



자동차 클러스터의 감시 및 제어를 위한 모델기반설계 기법 연구

Study on a Model-based Design Technique for Monitoring and Control of a Vehicle Cluster

김동현[†]
Dong Hun Kim[†]

경남대학교 전기공학과
Department of Electrical Engineering, Kyungnam University

요약

본 연구는 모델기반설계 기법을 이용하여 자동차 클러스터의 감시 및 제어를 하는 스테이션을 설계한다. 설계 도구로 매트랩 GUI(Graphic User Interface), M 프로그램, 시뮬링크(simulink), 스테이트 플로우(state flow), 툴박스(tool box)를 사용하여 실제 자동차 클러스터 시스템과 연동하여 자동차에서 들어오는 경고, 인터럽트 등의 각종 정보 등을 감시한다. 감시 수단으로는 PC(Personal Computer) 스테이션을 사용하여 자동차 클러스터 설계 시 툴 박스의 인터페이스 명령함수가 실제 자동차 클러스터 시스템과 연동하게 한다. 따라서, 기존의 텍스트 방식과 달리 모델기반설계로 개발된 자동차 클러스터 시스템은 각 기능 및 알고리즘을 블록과 상태플로우로 프로그램에 따라 작성하기 때문에 알고리즘의 수정이나 기능 추가가 용이하며, 또한, PC를 통해 모니터 상에서 동작 알고리즘을 검증하기 때문에 클러스터의 개발과 수정에 따른 많은 시간과 비용을 절감할 수 있는 효과를 준다.

키워드 : 모델기반설계, 클러스터, 그래픽 사용자 인터페이스, 감시 제어

Abstract

This paper presents the development of a monitoring and control system for a vehicle cluster using a model-based design technique. For MBD(model-based design), MATLAB GUI(Graphic User Interface), M programs, simulink, state flow, and tool boxes are used to monitor a number of data such as warning, interrupts, and etc. connected to a real vehicle cluster. As a monitoring tool, a PC(Personal Computer) station interworks with the real vehicle cluster through the interface commands of tool boxes. Thus, unlike existing text-based designs, the MBD based vehicle cluster system provides very easy algorithm updates and addition, since it offers a number of blocks and state flow programs for each functional actions. Furthermore, the proposed MBD technique reduces the required time and cost for the development and modification of a vehicle cluster, because of verification and validation of the cluster algorithm on the monitor through a PC.

Key Words : Model-based Design, Cluster, Graphic User Interface, Supervisory Control

Received: Nov. 4, 2016
Revised: Nov. 16, 2016
Accepted: Nov. 16, 2016
[†]Corresponding authors
dhkim@kyungnam.ac.kr

1. 서론

최근 자동차, 항공기, 선박과 같이 감시 제어용 클러스터를 포함한 이동수단에서와 같이 복잡한 알고리즘 코드를 다루는 유사 기술 분야에서는 모델기반설계 MBD가 새로운 소프트웨어(SW) 개발 방법으로 각광을 받고 있다. MBD는 '그림을 그리면' 코드가 자동으로 만들어지는 방식이며, COBOL, C++와 같은 컴퓨터 언어로 프로그램을 작성하는 전통적인 방법과 차별화되고 있다. 그 이유는 소프트웨어(SW) 기능이 고도화 되고 복잡해지면서 기존의 텍스트 기반의 프로그램 설계 방식인 TBD(Text-Based Design) 코드 작성을 위해 100만~1000만 라인이 필요해 사람이 일일이 대응하기가 힘들고 작업 중에 실수가능성이 높은 문제를 해결하기가 매우 어려워지고 있기 때문이다.

이러한 문제점에 대응하기 위해 MBD 방식의 SW 방법은 텍스트 기반과는 달리 사용자 편의 중심으로 프로그램의 수정과 문제 해결에 매우 용이하므로 유용하게 사용되고 있다[1][3]. 현재 임베디드 SW를 중심으로 하드웨어(HW)에 SW를 적용하는 방식이 빠르게 늘면서 솔루션(solution) 부재와 전문가 부족이 해결과제로 부상하고 있다[4][6]. 특히, 자동차의 기술 개발에 있어서 자동차

이 연구결과물은 2016학년도 경남대학교 학술진흥연구비 지원에 의한 것임.
This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

각종 장치 및 부품의 고급화에 대한 소비자의 요구 사항이 계속 증가하고 있다[7]. 즉, 자동차 자체 내에서 내비게이션, 경고등, MP3, DMB 등의 기능이 제공되지만 이러한 프로그램이 얼마나 소비자에게 사용하기 쉽게 제공하느냐가 매우 중요한 요인으로 작용하고 있다.

MBD는 초기에 모델 기반 디자인이라는 개념이 제창되어 제품 설계에 도움을 주고 있었지만, 최근에는 한층 진일보하여 모델 기반 개발이라는 사고방식으로 제조업 영역으로 확장되고 있다[10]-[12]. MBD는 현재 BMW, 현대모비스 등 자동차 업계와 국방, 항공 분야에서 임베디드 시스템의 설계에 사용되고 있으며, 사용 범위가 무궁무진하여 여러 분야에서 사용이 늘어나고 있는 추세이다[13]-[15].

