전방시현기 및 다기능시현기 구동을 위한 그래픽 영상생성기 설계 연구

Design of Graphic Generator for Driving HUD(Head-Up Display) and MFD(Multi-Function Display)

황 상 현* 이 재 억* 박 덕 배*
Hwang, Sang-Hyun Lee, Jae-Uk Park, Deok-Bae

ABSTRACT

This paper presents the design technology of a Graphic Generator which drives the embedded aircraft display equipments such as HUD(Head-Up Display) and MFD (Multi-Function Display) those provide pilot with the most important mission information.

The main issue of this design is how we can implement the real-time embedded graphic generator using a general purpose processor as a substitute for the obsolete the production of specific graphic processor in the military market. So we proposed two kinds of method that one is a software solution so called graphic kernel system, interpreting the display file, controlling the graphic system and pre-processing graphic primitives, the other is a hardware solution so called graphic engine, interpreting passed commands through the graphic kernel system, post-processing the looping calculation taking much of time as implemented by software. We have tested and verified the functionalities and the required performance of Graphic Generator.

주요기술용어 : Graphic Generator(그래픽 영상생성기), Avionic System Computer(항공전자시스템 컴퓨터), Graphic Primitive(그래픽 원시요소), Graphic Engine(그래픽 엔진), Cursive(커시브), Raster(라스터), Symbology Enhancement(심볼 고양)

1. 서 론

전투기와 같은 군용 항공기에서는 한 명 내지 두 명의 조종사가 항법 조종뿐만 아니라 적에 대한 공격 및 방어의 급박한 전투임무를 동시에 수행한다. 이러한 임무를 성공적으로 수행하기 위해 민첩하고 정교한 조종기술이 필요하며, 또한 자 항공기를 비롯한 주변의 상황에 대한 다양하고 복잡한 정보의 신속하고 정확한 인식 및 판단이 임무의 성공률을 높이는 큰 요소라 할 수 있다.

* 국방과학연구소 연구원

1980년대 이후 군사 선진국들은 제공권의 우위를 지키기 위해 항공전자 시스템의 정보처리 계통의 개발에 보다 많은 연구를 한 결과, 대체로 비슷한 형태의 조종석 환경을 개발하였다. 발전된 조종석 환경의 대표적인 것이 시현계통으로서 복잡한 아날로그 계기류를 단순화하여 몇 개의 큰 화면에 통합하는 글래스 콕핏(Glass Cockpit)화에 그 공통점이 있다. 이를 위해서 시현장비 및 항공기의 모든 센서들의 정보를 통합하고 처리하여 영상신호를 생성하는 항공전자시스템컴퓨터(ASC, Avionic System Computer)가 필요하다.

시현장비는 기능적으로 전방시현기(HUD, Head-Up Display) 및 다기능시현기(MFD, Multi-Function Display)로 분류된다. 전방시현기는 조종사의 전방 시야에 위치한 투명 반사경을 통해 외부를 주시하는 상태에서 문자 및 도형으로 단순화된 영상을 조사 및 반사시켜 정보를 제공하는 가장 중요한 시현장비이다. 다기능시현기는 항공전자 시스템의 관리 및 통제를 위한 시현장비로서 보다 다양하고 복잡한 정보를 시현한다. 다기능시현기의 영상 출력방식은 표준화된 라스터(Raster) 주사방식을 따르고 있고, 전방시현기는 태양광과 같은 외부의 입력광에 대향하더라도 영상의 인식이 가능토록 하기 위해 라스터 주사방식보다 광효율을 높일 수 있는 커시브(Cursive) 주사방식을 적용한 경우가 대부분을 차지한다.

ASC는 항법계통, 통신계통, 무장 발사계통, 시현계통, 항공전자 시스템 관리 및 통제계통을 관할하는 군용 항공기의 주 컴퓨터로서, 정보의 집중화 및 단순화에 의한 조종사의 상황 인식도 고취, 공대공/공대지 탄도계산에 의한 신속 정밀한 공격, 항공기 관리 및 통제기능등의 통합환경을 제공한다.

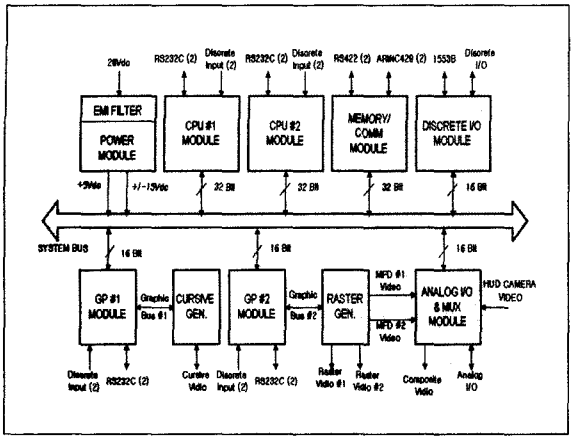
군용 시현장비의 인터페이스 규격은 표준화되는 추세에 있어, 이러한 입/출력 자원에 해당하는 장비의 국내 독자개발은 필수적이지는 않다. 그러나 ASC는 군용 항공기의 주 컴퓨터로서 항법계통, 통신계통, 무장 발사계통, 시현계통, 시스템 관리 및 통제계통을 관할하는 중추적인 역할을 담당하므로 인해 항공전자 시스템을 통합하기 위해서는 독자개발의 필요성이 절실하다 할 수 있다.

