

유연 IED를 위한 Network processor 플랫폼

*전현진, 이완규, 장태규
중앙대학교 전자전기공학부
e-mail : *jinitop@dmc.cau.ac.kr*, *tgchang@cau.ac.kr*

A Network processor based Flexible IED Platform

*Hyeon-Jin Jeon, Wan-Gyu Lee, Tae-Gyu Chang
School of Electrical Electronics Engineering
Chung-Ang University

Abstract

This paper proposes a flexible IED platform which is implemented with a network processor and a DSP. DSP algorithms are downloaded through the embedded Linux based network processor remotely from ethernet. This architecture gives the best flexibility to adaptively accommodate the various algorithms needed in the IED environment. The developed IED platform can simultaneously measure data of the maximum of forty channels. The developed IED platform shows the successful operation, which measures and transfers the 8 channels data of 16bit samples sampled at 3.84kHz per each channel. The detailed performance analysis of the developed IED platform shows the about 10% processing load of CPU running at 533MHz.

I. 서론

전력선의 전류, 전압, 위상, 주파수를 정밀 계측하는 것은 전력계통 보호계전의 기본 기능으로서 관련 알고리즘 및 구현 기술에 대한 많은 연구가 수행되어왔으며, 전력계통 보호계전 목적으로 광범위하게 적용되고 있다. 하지만 지금까지의 연구와 또한 실제로 적용되는

전력선 변수들에 대한 계측은 대부분의 경우 독립적 측정 기능을 확보하는데 국한되었다고 할 수 있다.

외부 source와 송수신 할 수 있는 processor와 결합된 형태의 유연 Intelligent Electronic Devices(IEDs)의 출현으로 다양한 알고리즘의 적용이 가능하게 되어, 전체 시스템의 정밀도와 신뢰성 등의 큰 향상을 가져오게 되었다[1]. 이러한 유연 IED를 위해서 다양한 형태의 보호계전 알고리즘들의 설정과 조합이 가능해야 하므로, 유연 IED 플랫폼 H/W의 설계는 기존의 ASIC과 같은 dedicated H/W 설계와는 달리 알고리즘의 재사용성 및 시스템 reconfigurability가 충분히 확보될 수 있도록 하는 설계기술이 필요하다.

본 논문에서는 확장성과 유연성을 가질 수 있도록 embedded Linux기반의 Network processor와 RTOS기반의 DSP를 결합한 형태의 H/W구조를 설계하였다.

II. 본론

2.1 Network processor 플랫폼 H/W 구조

서버와 통신을 할 수 있는 Ethernet port를 가진 INTEL의 Network Processor인 IXP425를 사용하였고 [2], 계전 알고리즘을 적용하기 위한 TI의 floating-point DSP인 TMS320C6711을 사용하였다[3]. Network processor(IXP425)에 PCI interface가 없는 DSP(TMS320C6711)를 연결하기 위해 PLX의 PCI

target interface chip인 PCI9030을 사용하였다[4]. 아래 그림 1은 전체적인 플랫폼의 블록도를 나타낸 것이다.