MBD 방식을 사용한 클러스터의 시스템 개발은 국외에서도 현재 연구 중에 있으며, 국외 유명 독일 자동차 연구소에서는 플랫폼이 이미 만들어져서 곧 MBD 방식으로 클러스터를 설계할 예정으로 알려져 있다. 한편, 국내에서는 현대모비스가 클러스터가 아닌 ECU(Engine Control Unit)와 관련해서 SW 개발에 MBD를 도입하였다[16].

따라서 본 연구의 목적은 위와 같은 문제점을 해결하기 위해 제시하는 것으로, 보다 상세하게는, 기존 텍스트 기반의 설계 방식과 달리 매트랩(MATLAB)의 GUI(Graphic User Interface)와, M 프로그램, 시물링크(simulink), 스테이트 플로우(state flow), 툴박스(tool box)를 이용하여 실제 자동차 클러스터 시스템과 연동하는 PC 스테이션 환경에서 모델기반설계 방식으로 변환한다. 결과로서, 클러스터 설계에 따른 사용자의 편의와 프로그램의 수정 및 문제해결이 용이하고, 사용자 중심의 GUI 인터페이스를 통해 모델기반설계 기법이 가능한 자동차 클러스터의 스테이션 설계 방법을 제안한다.

2. 문제 설정

본 연구에서, 모델기반설계 기법을 이용한 자동차 클러스터의 설계 방법은, 매트랩 GUI(Matlab Graphic User Interface)와, M 프로그램(M Program), 시물링크(Simulink), 스테이트플로우(Stateflow), 툴박스(Tool Box) 및 PC 스테이션 환경 수단을 필요로 한다[17][18]. 자동차 클러스터 설계 시, 툴박스의 인터페이스 명령함수가 실제 자동차 클러스터 시스템과 연동하는 PC 스테이션 환경에서 모델기반설계 기법으로 실제 설계가 이루어지도록 하는 것을 특징으로 한다.

그림 1은 모델기반설계 기법을 이용한 자동차 클러스터의 스테이션 설계 방법에 대한 기술적 구성을 개략적으로 나타낸

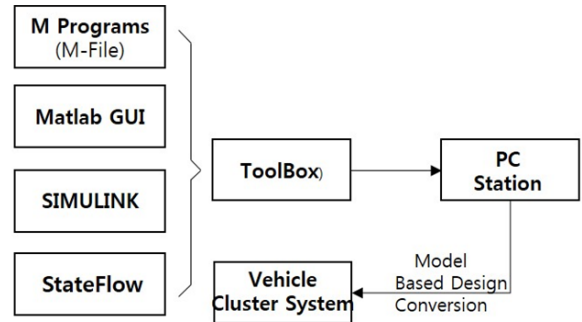


그림 1 매트랩을 이용한 MBD
Fig. 1. MBD using MATLAB

블록도를 보여준다. 그림 1에서, 매트랩 GUI는, 매트랩의 GUI 가이드(GUI Guide)를 이용하여 버튼 등과 같은 그래픽적인 화면을 구성하고 매트랩의 M 프로그램을 작성하여 GUI의 상세 동작 알고리즘을 작성하며 시물링크 파일과 연동하여 시물링크를 제어하는 역할을 한다.

여기서, 매트랩은 과학적 또는 공학적인 문제를 컴퓨터를 이용하여 효율적으로 해결할 수 있도록 프로그래밍과 수치해석을 위한 다양한 기능을 제공하는 프로그래밍 언어이다. 매트랩은 해결한 문제의 결과를 시각적으로 표현해주는 강력한 그래픽화면제공 기능을 가지고 있어 문제를 직관적으로 쉽게 이해할 수 있도록 해준다. 매트랩에서는 M 파일(M-file)을 이용하여 프로그래밍이 가능하고, 여러 가지 명령어를 순차적으로 실행할 수 있다.

그림 1에서, M 파일은, 매트랩 상에서 자동차 클러스터 설계 시 상세 동작을 작성할 때 사용되는 프로그램이다. 시물링크는, 자동차 클러스터의 동작 알고리즘을 모델기반설계 방식으로 설계하기 위한 도구로 사용된다. 스테이트플로우는, 시물링크 내의 블록 중 하나이며, 자동차 클러스터의 동작을 각각의 우선순위에 따른 알고리즘을 그래픽(Graphic)으로 작성하는 블록으로 사용된다. 툴박스는, PC 스테이션 환경을 갖는 자동차 클러스터 설계 시 모델기반설계 방식으로 변환시키는 함수의 수단으로, 자동차 클러스터 설계에 따른 인터페이스를 함수화 하여 모아 놓은 매트랩의 함수이다. 본 연구에서는 툴 박스의 명령 함수를 이용하여 실제 자동차 클러스터 시스템과 연동하는 PC 스테이션 환경에서 모델기반설계가 이루어지도록 변환한다.