본 논문에서 군용 항공기의 주 시현장비인 전방시현기 및 다기능시현기를 구동하는 항공전자시스템컴퓨터의 그래픽 영상 생성기 모듈을 설계 및 구현하여 기능시험 및 비행시험을 통하여 그 성능을 확인함을 연구의 목적으로 하였다. 본 논문에서는 ASC 설계 개요, GP 모듈 하드웨어 설계, 라스터/커시브 그래픽 모듈 하드웨어 설계, 그래픽 소프트웨어 설계, 제작 및 시험등의 내용을 기술한다.

2. 항공전자시스템컴퓨터(ASC) 설계 개요

ASC는 항공기의 항법 및 무장제어를 위해 모듈화된 다목적 컴퓨터로서 다음과 같은 기능을 수행한다.

- 무장투하 및 항법시스템 통제
- MIL-STD-1553B 제어 및 각종 데이타 통신 제어
- 항법, 비행자세 조종 및 무장조준을 위한 계산 및 무장발사제어
- HUD 커시브 심볼 생성
- MFD 라스터 심볼 생성
- 시스템 비디오 영상처리, 혼합 및 관리
- 외부장비 인터페이스 제어



[그림 1] ASC 모듈 구성 및 데이타 흐름도

ASC는 GPS/INS/ADC 및 통신 항법시스템으로부터 입력된 자 항공기의 위치, 자세 및 외기 정보를 이용한 항법정보 통합처리, WCS(Weapon Control System)와 연동하여 무장관리 및 공격통제, UHF/VHF를 원격 제어하는 기능을 갖는 다목적 주 컴퓨터이다. ASC는 총 10개의 6U 보드를 탑재하고 있으며, 이들의 구성 및 관련 인터페이스는 그림 1과 같다.

시스템 버스는 VME 버스를 일부 변형하여 적용하였고, 그래픽 버스는 GP(Graphic Processor) #1모듈과 커시브 그래픽(Cursive Generator) 모듈, GP #2모듈과 라스터 그래픽(Raster Generator) 모듈 및 비디오 신호처리(Video Multiplex) 모듈의 연결을 위한 그래픽 전용버스이다.

CPU #1 및 #2 모듈은 운영 소프트웨어(Operational Flight Program) 및 SCIL(Standard Computer Interface Library)이 탑재된 시스템 통제 모듈이다. 이는 인텔의 32비트 RISC(Reduced Instruction Set Computer)형 프로세서인 i80960KB를 탑재하고, 프로그램 저장기로 FLASH 메모리, 부트 모니터로 EPROM, 데이타 영역으로 SRAM, 시스템 버스 및 로컬버스 통제용 CPLD, 타이머, 인터럽터, RS-232 통신채널등으로 구성된다.

이들 2 장의 CPU 모듈은 동일한 하드웨어 형상을 갖고 있으나 CPU #1 모듈은 시스템 통제 및 임무 프로그램을 수행하고, CPU #2 모듈은 전방시현기 및 2개의 다기능시현기에 전시할 최상위 레벨의 전시파일(Display File)을 생성하여 GP #1 및 #2 모듈에 전달하는 역할을 한다. 이러한 분산처리는 시스템의 광역메모리(Global Memory/Comm.) 모듈을 이용한 정보공유에 의해 가능하다.

GP #1 및 #2 모듈은 CPU #2 모듈로부터 받은 전시파일을 읽고 해독하여 선분, 원(호), 이동 및 창(Window)과 같은 그래픽 원시요소(Primitive) 알고리즘을 수행하여 커시브 모듈 및 라스터 모듈에 전달한다. 커시브 및 라스터 모듈은 각각 전방시현기 및 다기능시현기를 구동하는 영상신호를 최종적으로 생성하는 모듈이다.

GP 모듈도 CPU 모듈과 유사한 하드웨어 구성을 갖고 있다. 군용급 그래픽 프로세서의 단종 및 시장주기의 급변에 대처하는 방안으로서, 범용 프로세서로 그래픽 프로세서의 기능을 모사하도록 설계하였다. 그래서 GP 모듈의 프로세서도 CPU 모듈에서와 같은 인텔의 i80960KB를 탑재하고, 전용 그래픽 프로세서로서의 기능을 수행하기 위해, 그래픽 펌웨어 및 그래픽 엔진을 고안 및 설계하였다. 2개의 GP 모듈은 하드웨어적인 형상은 같으나, 탑재 소프트웨어인 그래픽 펌웨어의 형상을 달리함으로써 라스터 및 커시브 그래픽을 전용으로 각각 처리할 수 있도록 설계하였다. 이들 모듈은 CPU 모듈의 타이머, 인터럽터, 리셋등의 논리회로등을 제외하고는 하드웨어적인 구성이 비슷하다.

