

INTERNET 사용을 위한 SMART POWER DEVELOPEMENT SYSTEM

김기봉

STMicroelectronics, Schaumburg, U.S.A.

SMART POWER DEVELOPEMEN SYSTEM FOR USE IN INTERNET

kiBong Kim, Ph.D.

STMicroelectronics, Schaumburg, U.S.A.

Abstract - 본 논문에서는 stepper motor 나 BLDC motor 를 구동하기 위하여, 최신 microcontroller, PC, control IC board 등 으로 구성된 새로운 시스템을 설명 하는데 목적이 있다. 이 시스템에서는 인터넷으로 전송 되는 microcontroller program 를 이용하여 SPDS(smart power development system) 을 구동시키고, 이로인하여 dedicated drive IC 를 제어하여 stepper 또는 BLDC motor 를 구동시키게 되며, 원격지에 있는 사용자의 motor 를 제어 함을 목표로 하고 있다. 본 논문에서 사용된 drive IC 는 SGS-Thomson 제품으로써 stepper motor 용으로 L6205/6/7/8, 그리고 BLDC motor 용으로 L6235 이다. SPDS 는 ST92F120 16-bit microcontroller 를 사용하고 있다. 이 논문에서는 인터넷 Protocol Converter 에 의한 프로그램 전송, Operating System Program 및 Driver Software Program 의 생성 및 구성, Drive IC 의 구성 및 특징, SPDS 의 hardware 및 software 의 구성 등에 대하여 설명한다.

interface, JAVA runtime interface 로 구성되어진다. Remote 에 도달한 신호는 gateway 를 통해서 device 로 전달된다. 전송언어는 HTML 을 사용할수 있으며 remote 의 gateway 에서는 HTTP server, device access service mode, device link module, net modele 을 거치면서 전송되어진 신호를 처리하게 된다. 최종 제어대상인 device 는 net module 과 target application device 로 이루어지는데, net module 을 통해 server HTTP 통신을 하며 그 처리는 내장 micro-controller 에서 처리되어진다. 두 개의 net module 은 RS232 로 연결되어 있으므로 직접통신이 가능하며, 이것을 거친 신호는 최종적으로 target 인 application device 에 도착한다.

1. 인터넷을 통한 프로그램/데이터 전송

많은 경우, 다수의 사용자는 원격지(remote)에 위치해 있으면서, microcontroller 를 구동시킬수 있는 프로그램을 직접 제작/수정하기엔 여러가지의 이유로 어려운 상황에 있게되는 경우가 있거나, 한사람의 기술자가 원격지에 위치한 다수의 motor 를 구동시키도록 하여야 할 경우가 있다. 또 다른 경우에서, 개발과정에서 여러 가지 실험을 통하여 최적 성능을 검증하고 설정할 필요가 있는데, 이러한 목적을 충족 시키기 위한 방법으로 smart power development system 을 도입하게 된다. 데이터 전송을 이루게 되면 원격지에서 기술자가 microcontrller 의 프로그램을 직접 생성 시키지 않더라도 컨트롤 보드에 있는 microcontroller 를 구동 시키게 할수 있을것이며, 이러한 과정은 여러 사용자로 하여금 다수의 기술자가 상주하면서 관리하지 않아도 되는 경제적 이익을 이루게 된다. 최근에 활발히 이용되고 있는 인터넷의 기술은 각종 프로그램을 down load 하여 음악이나 영상, 문자, 기타 각종 정보를 보내어 전자기술이 전문한 인터넷 사용자가 편리하게 응용기기를 사용하게끔 해당 기술이 사용되고 있다. 이러한 방법을 적용하여, 입출력장치를 통해 microcontroller 에 프로그램으로 입력시키게 되면 microcontroller 는 작동하게 될 것이다. 그림 1은 전체적인 개념을 설명하는 BLOCK DIAGRAM 이다. Local에서 web browser 를 통해 보내는 데이터나 명령은 remote 의 web server 에 지정된 URL 에 HTTP 로 전송되는데, HTTP client, browser user

