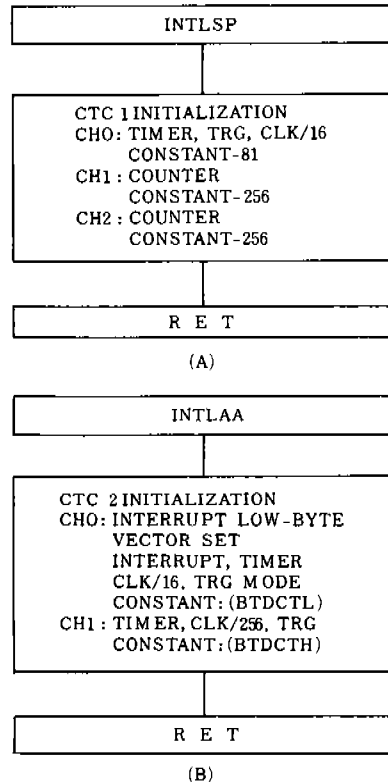


Fig. 3.2. Subroutine

또한 채널 1의 타이머 시정수는(BTDCTH)의 값으로 결정하여지고, TRG, CLK/256으로 설정하여 이용하고 있는데 이때 CTC 2의 타이머가 카운트할 값은 식(2)에서 구할 수 있다.

3.2.2. 인터럽트 서브루틴

그림3.3은 MC-IAACS에서 이용된 인터럽트 서브루틴을 흐름도로 보이고 있다.

PRSITD는 압력 데이터를 PIO A포트로 읽어 들어 (PRSDT) 번지의 RAM에 저장하기 위한 것이다.

SPDITP는 속도 데이터를 CTC 1의 채널 0, 채널 1, 채널 2에서 읽어 들여 각각 (SPDCTL), (SPDCTH), (SPDCTC) 번지의 RAM에 저장한 다음에 CTC 1을 리셋시키고, 새로운 속도 데이터 값을 읽기 위해 CALL INTLSP를 하여 CTC 1을 초기화하여 준다.

AAOITP는 점화 진각이 출력되었을때 다시 점화출력이 되는 것을 방지하기 위해 CTC2를 리셋시키고, 새로운 점화진각을 출력하기 위해 CALL INTLAA를 하여 CTC 2를 초기화시키고 있다.

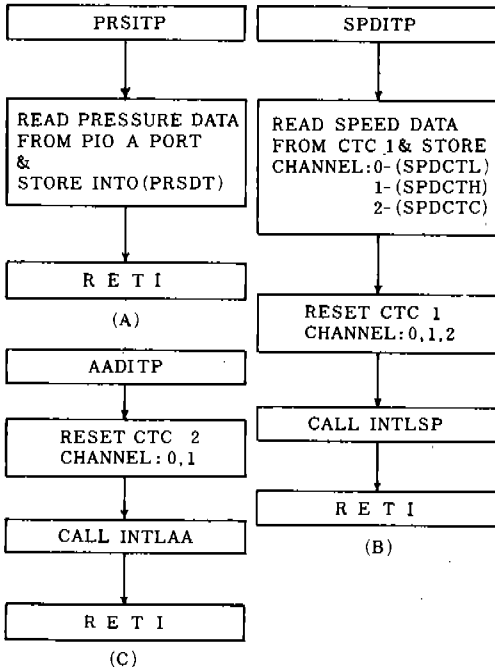


Fig. 3.3. Interrupt Subroutine

4. 실험 및 검토사항

자동차에 이 시스템을 장치하기 전에 실험실에서 엔진의 속도센서로부터 출력되는 펄스는 신호발생기의 주파수를 가변하면서 시뮬레이션을 행하였고, 압력센서로부터 출력되는 아나로그 전압은 가변저항을 이용하여 직류전압을 변화시켜 가면서 실험하였다.

사진4.1은 본 시스템을 동작시켜서 얻은 시스템의 각 단자의 파형을 보이고 있으며 파형(1)의 경우 파형 0은 사진4.1의 파형, () 내의 1은 fig.2.3의 파형으로 표현하여 비교하였다. 사진4.1의 a), b), c)에서 파형(1)은 엔진의 직렬 속도 펄스이고 파형 1(5)은 파형 0(1)를 16분주하여 캠축의 1회전을 한 주기로 나타내고 있다.