그림 1에서, PC 스테이션 환경은, 윈도우즈(Windows) 또는 리눅스(Linux) 운영체제 환경에서 매트랩 소프트웨어가 설치되어 있는 환경이며, 툴박스의 명령 함수를 통해 자동차 클러스터 시스템과 연동하여 모델기반설계(MDB) 방식으로 자동차 클러스터를 설계 가능하도록 하는 특수 목적의 컴퓨터 수단이다.

모델기반설계를 위하여, PC에서 매트랩 GUI의 가이드(Guide)를 이용하여 그래픽 화면을 구현하고, 매트랩 GUI의 M 프로그램을

통해 매트랩 GUI의 상세 동작 알고리즘을 작성한다.

먼저, 본 연구에서는 모델기반 설계를 이용한 자동차 클러스터 스테이션 개발을 위해 기본 스위치 입력을 통한 시뮬링크 프로그램을 작성한다. 즉, PC에서 매트랩 GUI의 가이드를 이용하여 그래픽 화면을 구현할 때, M 프로그램을 통해 매트랩 GUI의 상세 동작 알고리즘을 작성하고 기본 스위치 입력으로 자동차 클러스터의 모델기반설계를 위한 소프트웨어 도구(tool)인 시뮬링크 프로그램을 작성하는 제1 단계를 갖는다. 설계 2단계에서는 시뮬링크의 프로그램과 매트랩 GUI 프로그램과 스테이트플로우(Stateflow)와 연동하여 시뮬링크를 제어한다. 그리고 나서, 매트랩 GUI의 프로그램을 작성하여 스크린의 메뉴 선택과 현재의 클러스터 상태를 표시하는 차트 시뮬링크 프로그램을 구성시킨다. 설계 3단계에서는 기본적인 메뉴 선택과 CAN통신 혹은 RS232 통신 인터페이스로 수신된 자동차 상태 알림 스테이트플로우 차트를 통해 실제 자동차 클러스터의 동작 감시를 위한 PC 스테이션 환경을 제작한다. 설계 4단계에서는 모델기반설계 방식으로 설계가 가능하도록 자동차 클러스터의 모든 설계 정보를 툴박스의 명령함수를 통해 PC 스테이션 환경으로 정보를 이식한다.

그림 2는 모델기반설계 기법을 이용한 자동차 클러스터의 스테이션 설계에서 키보드 입력에 따른 클러스터의 기본 동작 확인을 나타낸다.

본 연구에서는, 자동차의 클러스터를 제작하기 전에 PC에서 알고리즘의 동작 확인을 위하여 매트랩의 시뮬링크에서 키보드 입력

라이브러리를 추가하여 키보드의 특정키를 자동차의 클러스터에 장착된 스위치 입력으로 시뮬링크를 설계하고, 제작할 자동차의 클러스터 스위치 입력에 따른 동작 알고리즘을 시뮬링크의 스테이트플로우 차트를 이용하여 설계한다. PC 스테이션 환경에서 작성된 매트랩 시뮬링크를 실행하여 키보드 입력을 받아 클러스터 스크린의 메뉴 선택을 통해 제작할 자동차 클러스터의 스위치 입력 동작 알고리즘을 검증하는 것을 특징으로 한다.

3. 단계별 구체적 MBD 설계 기법

이번 장에서는 2장에서 그림 1과 그림 2를 기반으로 자동차 클러스터의 MBD 설계에 대한 구체적인 설계 내용을 각각의 과정으로 나누어 다룬다.

본 연구에서, 매트랩 GUI의 프로그램 작성은 제작할 자동차 클러스터에 입력되는 입력변수 버튼을 생성한 후, 매트랩 시뮬링크의 입력변수 버튼 상태에 따른 입력변수 값을 표시하는 상수블록을 생성시키는 단계가 필요하다(3.1절). 그리고 나서, 자동차 클러스터와 매트랩 GUI의 간의 통신연결을 통하여 자동차 클러스터에 결과를 매트랩 GUI가 수신하여 표시하는 자동차 클러스터의 상태를 표시하는 이미지뷰(Image View)와, GUI의 입력변수에 의한 동작 알고리즘에 의한 이미지뷰를 생성시키는 단계가 필요하다(3.2절). 생성된 매트랩 GUI의 입력변수 버튼과 시뮬링크의 입력변수 상수블록, 매트랩 GUI의 동작 알고리즘 결과 이미지뷰와 시뮬링크의 동작 알고리즘 결과 출력변수는 서로 M 프로그램을 통하여 연동시킨다. 시뮬링크에서 자동차의 모든 입력변수의 상태에 따른 동작 알고리즘은 입력변수의 중요도에 따라 각 그룹으로 나누어지며, 각 그룹은 순번에 따라 상위와 하위로 나누어진다. 각 그룹은 상위, 하위 인터럽트와 같은 각 상황에 따른 인터럽트 수용 여부가 정해지며, 정해진 자동차 클러스터의 동작 알고리즘에 따라 시뮬링크의 스테이트플로우 차트를 설계하고 작성하는 단계와, 스테이트플로우 차트를 이용하여 작성된 시뮬링크 프로그램과 매트랩 GUI는 시뮬레이션을 수행하여 자동차 클러스터의 동작 알고리즘을 검증하는 단계가 필요하다(3.3절). 마지막으로 모델기반설계 방식으로 설계가 가능하도록 자동차 클러스터의 모든 설계 정보를 툴박스의 명령함수를 통해 PC 스테이션 환경으로 정보를 이식하는 제4 단계가 포함된다(3.4절).