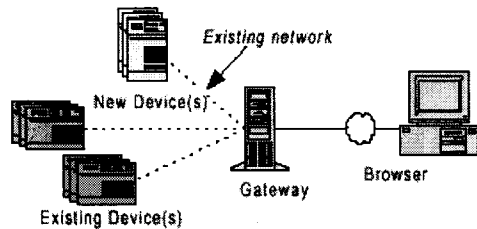


그림 1. 전송 및 시스템 개념도

그림 1에서와 같이 TCP/IP protocol 을 사용하면 인터넷을 통한 데이터 전송이 가능하므로 앞서 설명한 TCP/IP Converter 를 사용하여 HEX file 의 데이터를 변환하게 된다. 또한 사용자가 인터넷 조작이 가능하다면 WEB SITE 에 HEX FILE 을 올려놓고 이를 down load 하여 사용하게 된다. 제어대상의 컴퓨터에 도착한 데이터나 FILE 은 serial port 를 통해 SPDS 로 전송되며, 전송된 데이터나 file 은 SPDS 의 기억장소에 저장되어 프로그램된 바에 따라 작동을 하게된다. Baud rate 는 microcontroller 의 bit per second 와 같은 속도로 동기시켜야 하는데 실험에서는 57,600 baud rate 를 주로 사용하였다. 전송을 위하여 prompt command 에서 config file 을 만들어

```
mode com1:57600,e,8,1
```

```
copy filename.hex com1
```

의 명령을 사용한다. 이것은 data bit 를 8, parity를 even, stop bit 를 1, none flow control 로 지정함을 의미한다.

2. SPDS

Smart power development system (SPDS)는 16 bit ST92F120 micro-controller 를 중심으로한 여러부분의 주변장치로 이루어지며 그 외관은 그림 2와 같다. SPDS 에서 가장 중요한 부분이 되는 micro-controller 는 ST92F120 이며, 그 주요 데이터는 아래와 같다.

ST92F120
8/16-BIT FLASH MCU FAMILY WITH RAM, EEPROM AND J1850
BLPD

PRELIMINARY DATA

- ◇ 0 - 24 MHz Operation (internal Clock), 4.5 - 5.5 Volt voltage range
- ◇ Minimum instruction time: 83 ns (24 MHz internal clock)
- ◇ Internal Memory: Single Voltage FLASH up to 128 Kbytes, RAM 1.5 to 4 Kbytes, EEPROM 512 to 1K bytes
- ◇ DMA controller for reduced processor overhead
- ◇ 48 (77 on PQFP100 version) I/O pins
- ◇ 16-bit Standard Timer, two 16-bit independent Extended Function Timers (EFTs), two 16-bit Multifunction Timers
- ◇ 8-bit Analog to Digital Converter

이 micro-controller 에 있는 flash memory에는 SPDS 용의 Operating System 이 저장되어지며, serial port를 통해 전송되는 프로그램 데이터나 전송 명령들의 처리는 이 Operating System 에 의해 이루어지게되는데, bit wise 로 입력에 따라 ram영역에 저장되어지게 된다.



그림 2. SPDS

SPDS process 에서 가장 먼저 처리되어야 할 일은 전송 규격을 지정하는 것이다. 전항에서 설명된 protocol에 의해 원격지로부터 입력된 데이터는 이를 받은 computer 의 기억장소 에서 SPDS 의 micro-controller 의 기억장소로 옮겨야 하는데, 이에 쓰이는 전송방법은 가장 쉽게 hyper text transfer protocol 을 이용하는 방법을 들수

있다. 그림 3은 local 에서의 초기설정 모드 화면인데, 그림에서와 같이 baud rate 와 port 를 지정하게된다. ST92F120 는 PC 와의 통신에 필요한 asynchronous mode 와 device 와의 통신에 필요한 synchronous mode 를 동시에 가지고 있어 매우 편리하다. Asynchronous mode 의 경우, data sampling clock rate 가 16x 인데 그림 4에 설명된바와 같은 protocol 을 사용하게된다. 즉, data 와 clock은 비동기이며, transmitter 와 receiver 는 각각의 clock을 수신된 데이터의 sampling 에 사용하게 되고 각각의 수신 데이터는 하나의 clock 주기에 대하여 16 배로 sampling 된다.