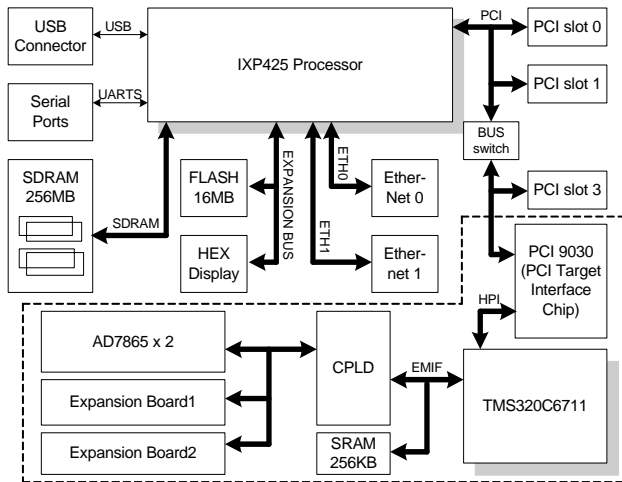


그림 1. Network Processor 플랫폼

IXP42에 embedded Linux를 올리기를 위해 16MB의 FLASH와 256MB의 SDRAM을 사용하였고, DSP에 RTOS를 올리기를 위해 256KB의 SRAM을 연결하였다. 전압신호를 얻기 위해 2개의 4채널 ADC를 연결하였고, 확장성을 위해 확장 커넥터를 만들었다. 따라서 최대 동시에 최대 40채널의 전압 데이터를 얻을 수 있다. DSP에서 다채널 전압신호를 ADC로부터 얻기 위해 CPLD를 사용하여 구현하였다.

2.2 유연 IED를 위한 S/W 설계

본 논문에서는 Network processor가 서버에서 새로운 알고리즘이 적용된 DSP프로그램을 다운받아 DSP에 다운로드하는 방식을 채택함으로써 다양한 알고리즘을 DSP에 적용할 수 있도록 S/W를 설계하였다. 즉, 새로운 알고리즘을 적용하기 위해 새롭게 하드웨어를 제작하거나 직접 DSP에 프로그램을 writing하는 방식이 아닌, 서버로부터 DSP프로그램을 받아 DSP로 다운로드하는 방식의 S/W다.

적용된 S/W는 embedded Linux상에 응용프로그램과 DSP를 위한 드라이버 모듈, DSP의 RTOS등으로 구성된다. S/W의 동작은 다음과 같다. 플랫폼이 부팅이 되면 서버로 전송받은 계전 알고리즘이 구현된 RTOS를 DSP로 전송한다. DSP의 RTOS가 실행되고, 결과 데이터는 서버로 전송된다.

III. Network processor 플랫폼 테스트

Network processor 플랫폼이 전압데이터를 서버로 전송하는 방식을 이용하여 플랫폼이 프로세서별로 어느 정도 부하를 보이는지 테스트 하였다.

DSP는 서버로부터 받은 RTOS를 동작시켜, 2개의 4채널 ADC가 채널당 60Hz의 전압신호를 64Hz로 sampling하고, Network processor는 sample된 전압데이터를 서버로 전송한다. 결과는 표1에 나타내었다.

Processor	프레임별 점유시간
IXP425(533MHz)	10ms미만/66.67ms
TMS320C6711(100MHz)	17.5us/66.67ms

표 1. 프로세서별 점유시간

IV. 결론 및 향후 연구 방향

본 논문에서는 유연 IED를 위해 다양한 계전 알고리즘을 적용하기 위한 목적으로 Network processor와 DSP를 이용한 H/W 및 S/W 구조를 제시하였다. 서버에서 알고리즘이 적용된 프로그램을 플랫폼에 전송하는 방식으로 알고리즘을 적용시킬 수 있도록 했다.

8채널의 전압 신호를 서버로 전송했을 때 100MHz로 동작하는 DSP의 프로세싱 load는 거의 없었고, 533MHz로 동작하는 Network processor는 10%정도의 부하가 있었다. 따라서 DSP에 복잡한 알고리즘을 실행시킬 여유가 충분하다는 것을 확인했다.

차후 논문에서는 complexity가 높은 알고리즘을 적용시켜 어느 정도 성능을 발휘하는지 확인하는 방향으로 연구가 진행될 것이다.

참고문헌

- [1] Ackerman, W.J, "The impact of IEDs on the design of systems used for operation and control of power systems", Power System Management and Control, 2002. Fifth International Conference on (Conf. Publ. No. 488), 17-19 April 2002 Page(s):445 - 450.
- [2] INTEL, "Intel® IXP42X Product Line of Network Processors and IXC1100 Control Plane Processor Datasheet"
- [3] TI, SPRS0880, "TMS320C6711, TMS320C6711B, TMS320C6711C FLOATING-POINT DIGITAL SIGNAL PROCESSORS.
- [4] PLX Technology, PCI 9030/C6000 AN, "Texas Instruments TMS320C6000 HPI bus to PCI 9030".
- [5] Texas Instrument, SPRA636A, "Application Using the TMS320C6000 Enhanced DMA"