커시브 그래픽 모듈 및 라스터 그래픽 모듈은 GP

모듈로부터 전달된 선분과 창에 해당하는 그래픽 원시요소를 전용 하드웨어 연산기, 즉 고안된 그래픽 엔진으로 고속 처리하여 비디오 신호처리 모듈로 해당 신호를 출력한다. 이 모듈들은 각 그래픽 채널마다 그래픽 명령코드(Graphic Operational Code)를 입력받는 FIFO, 입력된 명령 및 데이타를 해독하여 선분 및 창 그래픽 명령일 경우에는 그래픽 엔진(전가산기, 레지스터 및 State Machine으로 구성된 논리조합회로)을 동작시켜 고속의 그래픽 연산을 수행하고, 점이나 이동과 같은 명령일 때는 직접 데이타를 그래픽 버퍼(SRAM, VRAM)에 쓴다.

따라서 커시브 및 라스터 그래픽 모듈들은 데이타가 아닌 물리적인 비디오 신호를 생성하는 모듈들로서 특정의 그래픽 원시요소에 대한 가속과 비디오 신호생성을 위해 CPLD 및 FPGA로 구성된 고집적 논리소자를 이용하여 설계하였다.

비디오 신호처리 모듈(MUX)은 라스터 그래픽 모듈에서 생성된 2 채널의 비디오 신호, 1 채널의 외부 비디오 카메라 입력신호를 2 개의 MFD에 각각 출력하는 기능, 1 개의 AVTR에 3 채널의 입력신호를 선택적 녹화하는 기능, 카메라 비디오 신호와 라스터 그래픽 비디오 신호와 영상 합성하는 기능, 영상합성시 카메라 비디오 신호 대비 그래픽 비디오 신호의 인식성 향상을 위한 심볼고양(Symbology Enhancement) 기능, MFD의 그래픽 비디오 휘도 강조(Contrast) 조절기능등을 갖는다.

그 밖의 모듈로서 전원공급 모듈, 메모리 및 통신모듈, 이산신호 처리모듈들이 있다.

전원공급 모듈은 항공기 전원에서 28VDC를 공급받아 +5VDC, +/-15VDC의 전원을 생성하고 항공기의 장비의 내/외부 상방간의 전자파 간섭을 줄이기 위한 EMI 필터를 포함하고 있다.

메모리 및 통신모듈은 시스템의 공유메모리 및 RS-232, ARINC-429와 같은 직렬통신포트가 있다.

이산신호 처리모듈은 항공기의 주 시스템버스인 MIL-STD-1553B 표준 다중직렬 통신버스를 통제 및 관리하는 버스 컨트롤러와 이산신호 입/출력포트가 60 채널이 있다.

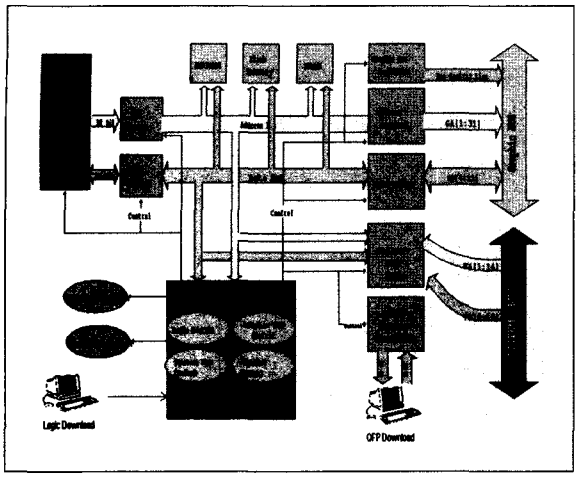
ASC의 그래픽 영상 생성기는 GP #1 모듈, GP #2 모듈, 커시브 그래픽 모듈, 라스터 그래픽 모듈, 비디오 신호처리 모듈들로 구성되며, GP #1 및 GP #2 모듈은 형상관리 및 유지/보수성을 고려하여 동일한 하드웨어로 구현되었다. 이를 위해 GP 모듈의 장착 슬롯을 그래픽 펌웨어가 감지하여 커시브 및 라스터에 해당하는 각각의 그래픽 모듈로 동작하도록 하였다.

3. GP 모듈 하드웨어 설계

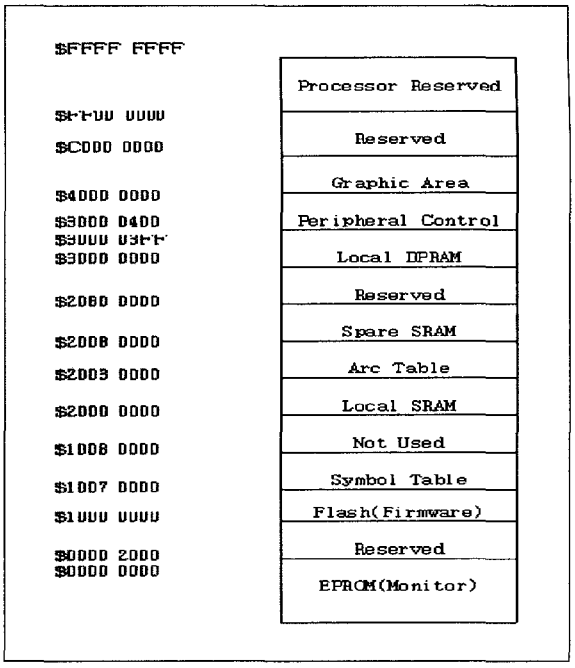
GP 모듈의 구성은 그림 2에서와 같이 i80960 프로세서, DPRAM, FLASH 메모리, SRAM, EPROM, CPLD 및 인터페이스 버퍼등으로 구성된다.