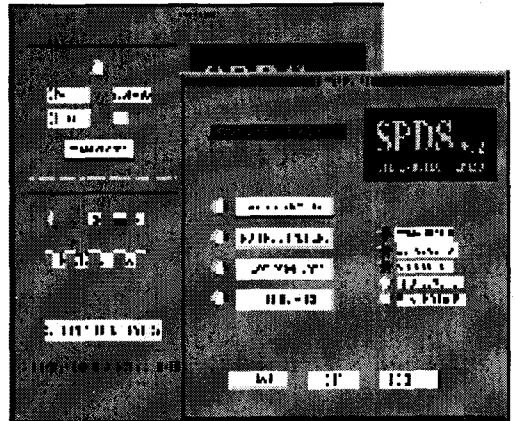


그림 3. 초기설정 모드 화면

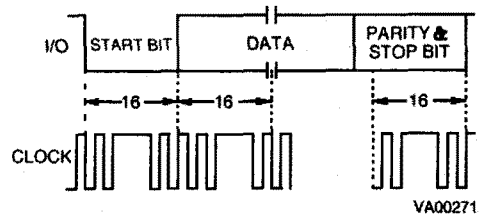


그림 4. Asynchronous mode protocol

Micro-controller 와 device board 의 통신은 data bit 에 전송되는 명령어에 따라 이루어지게 되는데, display 에 나타나는 speed profile 은 PC interface 에 있는 timer 에 의해 sampling 된 주기에 따라 명령이 전송된다. Lookup table 에 program 된 run profile 에 따라 구동되는 것이 기본 pattern 이지만, 때에 따라 속도를 변경할 필요가 생길수 있으며 이를 위하여 speed up/down 을 사용하게 된다. 이것은 multi functional timer 의 event control register 의 값을 변경시키는 방법으로 가능해진다. Display 를 위하여 전송되어진 register 값은 PC interface program 에서 속도로 변환 처리된다. 그림 5는 이러한 제어를 하기위하여 마련된 panel 을 나타내고있으며, 화면구성은 필요에 따라 그 내용을 변경할 수가 있다.

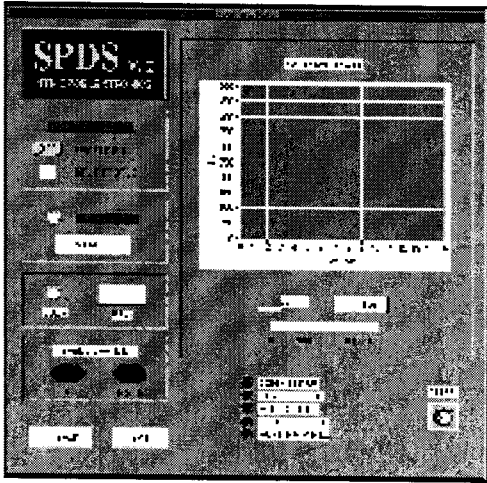


그림 5. Run Mode Display

3. L62xx Drive IC

종래에 주로 상용화되어 사용된 stepper drive IC 는 I^2L structure 에 의한 경우가 많았는데, 이 기술은 설계의 복잡성 및 난해성, 과다한 소비전력, lithography 에서의 응축력 제한 등의 문제가 따르고, 응용 설계시 시지연 소자를 구성하게 되는데, 이때 좁은 duty rate 및 고주파 신호에서는 에러가 발생하는 경우가 있다. 그리하여 새로운 BCD 기술 이 등장하게 되었는데, 이 BCD 기술을 적용한 IC를 smart power IC 라고 한다. [1] 여기에 소개된 device 들은 이것을 적용한 것이다.