3.1 기본 스위치 입력을 통한 시뮬링크 프로그램 작성

그림 3은 모델기반설계 기법을 이용한 자동차 클러스터의 스테이션 설계 방법에 대한 화면표시를 위해 매트랩 시뮬링크 구성의 한 예를 보여준다. 그림4는 자동차 클러스터의 감시, 제어를

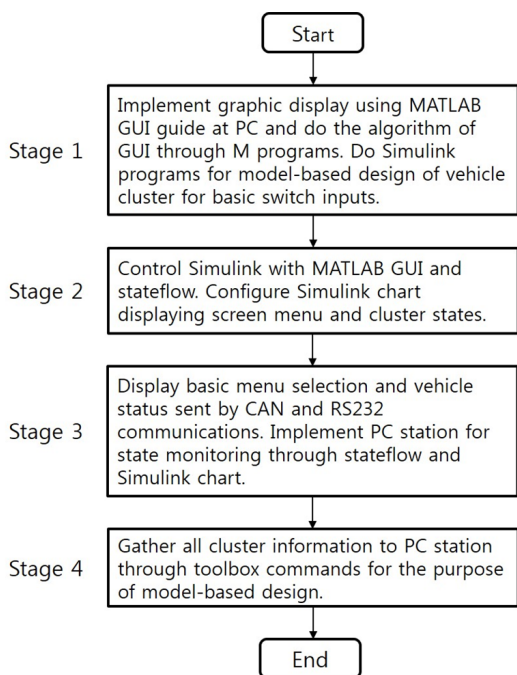


그림 2. 자동차 클러스터의 PC 스테이션 설계
Fig. 2. PC Station Design of a Vehicle Cluster

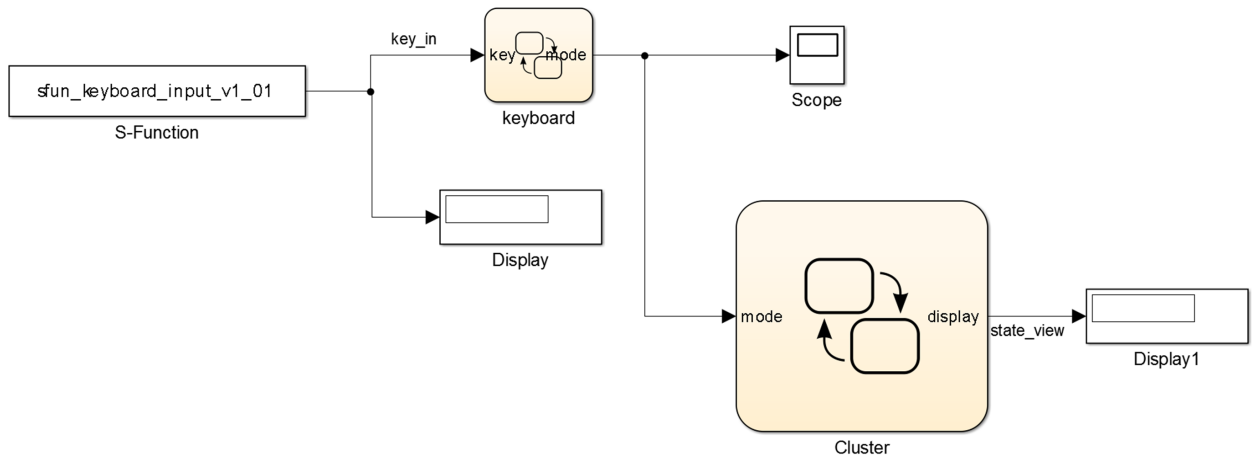


그림 3. 매트랩 시뮬링크 구성의 한 예
 Fig. 3. Example of Configuration by MATLAB Simulink

위한 전체 시스템 시뮬링크 화면을 보여준다. 각각의 그룹은 주행가능거리, Trip A, Trip B, 누적시간, 디지털 속도계, AV 모드, 서비스/인포모드의 각 현재 상태(state)와 메뉴 스위치, 상스위치, 하스위치, OK 스위치와 같은 사건(event)에 따라 구별된다. 각 그룹에 대한 자세한 내용은 본 논문이 산학과제를 통해 나온 결과이어서 회사의 보안 문제로 생략한다.