GP 모듈은 32 비트(0xFFFF FFFF)의 주소 영역을 가지며, 그림 3은 메모리 및 I/O자원들의 주소 할당영역을 나타낸다.

EPROM은 모니터 프로그램으로서 시스템 부팅, 모니터, 그래픽 펌웨어 및 심볼 테이블을 FLASH 메모리에 다운로드하는 기능을 수행한다. FLASH 메모리에 다운로드된 그래픽 펌웨어는 전시파일을 해독하여 그래픽원시요소 알고리즘을 수행하고 그래픽 영상 생성기의 통제 및 관리를 수행한다. 또한그래픽 명령코드를 생성하고, 이를 그래픽 버스를 통하여 커시브 및 라스터 그래픽 모듈의 FIFO에 전달한다. 심볼 테이블은 FLASH에 탑재된 도형(Embedded Symbology)



[그림 2] GP 보드의 내부 구성 및 데이타 흐름도



[그림 3] 메모리 맵 및 I/O 맵

이며, 원호 테이블(Arc Table)은 프로그램 실행시 원(호) 그리기에 수행되는 반복적인 실수 및 곱셈연산의 부하를 줄이기 위해, 초기화시 미리 심볼 테이블의 원호 성분을 검색하여 SRAM의 특정영역에 테이블 형태로 데이타 베이스를 구축하기 위한 영역이다. SRAM은 데이타 영역으로서 펌웨어가 구동시 변수 및 상수가 생성 및 저장되는 영역이다. DPRAM은 시스템 버스 및 로컬버스에 양방향으로 인터페이스되어 있어, CPU #2 모듈에서 전시파일을 쓰면, 로컬버스를 통하여 프로세서가 읽을 수 있다.

Peripheral Control 영역은 I/O 주소영역으로서 그래픽 시스템의 관리 및 통제를 위한 각종 레지스터 및 보조 기억장치들로서 표 1과 같다. LED Lamp Register는 GP 모듈의 시험 및 점검을 위한 LED를 동작시킬 때 사용된다. System Fail Register는 GP 모듈에 전원이 인가된 후 각 모듈이시스템 버스를 엑세스 가능함을 알리기 위한 신호로서, 이 신호는 Setup Control Register가 활성화되어야만 쓰기가 가능하다. Enable Program Register는 Flash 메모리에 프로그램을 쓸 때 사용하기 위한 것으로, 이 신호를 이용하여 Flash 메모리의 'WE' 신호를 만든다. CPU Configuration Register는 GP 모듈의 셋업을 위하여 사용하는 것으로 Setup Control Register가 활성화되어야만 이 신호를 쓸 수 있다. Setup Control Register는 CPLD내의 다른 레지스터등을 셋업(setup)하기 위한 신호로, 이 레지스터가 활성화되어야만 System Fail register와 CPU Configuration Register의 쓰기가 가능하다.

External Input Status Register는 외부의 스위치 입력 상태를 읽어 들이는 레지스터로서 다운로드및 시험용으로 사용된다. BD ID Read Register는각 모듈의 슬롯번호를 인식하기 위해 사용된다.

Cursive Draw Start는 Frame Synch Interrupt 발생시, GP 모듈은 Display Bank를 바꾸고 해당 주소에 한번의 쓰기 동작을 수행하여야 그리기가 시작된다.

[표 1] Peripheral Control Map

주 소	레지스터
0x3300 0000	LED Lamp Register Address
0x3300 0010	System fail Signal Clear Address
0x3300 0020	Enable Program Register Address
0x3300 0030	CPU Configuration Register Address
0x3300 0040	Setup Control Register Address
0x3300 0050	External Input Status Reg. Address
0x3300 0060	Board ID Read Register Address
0x6400 0000	Cursive Draw Start
0x6400 0010	Cursive Graphic Engine Draw Start
0x6400 0020	Cursive Command FIFO Reset
0x6400 0030	Cursive Command FIFO Address
0x6400 0040	Raster1 Graphic Engine Draw Start
0x6400 0050	Raster1 Command FIFO Reset
0x6400 0060	Raster1 Command FIFO Address
0x6400 0070	Raster2 Graphic Engine Draw Start
0x6400 0080	Raster2 Command FIFO Reset
0x6400 0090	Raster2 Command FIFO Address