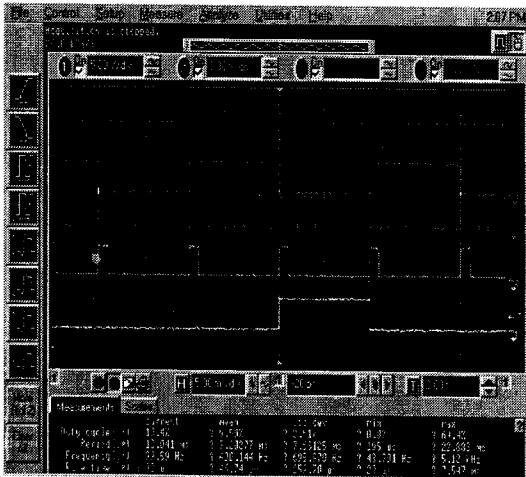


그림 6. 에러가 발생한 I^2L structure 의 출력모드

L620x series driver 에 의하여 구동되는 logic은 크게 나누어 normal mode, half wave mode, wave mode, custom mode 로 구분된다. L6208 을 제외하고

L6205/6/7 의 경우는 응용 프로그램을 사용하게 되면 micro stepping 구동이 되며, 이 경우는 device 를 system logic 과는 독립적으로 4개의 input 와 reference voltage 를 제어하여 mode 를 만들게 된다. 이 논문에서는 이 부분의 설명은 포함시키지 않았다. Half wave mode 는 그림 7에서 나타내어진 바와 같이 4 개의 출력이 비대칭적으로 발생하며, 이는 4개의 출력이 2쌍씩 대칭적으로 발생하는, 그림 8에서 나타내어진, normal mode 와 비교된다. Normal mode는 full wave mode 라고 하기도 하며, 그림 8 에서는 signal output 의 1A, 2A 를 나타낸 것으로써 1B, 2B 의 파형은 같은 주기에 위상이 반전된 것이다.

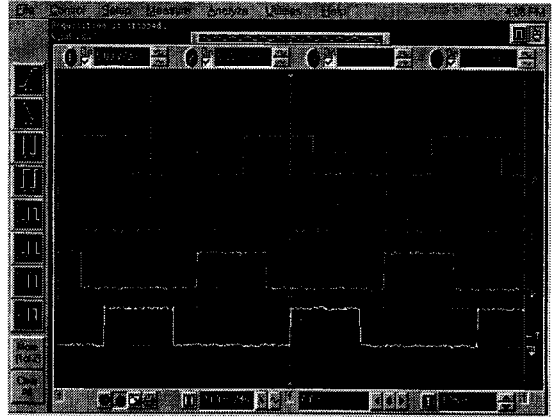


그림 7. Triggering sequence in 4 inputs (L6205-Half Wave Mode)

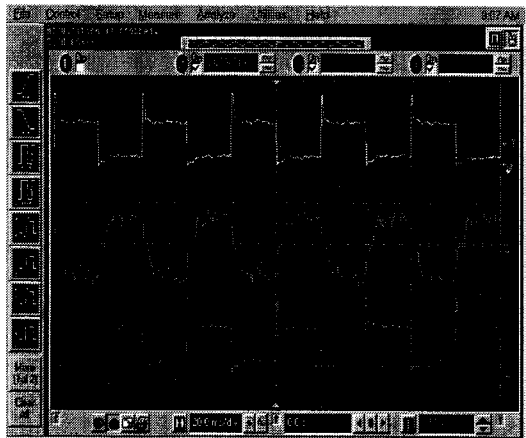


그림 8. Terminal voltage at 1A, 2A, and phase current in full wave mode (L6206)

그림9의 경우는 half wave mode 에서의 부하전류와 signal output 을 도시한것이며 이것을 그림 8과 비교하면 normal (full wave) mode 와 half wave mode 에서의 부하전류의 차이를 발견 할수 있을 것이다. Full wave mode 의 경우, 그림 8에서 나타내어진 바와 같이, 부하

전류의 형상은 정현파에 가깝게 되며, half wave mode에서는 dead time 이 주어져 있다. Half wave mode의 경우, 한 주기당의 step 수가 normal mode의 두배이며 따라서 구동이 부드럽다. 그림 10은 wave mode라 불리는 것으로써, sequence는 그림 10에 도시된 바와 같이 normal mode를 half wave mode 상에서 한 step shift시킨 것이다.