그림 3과 그림 4에서, 자동차의 클러스터를 제작하기 전에

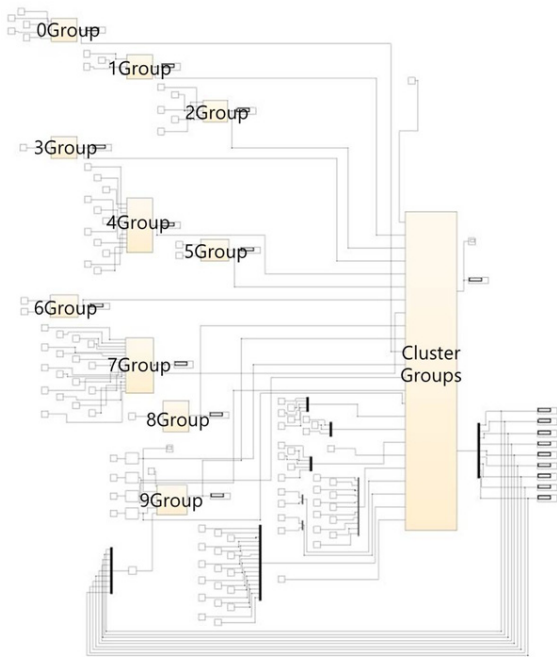


그림 4. 전체 시스템(회사 보안상 그림 내 하부그룹 내용은 표시하지 않음)

Fig. 4. Overall System Configuration

PC에서 알고리즘의 동작 상태를 확인하고, PC에서 매트랩의 시뮬링크를 통하여 개발될 자동차 클러스터의 알고리즘을 확인한다. 또한, 시뮬링크의 스테이트플로우 차트를 이용하여 상태도를 작성한다. 그리고 키보드 입력을 받아 클러스터 스크린의 메뉴 선택, 스테이트플로우 차트, 시뮬링크 프로그램을 작성하는 순서로 진행된다.

기본 스위치 입력을 통한 시뮬링크 프로그램을 작성하는 것을 상세하게 설명하면 다음과 같다.

먼저, 자동차의 클러스터를 제작하기 전에 PC에서 알고리즘의 동작확인을 위하여 매트랩의 시뮬링크에서 키보드 입력 라이브러리를 추가하여 키보드의 특징키를 자동차의 클러스터에 장착된 스위치 입력으로 시뮬링크를 설계한다. 그리고나서, 제작할 자동차 클러스터의 스위치 입력에 따른 동작 알고리즘을 시뮬링크의 스테이트플로우 차트를 이용하여 동작 알고리즘을 설계하고 작성한다. 그리고, PC 스테이션 환경에서 작성된 매트랩 시뮬링크를 실행하여 키보드 입력을 받아 클러스터 스크린의 메뉴 선택 시뮬레이션을 통하여 개발될 자동차 클러스터의 스위치 입력 동작 알고리즘을 검증한다.

3.2 매트랩 GUI 프로그램 개발

그림 2에서 1 단계 후, 시뮬링크의 프로그램과 GUI 프로그램, 스테이트플로우와 연동하여 시뮬링크를 제어하고, 스크린의 메뉴 선택과 현재의 클러스터 상태를 표시하는 차트 시뮬링크 프로그램을 구성하는 매트랩 GUI의 프로그램을 작성하는 2 단계를 포함한다.

설계 2 단계에서는, 매트랩 GUI를 통하여 자동차의 모든 상황을 표시하는 클러스터 시뮬링크 프로그램을 작성한 후, 매트랩 GUI가 매트랩 파일(Matlab File)을 이용한 M 프로그램과 모델기반설계