이때 쓰기 시의 데이터는 임의의 값이다. Graphic Engine Draw Start는 GP 모듈이 그래픽 모듈의 FIFO에 새로운 한 프레임의 디스플레이 파일을 모두 쓰기 전, 해당 주소에 한번의 쓰기를 수행하여야 Graphic Engine은 FIFO에서 해당 명령어를 읽기 시작한다. 이때 쓰기 시의 데이터는 임의의 값이다. Command FIFO Reset은 GP 모듈이 DPRAM(Dual Port RAM) Interrupt 발생시, Symbol Update Flag을 검사하여 Symbol Update Flag이 리셋되어 있는 경우, Command FIFO Reset 명령을 발생하여 FIFO를 리셋시킨 후, 새로운 한 프레임의 디스플레이 파일을 FIFO에 쓰기 시작한다. 이때 쓰기 시의 데이터는 임의의 값이다. 표 2에서와 같이 라스터 그래픽 영역은 8 비트 폭과 2 개의 뱅크(영상출력 및 데이터 입력을 별도의 영역으로 관리하는 더블 버퍼링용)를 가지는 VRAM 영역으로 구성되고, 커시브 그래픽 영역은 SRAM 영역으로서 32 비트 폭과 2 개의 뱅크로 구성된다. 그래픽 제어영역은 라스터 및 커시브 신호 및 데이타를 제어하기 위한 영역으로서 그래픽 보드의 각종 레지스터들에 할당된다. GP 모듈은 그래픽 모듈의 자기진단(BIT, Built-In Test)을 위하여 FIFO를 거치지 않고, 표 3의 주소로 직접 라스터

[표 2] 라스터 및 커시브 그래픽 영역

0x4000 0000 - 0x4FFF FFFF	Raster1 VRAM(DRAM) Area
0x5000 0000 - 0x5FFF FFFF	Raster1 VRAM(DRAM to SAM, SAM to DRAM) Transfer Cycle
0x6000 0000 - 0x7FFF FFFF	Reserved
0x8000 0000 - 0x8FFF FFFF	Raster2 VRAM(DRAM) Area/Cursive SRAM Area
0x9000 0000 - 0x9FFF FFFF	Raster2 VRAM(DRAM to SAM, SAM to DRAM) Transfer Cycle/Cursive SRAM Area
0xA000 0000 - 0xBFFF FFFF	Reserved(Raster)/Cursive SRAM Area

및 커시브 프레임 버퍼 즉, VRAM 및 SRAM에 엑세스하여 읽기를 수행할 수 있다.

라스터 화면은 512 x 512 픽셀로 구성되는 정방형이며, 각 픽셀은 8 비트의 색 코드를 가진다. 좌표계의 X, Y축은 표 3과 같이 각각 9 개의 데이터 비트들로 구성된다.

커시브 화면은 4096 x 4096 픽셀로 구성되며, 픽셀 폭은 4 비트이다. 커시브는 CRT의 전자빔 편향을 위한 수평 및 수직 주사전압을 직접 제어함으로써 영상을 생성하는 방식이다. 벡터방식으로 생성된 심볼의 각 점들은 화면의 좌표 값들로서 SRAM에 순차적으로 저장되며, 모든 심볼이 저장되었을 때, 표 4와 같이 픽셀코드, X 및 Y좌표의 데이터들을 특정의 속도로 읽어 내어 12 비트의 X 및 Y DAC(Digital to Analogue Converter)를 거쳐 -10 ~ +10Volts의 전압값으로 전방시현기에 출력된다. 이때 SRAM을 읽는 속도는 커시브 도형의 밝기와 관계가 있다. 느리게 읽을수록 밝지만 실시간 구현에 제한이 따른다. 그러나 라시터 주사방식과 달리, 영상이 아닌 도형을 처리하는데는 적은 시간으로 고휘도의 영상을 생성할 수 있다.

[표 3] 라스터 채널의 화면좌표

A31	A30	A29... A25	A24... A16	A15... A11	A10... A2	A1.. A0
0	1	0	X 좌표	0	Y 좌표	0

[표 4] 커시브 버퍼(SRAM)의 데이타

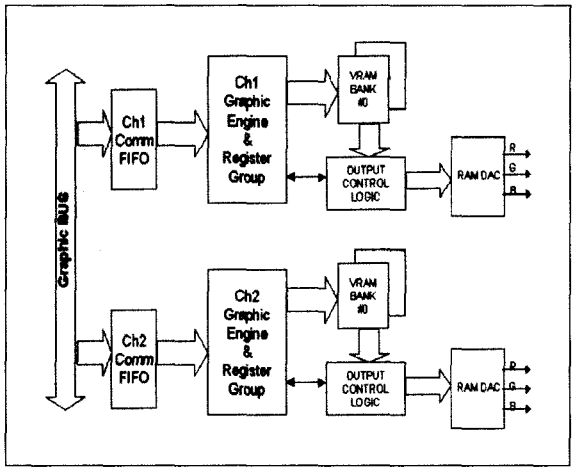
D31.. D28	D27.. D24	D23.. D12	D11.. D0
xxxx	Pixel Code	X Coordinate	Y Coordinate

4. 그래픽 모듈 하드웨어 설계

4.1 라스터(Raster) 그래픽 모듈 설계

라스터 그래픽 모듈은 그림 4와 같이 2 채널을 가지며, 각각 FIFO, 그래픽 엔진/레지스터, 제어로직, 비디오 버퍼 및 RAMDAC으로 구성된다. 라스터 방식은 순차주사 방식으로서 RS-170A 표준으로 설계되었다. 세부 설계 요구규격은 480 * 480 픽셀 해상도, 128 색상 및 비월 주사방식등으로 정하였다.

