
AJAX를 이용한 태양광 인버터의 모니터링 시스템 구현

권효상* · 양 오**

The Implementation of the Solar Inverter Monitoring System using an AJAX

Hyo-sang Kwon* · Oh Yang**

본 연구는 교육과학기술부와 한국연구재단의 지역혁신인력양성사업으로 수행된 연구 결과임

요 약

본 논문에서는 AJAX(Asynchronous JavaScript And XML)를 이용하여 태양광 인버터의 모니터링 시스템을 구현하고자 한다. AJAX는 DHTML(Dynamic Hyper Text Makeup Language)과 다른 자바스크립트 기술 등과 함께 RIA(Rich Internet Application)를 만들 수 있는 기술 중 하나이다. 이를 사용하여 일반 응용프로그램에 필적하는 강력한 응용프로그램을 만들 수 있다. 기존의 데이터 처리 기법으로는 동일 페이지에서 데이터의 Request와 Response를 동적으로 처리할 수 없다. 하지만 비동기 통신 방법인 AJAX 기반 기술을 이용하면 직관적이며 데이터 변동 상태를 실시간으로 제공하기 때문에, 한 페이지 내에서 모든 기능을 실행할 수 있고 페이지 전환 없이 데이터 전송량이 현저히 작은 AJAX를 사용하여 효율적인 관리 및 모니터링 할 수 있기 때문에 태양광 인버터모니터링 시스템에 AJAX를 적용하여 효율적인 모니터링 시스템을 구현하였다.

ABSTRACT

In this paper, the Solar Inverter will be monitored by using the AJAX(Asynchronous JavaScript and XML). AJAX is the one of the technologies that can make the RIA(Rich Internet Application) with DHTML(Dynamic Hyper Text Makeup Language) and other java script technology. By using this, a strong application program that is comparable to the general application program can be made. With an existing data-processing technique, the request and response of data can't be processed dynamically on the same page. However, real-time monitoring of data and operation statuses can be confirmed by using the AJAX an asynchronous method of communication. Also without changing the page, the amount of data transmission used the AJAX with significantly small amounts of data to build a Solar Inverter monitoring system that is able to efficiently handle management and monitoring, operating all functions within one page.

키워드

태양광인버터, 에이잭스, 웹서버, 실시간 웹 서버, 비동기통신

Key word

Solar Inverter Monitoring, AJAX, Web-Server, real-time web server, asynchronous communication

* 준회원 : 청주대학교 (hyosang500@nate.com)

접수일자 : 2012. 07. 24

** 정회원 : 청주대학교

심사완료일자 : 2012. 08. 17

I. 서 론

HTML은 특정한 페이지를 읽고자 할 때 전체의 구성 요소와 프로세스에 따라 연결된 모든 페이지들을 모두 로딩 할 때 소요되는 시간으로 멀티미디어 콘텐츠의 수용에 한계가 있다. 또한 페이지 기반의 클라이언트 모델에 대한 의존도가 높아 사용자가 페이지를 표시될 때까지 기다려야 되는 시간 낭비와 전체페이지를 반환해야 하는 불편함이 있다. 이러한 HTML의 단점들을 보완하고 사용자에게 조금 더 편한 UI기능을 제공하기 위해 등장한 것이 AJAX기술이다. AJAX는 새로운 기술이 아니라 웹에서 비동기 통신을 하는 기술과 DHTML, CSS, 자바스크립트를 포괄하는 개념이다. AJAX를 사용함으로써 일반적인 웹 응용프로그램에서 볼 수 있는 페이지 흐름을 변화시킬 수 있고 서버에 대한 요청이 찾아지게 되며 비 HTML 데이터의 응답을 작게 만들어 준다. 또한 AJAX 웹 응용프로그램은 보통의 웹 응용프로그램의 커뮤니케이션 흐름을 따르고 있으며 이에 새로운 형식의 요청을 추가한다. 서버에서 보면 이는 일반적인 페이지 요청처럼 보이지만 웹브라우저에서 보면 차이가 있다. 이는 완료시점에 페이지 리로드가 되지 않는 요청이며 사용자에게 의해 직접 시작될 필요가 없다. 많은 경우에 이런 AJAX 요청은 작을 것이며 필드를 확인하거나 어떤 데이터를 미리 가져오는 경우 같은 형태를 취하게 된다[1-3].