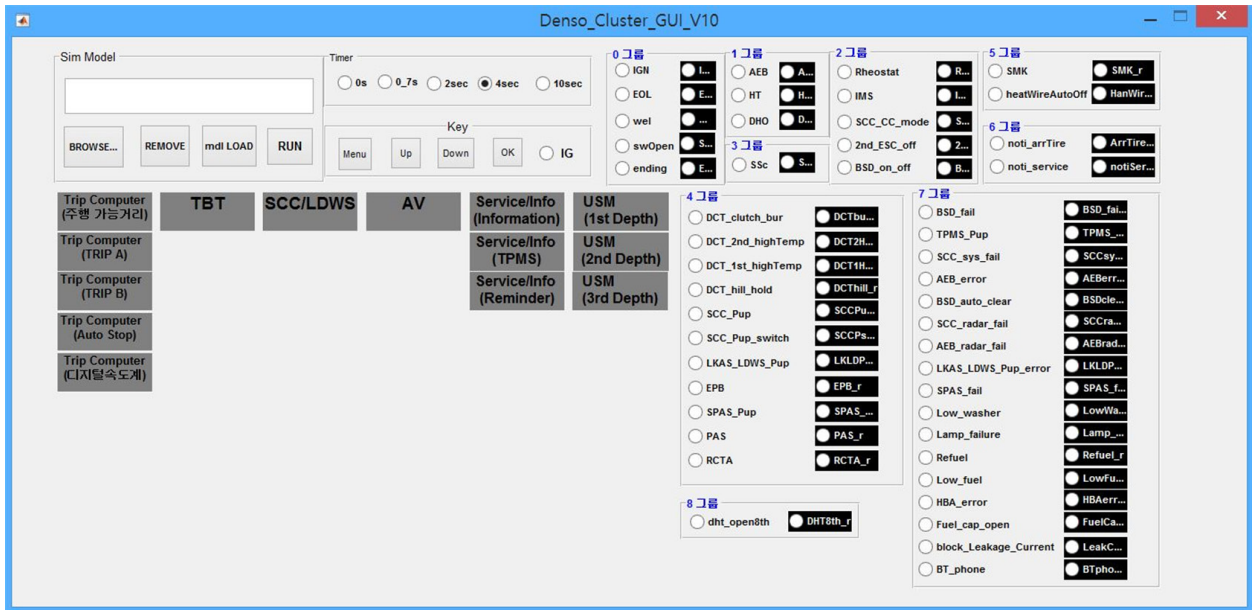


그림 5. 자동차 클러스터 감시 및 제어를 위한 GUI
 Fig. 5. GUI of Monitoring and Control for a Vehicle Cluster

S/W 도구를 갖는 시뮬링크 및 우선순위 알고리즘을 갖는 스테이트플로우와 연동하여 스크린의 메뉴 선택과 현재의 클러스터 상태 표시 차트 시뮬링크 프로그램을 설계한다.

그림 5는 모델기반설계 기법을 이용한 자동차 클러스터의 스테이션 설계 방법에 대한 화면표시와 입력제어를 위한 GUI 구성을 보여준다.

맷랩 GUI의 프로그램 작성에 대해 설명하면 다음과 같다. 먼저, 맷랩 GUI를 통하여 자동차의 모든 상황을 표시하는 클러스터 시뮬링크 프로그램을 작성한다. 맷랩 GUI 작성을 위하여, 개발된 자동차 클러스터에 입력되는 입력변수 버튼을 생성하고, 맷랩 시뮬링크의 입력변수 버튼에 상태에 따른 입력변수 값을 표시하는 상수블록을 먼저 생성시킨다.

다음은, 맷랩 GUI에 자동차 클러스터의 상태를 표시하는 이미지뷰(Image View)와 맷랩 GUI의 입력변수에 의한 동작 알고리즘 결과 이미지뷰를 생성시킨다. 여기서, 자동차 클러스터의 상태 표시 이미지뷰는, 클러스터와 맷랩 GUI의 간의 통신연결을 통하여 자동차 클러스터에 결과를 맷랩 GUI가 수신하여 표시하는 부분이고, 입력변수에 의한 동작 알고리즘 결과 이미지뷰는 맷랩 GUI의 입력변수 버튼을 통한 시뮬링크의 동작 알고리즘 결과를 나타내는 부분이다. 또한, 생성된 맷랩 GUI의 입력변수 버튼과 시뮬링크의 입력변수 상수블록, 맷랩 GUI의 동작 알고리즘 결과 이미지뷰와 시뮬링크의 동작 알고리즘 결과 출력변수는 서로 M 프로그램을 통하여 연동시킨다.

시뮬링크에서 자동차의 모든 입력변수 상태에 따른 동작

알고리즘은 입력변수의 중요도에 따라 각 그룹으로 나누어지며, 각 그룹은 순번에 따라 상위와 하위로 나누어진다. 그리고, 각 그룹은 상위, 하위 인터럽트와 같은 각 상황에 맞추어 인터럽트 수용 여부가 정해진다. 이렇게 정해진 자동차 클러스터의 동작 알고리즘에 따라 시뮬링크의 스테이트 플로우 차트를 설계하고 작성한다. 이렇게 작성된 스테이트플로우 차트를 이용하여 설계된 시뮬링크 프로그램과 맷랩 GUI는 시뮬레이션을 수행하여 자동차 클러스터의 동작 알고리즘을 검증할 수 있다.

3.3 PC 스테이션 제작

다음은 PC 스테이션을 제작하는 단계를 갖는다. 설계 2단계 후, 설계 3단계에서는 기본적인 메뉴 선택과 CAN통신 및 RS232 통신 인터페이스로 수신된 자동차 상태 알람에 대한 스테이트플로우 차트, 시뮬링크 프로그램을 통해 실제 자동차 클러스터의 동작 감시를 위한 PC 스테이션 환경을 제작한다. 따라서, 제3 단계는 기본적인 메뉴 선택과 CAN통신 및 RS232 통신 인터페이스로 수신된 자동차 상태 알람 스테이트플로우 차트 시뮬링크 프로그램을 개발하고 실제 자동차 클러스터의 동작 감시를 위한 PC 스테이션을 제작하는 단계이다.

