

2. 선박 디젤엔진 원격제어 시스템 개발에 관한 연구

기관공학과 정 경 열
지도교수 류 길 수

오늘날의 대부분의 자동화 시스템은 마이크로프로세서 기술의 발달에 힘입어 시스템 구축의 유연성 및 사용자 편리성을 가진 고성능의 마이크로프로세서 기반 시스템으로 개발되고 있으며, 이로 인해 실시간으로 제어하는 동시에 많은 기능을 수행하는 것이 수월하게 되었을 뿐만 아니라 개발환경이 개선되고, 개발시간도 단축시킬 수 있게 되었다.

선박 분야에 있어서는 가혹한 해상 환경에서의 근무여건 개선 및 선원인력 감소에 대비한 선박 자동화 시스템이 계속적으로 개발되고 있는 추세이다. 특히, 운항 중인 선박의 자동화 시스템에 고장이 발생하면 승무원 자체의 기술력만으로 대처해야하기 때문에 선박운항에 직접적으로 관련있는 시스템 즉 선박 디젤엔진 원격제어 시스템과 같은 시스템의 고장은 대부분 인명과 선박자체의 손실로 직결된다. 그러므로 선박 자동화 시스템은 고성능·고신뢰성이 육상의 시스템보다 더욱 절실히 요구되고 있다.

조선 선진국에서는 이러한 선박 디젤엔진 원격제어 시스템을 마이크로프로세서 기반 시스템으로 개발하고, 원격제어실과 브릿지로부터 엔진의 순수제어 기능과 안전관리 기능들을 수행하도록 구성하여, 상품화하고 있으나 각 나라의 시스템들은 회사별로 독특한 방식에 의하여 독점 설계·생산되어 있으며, 보안이 철저하게 이루어져 있어서 서로 호환성이 없고, 당연히 설계의 정형화된 형식도 없다. 또한 제어 알고리즘도 각기 달라서 통일된 규격이 없으며 각각의 알고



리즘을 비교하여 최적의 기법을 찾는 것은 거의 불가능하다. 실제로 상품화되어 있는 선박 디젤엔진 원격제어 시스템의 동작기능을 살펴보면 NABCO사의 제품은 소형선박을 대상으로 하여 기관제어실에서만 조작 가능한 시스템으로 통신기능이 없는 단순구조로 이루어져 있으며, NORCON사나 LYNCSO MARINE사의 경우에는 대형선박을 대상으로 하여 여러 곳에서 엔진조작이 가능하도록 하거나 원격제어 기능과 엔진 안전 감시 기능을 분리하여 서로 다른 하드웨어 유니트에서 동작되도록 하는 분산시스템 형태로 구축하고 있다. 그러나 이들 제품에서 경제성, 신뢰성 및 사용자 편리성 등을 모두 고려한다면 시스템의 하드웨어를 보다 더 간결하게 하고 소프트웨어 기능을 강화시키되 실시간 성능이 떨어지지 않도록 하는 알고리즘 기법이 필요하다.

일반적으로 제어시스템 구축을 위해서는 먼저 시스템의 특성을 파악하고 이 특성에 적합한 마이크로프로세서를 결정하여 제어시스템의 하드웨어를 설계해야 한다. 하드웨어가 제작되면 입출력에 필요한 I/O 소자를 구동할 수 있도록 각각의 어드레스를 파악하고 구동드라이버를 작성한 후, 제어하고자 하는 목적에 맞도록 소프트웨어를 개발하여야 한다.

본 논문에서는 경제성, 신뢰성 및 사용자 편리성을 모두 도모하기 위한 마이크로프로세서 기반형 선박 디젤엔진 원격제어 시스템을 분산 형태로 구축하는 방법을 제안하였다. 이를 위해 선박 디젤엔진 원격제어 시스템을 세 개의 유니트 즉, Remote Control System Unit(RCSU), Bridge Control System Unit(BCSU), Safety System Unit(SSU)로 분산하여 구축하고, 또한 각 유니트의 하드웨어는 경제성과 실행속도를 고려하여 Intel 계열의 8088CPU를 이용하여 설계 제작하였다. 특히 소프트웨어를 개발함에 있어서 소프트웨어의 재사용, 하드웨어의 간결성, 실시간 성능 및 고 기능화를 목표로 하여 다음과 같은 방법을 도입하였다.