또한 AJAX는 지속적인 스펙트럼인 플래시 위젯을 포함하는 애플리케이션, 또는 리모팅 기술을 사용하지 않는 애플리케이션도 AJAX의 ‘일부분’으로 고려되므로 스펙트럼으로 간주하는 것이 효과적이다[4].

본 논문에서는 태양광 인버터를 모니터링 하기 위하여 LM3S8962 Evaluation Kit를 사용하여 간단한 웹 서버는 구축하였고 제한된 자원을 수용하면서 인버터를 웹 브라우저를 통해 서버의 데이터를 접근하는데 기존의 플러그인 도움 없이 웹브라우저만을 이용하여 응답성, 사용성을 높일 수 있는 AJAX를 사용함으로써 보다 동적인 페이지를 구성할 수 있고, 서버의 트래픽을 줄이므로 서버의 부하를 줄일 수 있다. 또한, 모든 브라우저에서 동일하게 작동하고, 확장성과 데이터 교환이 용이하며 빠른 응답이 보장되는 AJAX를 이용하여 임베디드 기반의 웹 서버를 구축하여 태양광 인버터를 실시간으로 모니터링 한다[5,6].

II. AJAX 기술 분석 및 장단점

2.1. AJAX와 기존방식의 차이점

그림 1은 기존의 웹브라우저를 방법을 구성한 것이다.

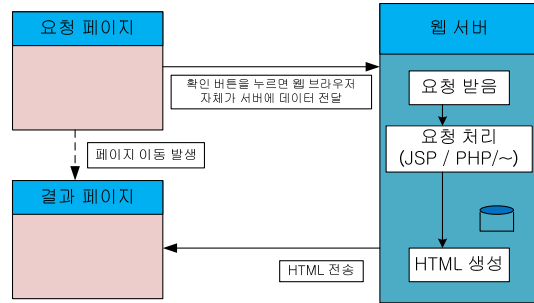


그림 1. 기존의 웹브라우저
Fig. 1 Existing web browsers

그림 1은 웹브라우저가 웹 서버에 요청을 전송하고 웹 서버는 JSP 등의 서버 어플리케이션을 사용해 사용자의 요청 처리 후 결과를 HTML로 생성해서 웹브라우저에 전송하고, 웹브라우저는 응답으로 받은 HTML을 분석한 뒤 그 내용을 화면에 그린다. 결과적으로 웹브라우저가 웹 서버와 통신을 하고, 요청 결과는 HTML로 생성되고 사용자 입장에서 페이지 이동이 발생하게 된다. 또한 기존의 웹 애플리케이션은 브라우저에서 폼을 채우고 이를 웹 서버로 제출(Submit)을 하면 하나의 요청으로 웹 서버는 요청된 내용에 따라서 데이터를 가공하여 새로운 웹 페이지를 작성하고 응답으로 되돌려준다. 이때 최초에 폼을 가지고 있던 페이지와 사용자가 이 폼을 채워 결과물로서 되돌려 받은 페이지는 일반적으로 유사한 내용을 가지고 있는 경우가 많다[7]. 결과적으로 중복되는 HTML 코드를 다시 한 번 전송 받게 됨으로써 많은 대역폭을 낭비하게 된다. 대역폭의 낭비는 금전적 손실을 야기할 수 있으며 사용자와 대화(상호 반응)하는 서비스를 만들기 어렵게도 한다. 또한 기존의 웹 어플리케이션 방식의 문제점은 서버 상의 스크립트/프로그램이 처리되면서 새로운 사용자입력 폼을 다시 보내는 동안 사용자는 대기해야 하고, 서버로부터 데이터를 다시 받을 때까지는 스크린 상에 없게 되므로 결국 상호대화성은 낮게 된다. 그러므로 사용자는 즉각적으로 응답을

받지 못하게 된다.