그림 6은 그림 1에 대한 모델기반설계 기법을 이용한 자동차 클러스터의 스테이션 설계 방법에 따른 맷랩의 시뮬링크와 GUI를 이용하여 모델기반설계 기법을 기반으로 자동차 클러스터의 감시 및 제어 시스템에 대한 전체적인 구성을 보여준다.

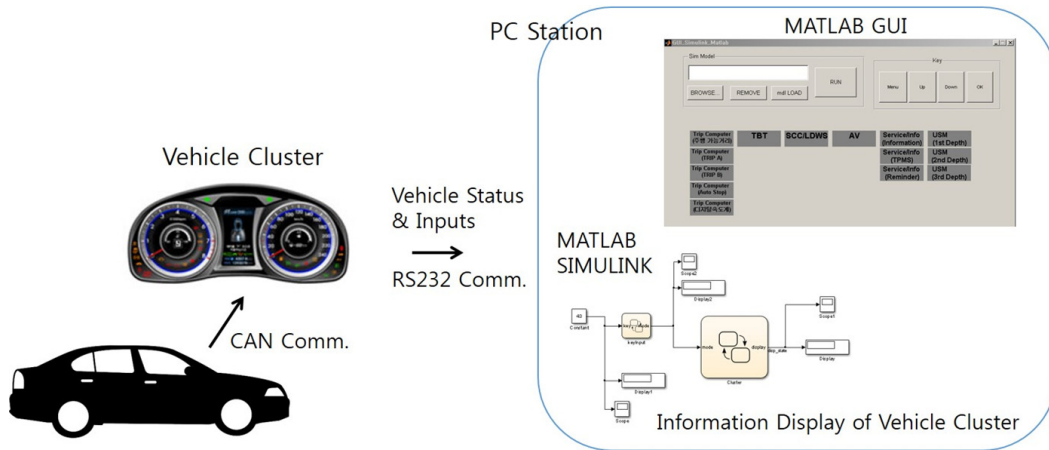


그림 6. 자동차 클러스터의 감시 및 제어 시스템에 대한 전체적인 구성
 Fig. 6. Overall Configuration of Monitoring and Control for a Vehicle Cluster

3.4 자동차 클러스터의 모든 정보 이식

3 단계 설계후, 모델기반설계 방식으로 설계가 가능하도록 자동차 클러스터의 모든 설계 정보를 툴박스의 명령함수를 통해 PC 스테이션 환경으로 정보를 이식하는 제4 단계가 포함된다. 텍스트 언어가 아닌 모델 기반 설계 MBD로 작성된 프로그램은 오토제너레이션(Auto Generation)을 통하여 사용자가 원하는 MCU(Micro Control Unit)에 맞게 TBD로도 쉽게 변경가능하다.

마지막으로, 각 그룹에 대하여 모든 스테이트플로우를 작성한 후 자동차 클러스터 MCU에 모든 정보를 이식하는 단계를 갖는다.

4. 결론

본 연구의 모델기반설계 기법을 이용한 자동차 클러스터의 스테이션 설계 방법은, 자동차 클러스터 설계에 있어서 기존의 텍스트 기반의 설계방식인 TBD와 달리 사용자 편의중심의 프로그램 수정과 문제해결이 용이한 MBD(Model Based Design) 방식으로 설계가 가능하도록 구현하였다. 장점으로는 다음과 같다.

(a) 분석 설계 단계에서 문제를 조기에 해결할 수 있고, 시험 단계에서 모든 경우의 테스트 케이스를 산출하기가 용이하며, 적절한 설계수법으로 잠재적인 버그를 조기에 발견하여 수정이 가능하기 때문에 기존의 텍스트 기반의 설계에 비해 프로그램 수정과 알고리즘 추가가 용이하다.

(b) 향후, 클러스터의 고도화로 복잡해지는 프로그램을 쉽게 처리 가능하고, 제작단계에서 자동화 코드생성으로 핸드코딩을 줄여 시간과 비용을 절감하고 제작단계에서 발생하는 핸드코딩의 휴먼에러를 방지할 수 있다.

(c) 자동차, 항공기, 선박과 같이 감시 제어용 클러스터를 포함한 이동수단에서와 같이 복잡한 알고리즘 코드를 다루는 유사 기술 분야에 MBD 방식의 클러스터 설계 기법을 도입할 경우, 작업자 간의 상호 시너지 효과를 통해 기술력이 향상되고 인프라가 확립됨에 따라 고신뢰성, 고안정성 클러스터 제품 개발이 가능하다.

위의 장점들에 대하여 제안된 MBD방식과 기존의 방법인 TBD 방식을 자동차 클러스터의 설계에서 정량적으로 결과 비교하기에는 어렵다. 알고리즘의 작성에 있어서 개개인의 차이가 있기 때문이다. 하지만, TBD방식의 단점을 없애기 위하여 MBD방식을 도입하는 것은 현재의 기술 추세인 것은 분명한 점이다. 본 연구와 직접 관련된 클러스터에 MBD를 도입한 연구논문은 아직 학술적으로 공개되어 있지 않고, 자동차 회사들은 보안으로 내용을 다루고 있어서 본 논문으로부터 얻어진 결과를 정량적으로 비교하기가 어려운 실정이다.