라스터 GP 모듈로부터 그래픽 명령코드를 50Hz 주기로 FIFO를 통해서 입력받아, FPGA내부에 구성된 그래픽 엔진이 이를 해독하여 선분, 이동, 창,점 및 색상등의 그래픽 명령을 처리하여 그리기(화면좌표상의 픽셀위치, 속성)를 수행한다. 그리기는 VRAM으로 구성된 프레임 버퍼에 해당 행/열 주소에 속성 데이타를 쓰는 것이다. 물리적인 영상신호(RS-170A)는 30Hz 주기로 VRAM을 행 단위로 순차적으로 읽어 쉬프트 레지스터와 RAMDAC에 전달되면 R(Red), G(Green, Sync), B(Blue)의 출력신호가 생성된다.

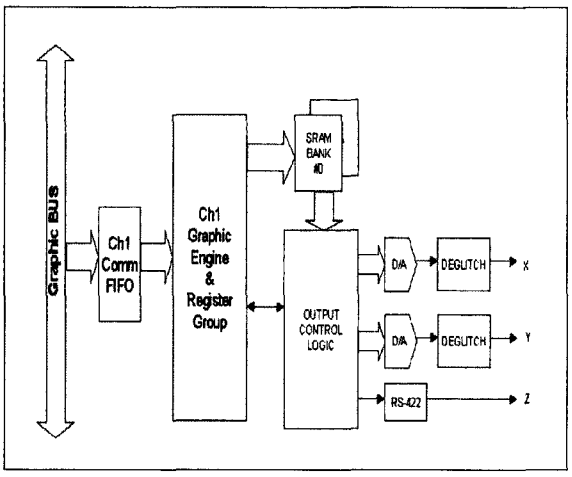


[그림 4] 라스터 그래픽 모듈의 구성도

4.2 커시브(Cursive) 그래픽 모듈 설계

커시브 그래픽 모듈은 그림 5와 같이 한 채널의 FIFO, 그래픽 엔진/레지스터, 제어로직, 비디오 버퍼 및 X/Y DAC(Digital to Analog Converter)등으로 구성된다.

커시브 방식은 일반적인 라스터 방식과 달리 주사를 위한 수평, 수직편향 전압 및 주사속도를 직접 제어하여 심볼을 개체단위로 그리는 방식이다. 이와 유사한 방식은 X-Y 플로터, 아날로그 오실로스코프등이 있다. 커시브 방식은 해상도, 갱신률 및 진폭과 같은 표준 규격이 없지만, 본 연구에서는 해상도를 4096 * 4096 픽셀, 영상 갱신률을 16Hz, 편향전압을 +/-10Volts 로 정의하였다. 커시브 GP 모듈로부터 그래픽 명령코드를 50Hz 주기로 FIFO를 통해서 입력받아, FPGA내부에 구성된 그래픽 엔진이 이를 해독하여 선분, 이동, 창 및 점등의 그래픽 명령을 처리하여 그리기(화면좌표상의 픽셀위치 및 속성)를 수행한다. 커시브 영상신호는 그래픽 엔진이 SRAM으로 구성된 프레임 버퍼에 순차적으로 점 좌표 데이타를 기록하면, 출력제어기는 순차적으로 점 좌표 데이터를



[그림 5] 커시브 그래픽 모듈의 구성도

50Hz로 읽어 X/Y DAC에 전달하여 수평/수직 편향 전압과 Z 신호를 발생함으로써 생성된다. 빔의 "On/Off"를 제어하는 Z신호는 RS-422형태의 디스크리트 신호로 발생된다. 전방시현기는 편향 및 전자빔의 제어에 따르는 고유한 동적 주파수 영역이 존재한다. 즉 물리적인 응답지연 및 과도응답를 고려한 적절한 신호를 생성하여야 한다. 예를 들어, 한 심볼 개체를 그린 후 빔을 꺼고 상대적으로 먼 좌표로 빔을 이동하여 그리기를 곧바로 수행하면, 편향전압 및 빔이 안정되지 않은 상태에서 빔이 켜져 의도치 않은 그리기가 수행된다. 그래서 (1) Bright Up Delay Start/End, (2) Pre-slew Delay, (3) Beam Position Delay와 같은 타이밍 제어가 필요하다. 커시브 그래픽 프레임 버퍼에 저장된 좌표 데이타와 빔 "On/Off" 데이타는 동시에 읽혀지지만, 이들은 각기 다른 경로를 통해서 출력된다. 즉 좌표 데이터는 DAC를 경유하지만 빔 데이터는 곧바로 출력된다. 따라서 이들을 동기화하기 위해 (1)과 같은 빔의 시간지연 제어가 필요하다. (2)의 용도는 빔의 "On/Off"시 응답지연에 의한 화면 잡음이 발생되며, 이를 방지하기 위해 빔이 안정될 때까지 기다렸다가 이동을 하는 지연시간이 필요하다. (3)은 편향전압의 응답특성 즉, 빔이 이동하여 안정될 때까지 필요한 지연시간이다.