파형 2(7)는 속도센서 인터페이스 장치의 카운터가 동작하여 채널 1에서 카운트하는 펄스를 나타내고 있다. 파형 3은 CTC 1의 시정수를 81,16분주하였기 때문에 식(1)을 81×16으로 나눔으로서 카운트 파형의 갯수를 구할 수 있다.

즉, 엔진의 r.p.m이 2000, 4000, 6000일때,

$$\frac{3.0 \times 10^6}{2,000 \times 1,296} \approx 115$$

$$\frac{3.0 \times 10^6}{4,000 \times 1,296} \approx 57$$

$$\frac{3.0 \times 10^6}{6,000 \times 1,296} \approx 38$$

이 되어 사진4.1 a), b), c)의 파형 3과 일치한다.

파형 4(9)는 압력센서 인터페이스 장치의 A/D 변환기를 동작시켜 변환이 완료되면 이때 인터럽트를 요구하기 위한 파형이다. 파형 5는 CPU에 인터럽트를 요구하는 파형들이다. 파형 6(10)은 점화진각 펄스 출력장치의 타이머는 56.25도에서 트리거하는 파형이다.

파형 7은 파형 6(10)에 의해 타이머가 동작하여 타이머의 카운트 값이 0(zero)으로 되었을때 점화진각이 출력되는 것을 보여준다.

이 파형은 부록Ⅲ에서 보인 점화진각 테이블에 있는 값과 즉 엔진의 r.p.m이 2,000이고 압력이 131mmHg일때 점화진각 테이블 값은 34도를 보이고 있어서 사진 4.1 a)와 일치함을 알수 있다. 또한 엔진의 r.p.m이 4,000이고 압력이 350mmHg

일때 점화진각 테이블 값은 40도를 보이고 있어서 사진 4.1 b)와 일치함을 알수 있다. 엔진의 r. p. m이 6,000이고 압력이 613mmHg일때 점화진각 테이블 값은 33도를 보이고 있어서 사진 4.1c)와 일치함을 보이고 있다.

사진 4.1 d)는 진각 출력장치에서 사용된 램치 회로의 파형을 보이고 있으며, 파형 0과 1은 사진 4.1 a), b), c)의 파형과 같고, 파형 2는 앞 사진들의 파형 3에 해당되며 파형 3은 단안정 멀티 바이브레이터의 A 단자의 파형이고, 파형 4는 앞 사진들의 파형 7에 해당되고 파형 5는 앞 사진들의 파형 6에 해당되고 있다.

특히 파형 3은 A 단자의 파형으로서 점화펄스가 출력되면 A 단자는 high상태로 되어 동작하지 않음을 보여주고 있다.

또한 그림 4.2는 압력이 131mmHg, 350mmHg, 613 mmHg일때 엔진속도에 따른 이론적인 점화진각과 실험에서 구한 점화진각을 도표화하여 나타내고 있는데 거의 이론적인 값에 일치하고 있음을 보이고 있다.