첫째, 태스크들은 프로그램길이가 길기 때문에 단순하게 속도면을 중시한 어셈블리 언어로만 개발하는 것은 개발시간이 많이 소요될 뿐만 아니라 디버그 및 수정 등이 극히 곤란하다. 따라서 소프트웨어의 구조가 명확하고 시스템 프로그램에까지 접근가능하며 범용성이 있는 C 언어를 이용하였다.

둘째, 이 태스크들은 상호 배타적인 실행, 시퀀스 실행, 비주기적인 실행 등 각기 실행 특성이 다르다. 따라서 이것들을 실시간내에 실행시키기 위해서는 각기 다른 실시간 운영기법을 포함시켜 프로그래밍하거나 아니면 실시간 운영기법을 별도의 시스템으로 프로그래밍하여 이 실시간 운영시스템이 상기 태스크들을 통합관리하면서 실행시키도록 프로그래밍 하여야 한다. 본 논문에서는 후자와 같은 실시간 멀티태스크 기법을 채택하여 이를 알고리즘화 하였다.

셋째, 통신방식을 구현함에 있어서 하드웨어를 간결하게 하기 위해 수신은 인터럽트 방식을 취하고 송신은 폴링방식을 취하도록 하는 통신프로토콜과 통신알고리즘을 제안하였다.

넷째, 시스템이 큰 메모리를 차지하는 것에 대비하여 메모리를 절약하는 방법으로 수신된 데이터를 처리하기 전에 임시로 저장해 놓는 버퍼영역을 가급적 적게 차지하도록 하는 가상 링버퍼를 사용하는 방법을 제안하고 이때에 발생할 수 있는 비정상적인 상태를 수정하기 위한 방법도 제시하였다.

다섯째, 디젤엔진의 원격제어 태스크들로서 엔진기동 및 엔진정지, 디젤엔진의 속도명령제어 특히 임계 회전수 영역이 존재하는 경우의 속도명령제어, 비상후진(crash astern), 비상전진(emergency ahead), 제어위치 절환, 경보를 다단계로 구별하여 표시하기 위한 플리커링(flickering) 알고리즘들을 제시하였다. 이중 비상전진 알고리즘은 메이커와 선박기관사들의 요청에 의해 추가된 기능으로서, 지금까지 개발된 제품들에는 없는 기능이며, 이 기능은 필요에 따라 선택 가능하도록 하였다.

본 논문에서는 선박 디젤엔진 원격제어 시스템을 개발함에 있어서 상기한 바와 같은 방법으로 시스템을 구현하여 다음과 같은 결과를 얻었다.

첫째, 실시간 멀티태스크 소프트웨어를 구축함으로써, 기존의 절차형 프로그램 작성법과 달리 대규모 시스템에 적합하며 기능의 추가 및 삭제가 용이하여 새로운 환경변화에 대한 확장성이 뛰어난 시스템을 구현하였다.

둘째, 태스크의 기능에 따라 타임슬라이스 방식과 이벤트드리븐 방식을 혼용하였으며, 이로 인해 불필요한 태스크의 절환시간을 줄임으로써 전체 시스템의 실시간 성능을 향상시켰다.

셋째, 통신 방식에 있어서 수신만을 인터럽트방식으로 처리하기 위해, 무효 데이터의 처리기법, 가상 링버퍼 개념 및 링버퍼 수정법을 도입함으로써 타이머 소자와 통신선의 수를 줄이고 메모리를 절약할수 있는 등 하드웨어를 간결하게 하였으며 데이터의 신뢰성을 확보하였다.

넷째, 시뮬레이션과 엔진장착 테스트를 수행함으로써 본 논문에서 개발된 선박 디젤엔진 원격제어 시스템이 정확하게 동작함을 확인할 수 있었으며, 본 논문을 통하여 개발된 디젤엔진 원격제어 알고리즘 및 실시간 멀티태스크 알고리즘은 실시간 운전이 필요한 플랜트 장치의 마이크로프로세서 기반형 자동화 시스템 구축에 대한 초석이 될 것으로 사료된다.