5. 그래픽 소프트웨어 설계

그래픽 소프트웨어는 라스터 및 커시브 GP 모듈의 FLASH 메모리에 탑재되어, 전시파일 해독, 그래픽 원시요소 알고리즘 수행 및 그래픽 주변장치 제어를 수행하는 하드웨어 기반 그래픽 커널 소프트웨어이다. 프로그램은 (1) 주 루틴, (2) 전시파일 갱신 인터럽터

[표 5] 전시파일(Display File) 구조

1	PRCB	- board ID, channel enable - BIT enable, BIT report - CCB, R1CB, R2CB pointer - check-in report
2	R1CB/CCB	- display file address - harmonization - cursive writing speed - bright up delay - beam position delay - pre-slew delay - x/y offset - video enhancement - global scale - R1 Palette
3	R2CB	- display file address - harmonization - x/y offset - video enhancement - global scale - R2 Palette
4	R1/C FILE	display file (raster channel 1 or cursive channel)
5	R2 FILE	display file (raster channel 2)

서비스 루틴, (3) 커시브/라스터 출력 인터럽터 서비스 루틴등으로 구조화되었다. (1)은 그래픽 시스템 초기화, 데이터 구조 초기화, 자체진단(PBIT), 인터럽터 셋업 및 무한 반복 자체진단(CBIT) 루틴으로 구성된다. (2)는 해당 인터럽터 발생시 전시파일을 읽어 원시요소 알고리즘을 수행하여 라스터/커시브 그래픽 모듈의 FIFO에 그래픽 명령코드를 전달하고, 그래픽 제어용 각종 레지스터를 통제한다. (3)은 50Hz 커시브 리프레쉬(Refresh) 타이머, 30Hz 라스터 리프레쉬 타

[표 6] 그래픽 원시요소 명령어

ID	Command	ID	Command
1	move	13	jump_call
2	draw_line	14	subroutine_return
3	draw_invisible_line	15	infill_rectangle
4	draw_symbol	16	infill_triangle
5	define_window	17	infill_arc
6	define_translate	18	set_anti_alias
7	define_roll	19	reset_Y_depression
8	define_localscale	20	reset_window
9	define_flash_rate	21	reset_effect
10	define_fore_color	22	null_command
11	define_back_color	23	draw_arc
12	define_line_type		

이머에 의한 인터럽터 서비스 루틴으로서 우선순위가 가장 높다. 이는 더블 버퍼링을 위한 프레임 버퍼 관리를 수행한다. 표 5는 CPU #2 모듈이 GP #1, #2 모듈에 전달하는 그래픽 전시파일의 데이터 구조체를 나타낸다. 프로세서 통제블럭(PRCB), 커시브 통제블럭(CCB), 라스터 통제블럭(RCB) 및 라스터/커시브 전시파일로 구성되며, 이들은 DPRAM의 시작주소로부터 순차적으로 할당된다. 그러나 각각의 구조체들의 크기가 가변적이므로 포인터를 참조하도록 하였다. 가능한 시스템 버스의 부하를 줄이기 위해 전시파일은 정적 데이터 및 동적 데이터영역으로 구분되어 동적 데이터만 갱신된다. 하지만 전시모드의 변경과 같이 새로운 페이지를 그릴 때는 대부분의 데이터가 갱신된다. 또한 GP 모듈이 오프라인에서 다운로드된 심볼 데이터 구조체를 FLASH 메모리에 탑재하고 있어, 응용 소프트웨어에서 직접 그리기를 수행하지 않고, 위치, 크기, 회전, 속성등의 요소와 함께 심볼 테이블 참조 포인터를 전달함으로써 데이터 통신 부하 및 처

리속도에 향상을 도모하였다. 표 6은 전시파일의 그래픽 원시요소 명령어들을 나타낸다. 그래픽 원시요소중에서 반복 연산(덧셈, 비교)이 가장 많은 선분, 창, 이동 알고리즘은 소프트웨어뿐만 아니라 고안된 하드웨어 그래픽 엔진을 통해 분산 처리함으로써 처리속 도를 높일 수 있었다. 그러나 원(호)는 64 비트 실수 곱셈을 반복적으로 수행함으로 인해 실행속도의 저하의 큰 요인으로 작용하여, 시스템 초기화시 심볼 테이블의 모든 원(호) 성분을 검색하여 미리 연산을 수행하고, 이를 데이터 베이스화하여 원호 테이블에 저장한다. 실행시 원(호) 명령을 수신하면 원호 테이블의 목록을 참조하여 포인터를 얻어 원(호)를 재생한다. 이는 단지 메모리 읽기에 해당하는 동작을 반복적으로 수행하는 것이어서 고속의 그리기가 구현된다. 모든 그래픽 원시요소의 시작점, 끝점 및 이동은 최적경로로 수행되어야 커시브방식의 그리기를 구현할 수 있다. 그렇지 않으면 전자빔 및 편향제어의 효율 및 영상의 질이 저하된다.