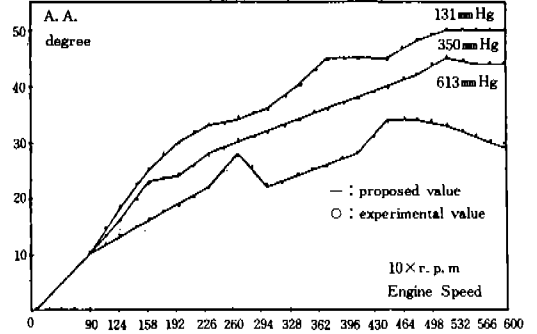
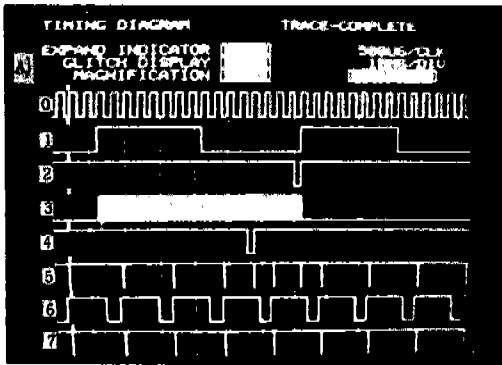
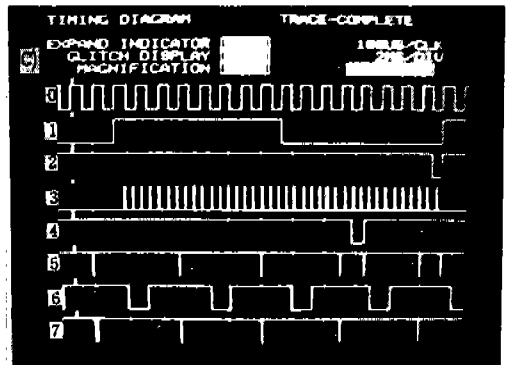


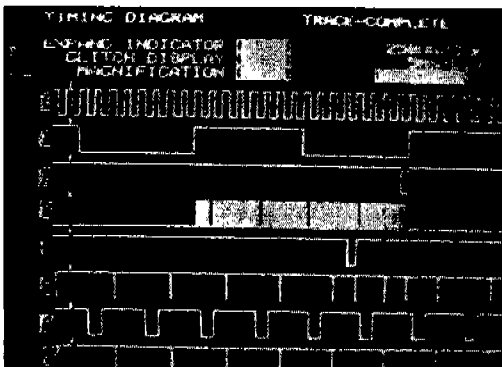
Fig. 4. 2 Comparison between proposed A. A & experimental A. A



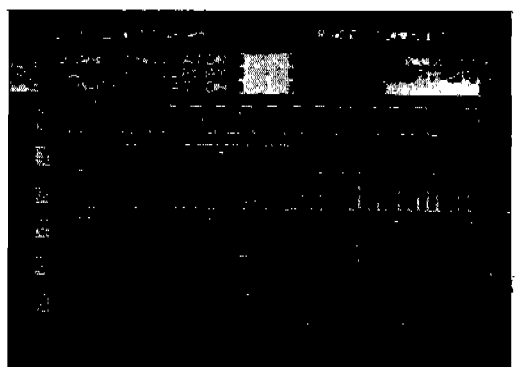
엔진속도 : 2,000RPM
압 력 : 131mmHg



엔진속도 : 6,000RPM
압 력 : 613mmHg



엔진속도 : 4,000RPM
압 력 : 350mmHg



엔진속도 : 5,000RPM
압 력 : 613mmHg

Photo. 4. 1. The Waveform of MC-IAACS on Engine Speed and Intake Manifold Absolute Pressure.

5. 결 론

자동차의 점화시기 제어를 행하기 위해 엔진의 흡기 다기관에 따른 압력에 따른 데이터를 입력하기 위해 Z80 주변소자 PIO와 CTC, A/D 콘버터를 이용하여 부록 I 과 같이 구성하고, 부록 III 에서 보인 진각데이터 테이블, 앞에서 보인 메인루틴 과 서브루틴을 ROM에 넣고 속도와 압력에 따른 점화진각을 출력시킨 결과 다음과 같은 결론을 얻었다.

1. 엔진 2,000rpm 압력 131mmHg일때, 점화진각 34도, 엔진 4,000rpm 압력 350mmHg일때, 점화진각 40도, 엔진 6,000rpm 압력 613mmHg일때, 점화진각 33도 일때 시뮬레이션한 결과 사진 4.1과 그림 4.2에서 보인 것처럼 이론적인 값과 거의 일치한 점화진각을 얻을 수 있었다.

2. 소프트웨어만 바꾸어 주면 4기통인 차종에 서는 차종에 관계없이 사용할 수 있다.

3. 실용화를 위해 센서개발과 국산차량의 점화진각 테이블이 필요하고, 국산자동차에 실장실험을 행하는 것이 남은 과제이다.