향후 연구는 자동차 클러스터뿐만 아니라 선박, 항공 분야에도 제안된 설계 방법을 적용하는 것이다.

References

[1] L. Marcil, "MBD & code generation: A cost effective way to speed up HMI certification," *IEEE/AIAA 30th Digital Avionics Systems Conference (DASC)*, pp. 8B1-1 -10, 2011.

[2] V. Socci, "Implementing a model-based design and test workflow," *IEEE International Symposium on Systems Engineering (ISSE)*, pp. 130-134, 2015.

- [3] J. C. Jensen, D. H. Chang, and E. A. Lee, "A model-based design methodology for cyber-physical systems," *7th International Wireless Communications and Mobile Computing Conference*, pp. 1666-1671, 2011.
- [4] L. Wang, G. Wang, W. W. X. Zhan, X. Liu, and P. Chen, "MBD-DSP: A model based design solution for DSP," *International Conference on Electrical and Control Engineering (ICECE)*, pp. 4561-4564, 2011.
- [5] L. Jowiak and J. Madsen, "Quality-driven model-based design of multi-processor embedded systems for highly demanding applications," *2nd Mediterranean Conference on Embedded Computing (MECO)*, pp. 1-3, 2013.
- [6] L. Ost, L. S. Indrusiak, S. Maatta, M. Mandelli, J. Nummi, and F. Moraes, "Model-based design flow for NoC-based MPSoCs," *17th IEEE International Conference on Electronics, Circuits, and Systems (ICECS)*, pp. 750-753, 2010.
- [7] A. Kim, S. Rhee, and H. Jang, "Lane Detection for Parking Violation Assessments," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 16, no. 1, pp. 13-20, 2016.
- [8] L. Fu, W. Wu, Y. Zhang, and R. Klette, "Unusual Motion Detection for Vision-Based Driver Assistance," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 15, no. 1, pp. 27-34, 2015.
- [9] H. Son, H. Ban, K. Yang, and Y. Rhee, "Implement of Intelligent Head-Up Display for Vehicle," *Journal of Korean Institute of Intelligent Systems*, vol. 20, no. 3, pp. 400-405, 2010.
- [10] H. M. R. Ugalde, D. Ojeda, V. L. Rolle, D. Andreu, D. Guiraud, J. Bonnet, C. Henry, N. Karam, A. Hagège, P. Mabo, G. Carrault, and A. Hernández, "Model-Based Design and Experimental Validation of Control Modules for Neuromodulation Devices," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 7, pp. 1551-1558, 2016.
- [11] T. Miyajima, H. Fujimoto, and M. Fujitsuna, "A Precise Model-Based Design of Voltage Phase Controller for IPMSM," *IEEE Transactions on Power Electronics*, vol. 28, no. 7, pp. 5655-5664, 2013.
- [12] K. Janschek and A. Morozov, "Dependability aspects of model-based systems design for mechatronic systems," *IEEE International Conference on Mechatronics (ICM)*, pp. 15-22, 2015.
- [13] G. Yang, M. Zhao, L. Hong, and Z. Wu, "SmartOSEK Based Design and Verification for In-vehicle Network System: A Model-Based Approach," *9th International Conference on Control, Automation, Robotics and Vision*, pp. 1-6, 2006.
- [14] M. Hufner, C. Sonntag, S. Engell, and S. Grobosch, "A customized design framework for the model-based development of engine control systems," *Industrial Electronics Society, IECON - 39th Annual Conference of the IEEE*, pp. 6916-6921, 2013.
- [15] S. D. Wall, "Model-based engineering design for space missions," *Aerospace Conference, Proceedings, IEEE*, vol. 6, pp. 3907-3915, 2004.
- [16] Available: <http://www.etnews.com/201309270318>
- [17] Available: <http://kr.mathworks.com/matlabcentral/fileexchange/?utf8=%E2%9C%93&term=Model+Based+Design>
- [18] Available: http://kr.mathworks.com/videos/designing-displays-using-model-based-design-98151.html?form_seq=conf1008&elqsid=1433916065028&potential_use=Education&country_code=KR

저자 소개



김동헌(Dong Hun Kim)

2001년 : 한양대학교 전기공학과(공학박사)
 2001년~2003년 : 미국 듀크 대학교 연구원
 2003년~2004년 : 미국 보스턴 대학교 연구원
 2004년~2005년 : 일본 동경대학교 박사 후 과정
 2012년~2013년 : 영국 에식스대학교 방문교수

2005년~현재 : 경남대학교 전기공학과 교수

관심분야 : 스윙 시스템, 이동로봇 경로계획, 지능 로봇 제어, 비선형 적응제어

Phone : +82-55-249-2629

Fax : +82-505-999-2161

E-mail : dhkim@kyungnam.ac.kr