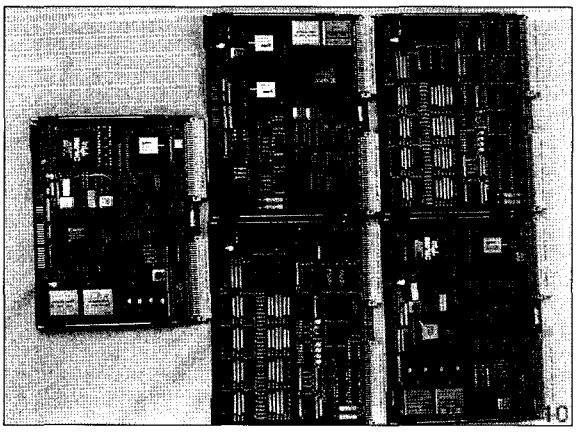
6. 제작 및 시험

하드웨어는 그림 1에서와 같이 동일 형상인 라스터 GP 모듈 및 커시브 GP 모듈, 라스터 및 커시브 그래픽 모듈, 비디오 신호처리 모듈으로 구성되는 5개의 모듈이 제작되었다. 이들은 프로그램 가능한 고집적 논리소자인 Xilinx사의 FPGA 4013, Lattice사의 CPLD 1048 및 GAL22V10등을 이용하여 소프트웨어적으로 제작됨으로써 하드웨어의 융통성을 한층 높였다. 그림 6은 제작된 5개 모듈의 외형을 나타낸다.

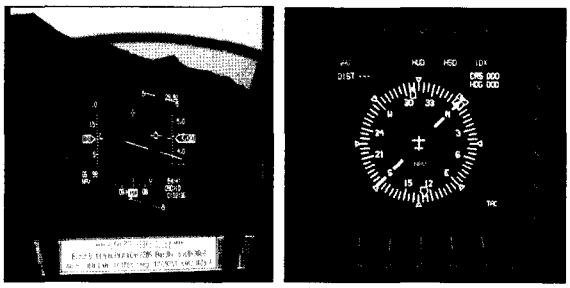
소프트웨어는 C언어로 작성되어 i80960KB용 크로스 컴파일러를 이용하여 실행파일로 만들어 FLASH 메모리에 탑재되었다. 또한 심볼 테이블도 자체 제작한 전용 그리기 도구를 이용하여 오프라인에서 작업하여 FLASH 메모리의 하위영역에 저장되었다.

하드웨어 기능시험은 시뮬레이션, 자체진단(BIT) 프로그램등으로 수행하였고, 소프트웨어 기능시험은 유닛시험을 수행한 후 완성된 하드웨어상에서 계측기 및 모니터를 이용하여 수행하였다. 그래픽 영상생성기의 기능시험을 마친 후 ASC 랙에서 통합시험을 응용소프웨어 및 항공전자시스템 개발장비(ASDR, Avionic System Development Rig)와 같이시험하였다. 그림 7은 ASDR 의 전방시현기 및 다기능시현기에 시현된 영상화면을 나타낸다.

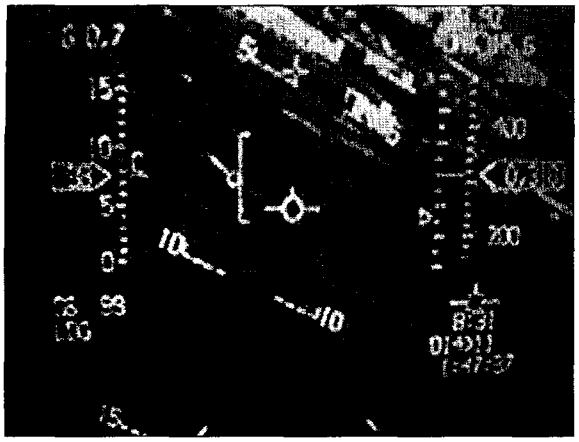
통합시험 환경에서 소프트웨어에 재생 카운터를 임



[그림 6] 그래픽 영상생성기 하드웨어



[그림 7] 전방시현기 및 다기능시현기 화면



[그림 8] 비행시험 장면

시로 추가하여 프레임 갱신률을 확인한 결과, 커시브 및 2 채널의 라스터 그래픽이 16Hz 이상의 갱신률로 화면을 재생함을 확인하였다. 그래픽 영상생성기는 다른 모듈의 하드웨어와 같이 환경시험 및 EMI 시험을 통과하였고, 또한 항공기에 장착되어 지상시험과 비행시험을 통해 그 성능을 입증하였다. 그림 8은 비행시험시 전방시현기에 장착된 카메라로 촬영한 사진이다.

7. 결 론

현대전은 물론 미래전은 공중전의 우위확보가 성패의 관건이라 할 수 있다. 선진 각국에서는 일찍이 항공전자분야에 관심을 갖고 각종 컴퓨터 및 첨단 센서 장비를 포함하는 고성능 항공기를 개발하였다. 특히 정보획득의 주요 수단인 시현계통의 기술발전이 돋보인다.

본 연구에서는 시현계통의 주 시현장비인 전방시현기 및 다기능시현기를 구동하여 다양한 정보를 실시간으로 제공하는 항공기용 그래픽 영상생성기를 국내 독자기술로 설계, 구현하였다. 또한 군용으로 사용 적합성을 입증하기 위해 각종 기능 및 성능시험을 성공적으로 마쳤다.

참 고 문 헌

[1] Mark Hewish, "Integrated avionics the heart of future combat aircraft", Defense electronics & computing(supplement to IDR 9/1992).

[2] Irv Englander, "The architecture of computer hardware and systems software", 2000, John Wiley & Sons, Inc.

[3] Sheldon B. Herskovitz, "Set it and forget it, The world of embedded systems", 1991, Journal of electronic defense.

[4] Edward Angel, "Computer Graphics", 6/1990, Addison-Wesley Publishing Company, Inc.