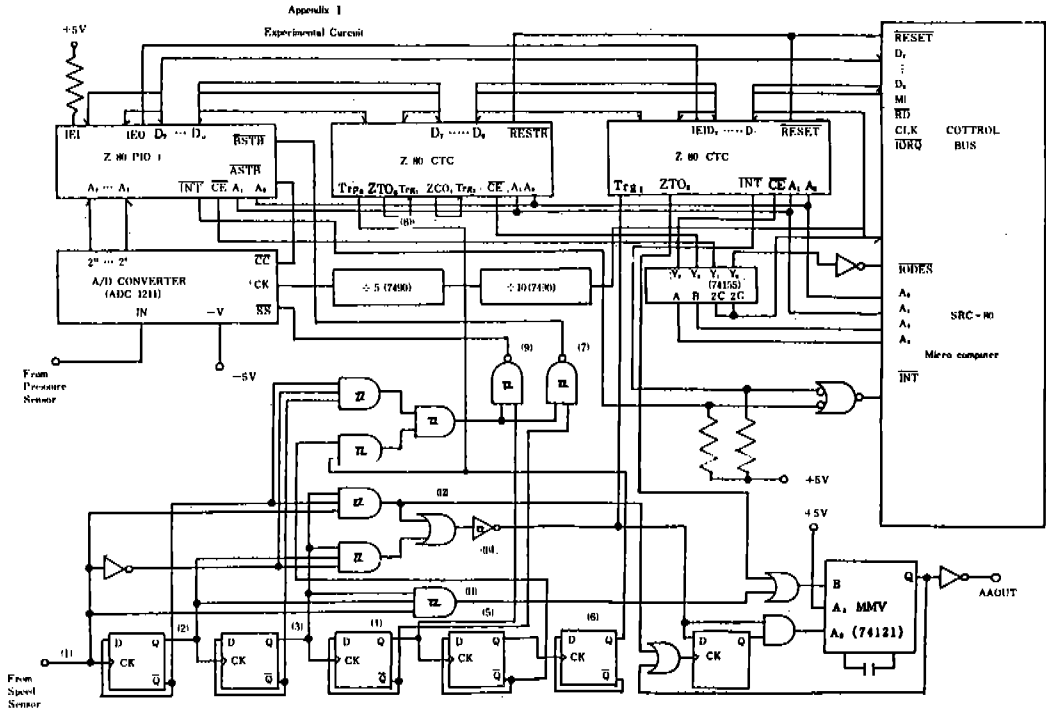
참 고 문 헌

1. T. O. Jones, R. L. Colling, "Application of Microprocessors to the Automobile", SAE Paper 750432 p. 66, 1975.
2. David F. Moyer, "Electronic Engine Control by an On-Board Computer," Proceedings of international conference on automotive Electronics and Electric Vehicles, SAE CATALOG number p. 68 page 5, 1976.
3. John W. Weil, "Digital Engine Management," Proceedings of international conference on automotive Electronics and Electric Vehicles SAE CATALOG number p-68, page 7, 1976.
4. E. W. Meyer, "The Electronic Lean Burn System," Proceedings of international conference on automotive Electronics and Electric Vehicles, SAE CATALOG number p-68, page 1, 1976.
5. J. E. Maund, "Current Status of Automobile Electronics in Europe," Automobile Electronics, IEE Conference Publication number 141, London, 1976.
6. Trevor O. Jones, "The Status of Automotive Electronics, IEE Conference Publication number 141, London, 1976.
7. Takio Kitano, "Current Status of Automobile Electronics in Japan," Automobile Electronics, IEE Conference Publication number 141, London, 1976.
8. H. Manger Development Trends for Battery Ignition Systems," Automobile Electronics, IEE Conference Publication number 141, London, 1976.
9. Rosslyn J. Cannon, "Applied Automotive Electronics, SAE Paper 780859 p. 137, 1978.
10. D. F. Hagen, "Electronic Engine Controls At Ford Motor Company" SAE Paper 780842, p. 59, 1978.
11. J. L. Webster, "Chrysler's digital spark timing Computer," SAE Paper 780844, 1978.
12. J. W. Hoard, "New State of the Art in Engine Control," SAE Paper 810061, 1981.
13. Ronald J. Freimark, "Cost Effective Engine Controls," SAE Paper 810157, 1981.
14. Mark B. Kearney, "Microprocessor Based Systems in the Automobile." SAE Paper 810160, 1981.
15. S. R. Bhot, "The control of ignition timing to achieve maximum fuel economy," Third international conference on automotive Electronics, I. Mech. E. Paper C174/81, 1981. London.
16. R. J. Freimark, "Engine Control with low-cost microcomputer," Third international Conference on automotive Electronics, I, Mech. Paper C174/81, 1981, London
17. R. Dell Acqua "Technological Approaches to ignition System in the European Market," SAE Paper 840449, 1984.
18. H. Laurent "Integrated Engine Control," S