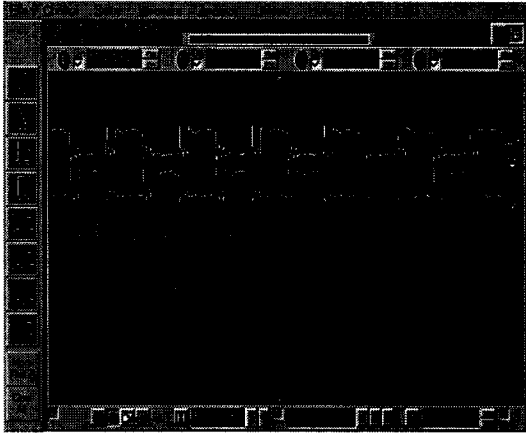


그림 9. Terminal voltage at 1A,1B,2A, and phase current in half wave mode (L6207)

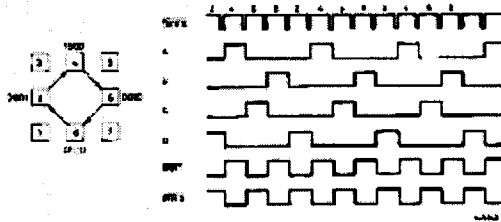


그림 10. Wave Mode에서의 동작 sequence 및 signal form

Brushless DC motor의 구동을 위한 L6235 device의 제어는 stepper와는 달리 별도의 구동 logic을 필요로 하지 않는다. 그러나 current limiting과 speed loop를 위한 별도의 제어가 필요하며, 이러한 제어 역시 SPDS를 통한 PC interface에 의해 이루어진다. 그림 11에서 보는 바와 같이 sensor에서 검출되는 부하전류는 V_{ref} 의 제어에 따라 comparator에서 thresh hold가 변화하게 된다. 즉, V_{ref} 의 입력량에 따라 전류의 limit가 변화하게 되므로 SPDS가 V_{ref} 를 제어하도록 하여 그것에 의해 PWM의 duty rate가 변화하도록 하고 그 결과 전류 limit를 변화시킨다. 이것은 동시에 tacho에서 출력되는 속도검출량을 SPDS에 지정된 속도값과 비교하여 V_{ref} 를 제어함으로써 속도 제어가 가능하여진다.

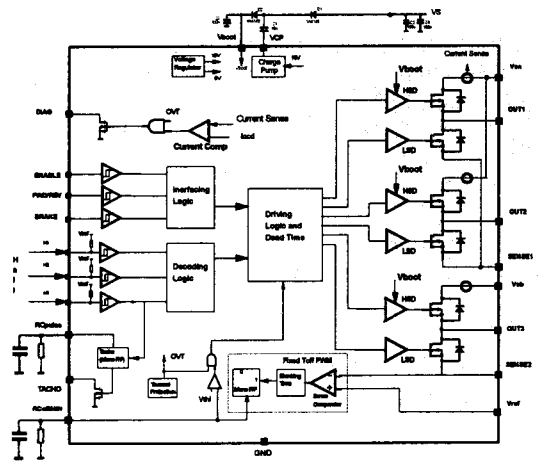


그림 11. Brushless DC motor의 구동을 위한 L6235 device schematic

4. REVIEW

이 논문에서는 전체적으로 Internet protocol converter, PC interface, 16 bit micro-controller를 사용한 SPDS, 그리고 driver board로 구성된 시스템의 개괄을 설명하는데 초점을 두었고, 그 목적은 local에서 제어하여 remote의 device를 구동을 실현하는데 있다. 이 목적을 제한된 지면으로 실현하고자 함에 있어서 어떤 한 부분을 부각시켜 설명하면 전체적 구성을 설명하는데 미흡한 반면, 세부적 구성의 특성을 설명하는데 소홀할수 밖에 없는 문제가 있다. 본 논문에서 강조된 것은 internet을 이용하여 데이터를 전송하여 이를 PC interface시켜 device를 구동케 하는 system software를 제작하는 연구내용이다. 즉, 이 논문에서는 protocol converter의 제작이 목적이 아니라 전송된 데이터를 처리하여 drive시키는 tool의 연구내용을 중심으로 설명되고 있다. 향후의 과제는 과정마다의 세부적 특성을 분리하여 설명하는 것, 특히 본 논문이 최종 목적하는 바대로 protocol에 의한 전반적 제어가 중심이 될것이다.

[참고 문헌]

[1] B.Murari, F.Bertotti, G.A.Vignola, "Smart Power I-technologies and applications", Springer 1995