AE Paper 840540, 1984.

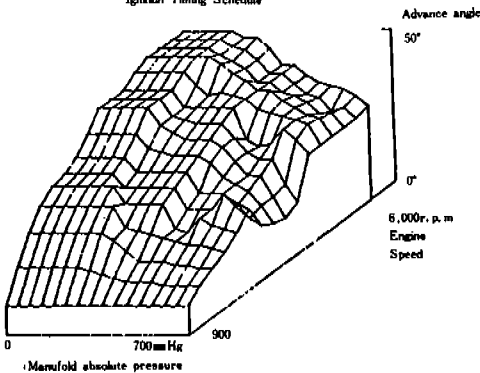
19. 김용립. "Microprocessor Based Ignition Timing Control" 한국과학원 석사논문, 1980.
20. 전병실, "자동차의 마이크로컴퓨터 응용에 관한 연구" 전북대학교 공업개발연구소, 제11집, 1981.
21. 전병실, "차량용센서 개발에 관한 기초연구" 전북대학교 논문집 제25집 자연과학편, 1983.
22. Zilog, "Microcomputer Components Data Book, Zilog.

23. Bruce A. Artwick, "Microcomputer interfacing," Prentice-Hall, 1980.
24. Douglas V. Hall, "Microprocessors and Digital Systems" McGraw-Hill, 1980.
25. National Semiconductor Manual, ACD 1211 data.
26. Zilog, "Z-80 Assembly Language Programming Manual" Zilog, 1980.
27. SRC-80A manual, Z-80 Microcomputer. SUNG MO Electronic Co. 1982.



Appendix II

Ignition Timing Schedule



Appendix III

Spark Timing BTDC(DEGREE)10 × r.p.m.

Speed	90	124	156	192	226	260	294	328	362	396	430	464	498	532	562	600
Pressure																
700	10	13	15	17	20	24	17	15	14	16	27	27	27	27	27	28
657	10	13	15	18	21	26	19	20	20	21	31	33	32	31	30	31
613	10	13	16	19	22	28	22	24	26	28	34	34	33	31	29	33
569	10	13	16	19	24	24	26	26	26	28	30	35	35	34	33	34
525	10	14	17	20	20	24	26	26	30	30	35	36	37	36	34	35
481	10	14	17	21	21	24	28	32	34	34	31	37	40	37	37	37
438	10	14	17	22	26	30	32	34	36	36	40	40	40	40	40	40
394	10	15	20	23	26	30	32	34	36	39	40	41	43	40	40	40
350	10	16	23	24	28	30	32	34	36	39	40	42	45	44	44	45
306	10	17	25	30	28	30	32	35	38	39	40	41	48	46	47	48
263	10	18	25	30	33	30	33	36	40	40	40	44	50	48	48	48
219	10	18	25	30	33	34	36	40	45	45	45	48	50	50	50	50
173	10	18	25	30	33	34	36	40	45	45	45	48	50	50	50	50
131	10	18	25	30	33	34	36	40	45	45	45	48	50	50	50	50
89	10	18	25	30	33	34	36	40	45	45	45	48	50	50	50	50
44	10	18	25	30	33	34	36	40	45	45	45	48	50	50	50	50