AJAX 방식인 그림 2는 사용자가 이벤트를 발생시키고 자바스크립트는 DOM을 사용해서 필요한 정보를 구한 뒤, XMLHttpRequest로부터의 요청을 알맞게 처리 후 결과를 XML이나 단순 Text를 생성해서 XMLHttpRequest에 전송한다. 서버로부터 응답이 도착하면 XMLHttpRequest 객체는 자바스크립트에 도착 사실을 알리고 자바스크립트는 응답 데이터와 DOM을 이용해 사용자 화면에 반영을 하게 된다.

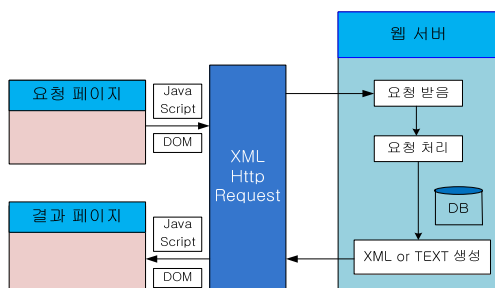


그림 2. AJAX 구성 방식
Fig. 2 AJAX Construction

결론적으로 사용자 입장에서는 페이지 이동이 발생되지 않고 페이지 내부 변환만 일어나게 된다. 또한, AJAX와 같은 웹 기술은 ActiveX 처럼 추가로 설치할 필요 없이 웹브라우저에 내장되어 있어서 편리하다. 사용자에게 번거롭지도 않고, 이미 표준처럼 잘 쓰이는 추세이다. 기존에 만들어져 있는 시스템을 AJAX로 구현하면 많은 Data처리가 필요한 페이지의 속도를 향상시킬 수 있고, 다양한 기능을 구현할 수가 있다. 이러한 요소는 User Experience의 향상으로 이어진다. AJAX는 이외에도 표준과 개방성, 확장성이 좋다. 웹 서버와 웹브라우저 사이의 데이터를 XML 형식으로 주고받기 때문에 문서 관리가 쉬워진다는 것과 기기나 브라우저에 구애받지 않는다는 장점이 있다[8].

2.2. AJAX 기술의 유용성

비동기 자바스크립트인 AJAX는 유용하게 사용되고 있다. 서버의 응답을 기다리지 않고 작업이 가능하므로 대기시간이 줄어들고, 이에 따라 서버의 부담이 줄기 때문이다. 이전의 동기방식은 이용자가 아이콘을 누르면 서버에 일일이 결과를 요청하고 이 결과를 받아서 브라우

우저 화면에 보여주었다. 당연히 사용자는 아이콘 하나를 누르고 서버에서 결과처리를 처리해서 보내줄 때까지 기다려야 했다. 그 동안 자바스크립트와 브라우저는 전달자 역할만 하고 있었다. 그렇지만 AJAX를 이용하면 일일이 서버에 묻지 않고 AJAX를 읽은 브라우저가 스스로 생각하면서 작업을 한다. 브라우저 안의 자바스크립트로 XMLHttpRequest를 통해 자료를 관리하기 때문에 다시 페이지를 불러올 필요가 없는 것이다. AJAX가 단순히 속도만 좀 빨라지는 것에 불과하다면 “혁명적인 차세대 기술”로 주목받기 어려울 것이다.

AJAX의 진정한 힘은 새로운 방식의 웹 접근이 가능하다는 점이다. AJAX는 브라우저의 HTML의 문서 안에서 이루어지기 때문에 프로그램 설치를 하거나 별도의 기능을 갖춘 새로운 창이 뜨지도 않는다. 일반 사이트의 HTML 문서를 보는 것처럼 일반 브라우저 화면 안에서 사용이 가능하다. 또한 브라우저 화면 안의 글이나 객체도 자유롭게 제어할 수 있다. 지금까지는 웹사이트 문서를 브라우저에 출력한 상태에서 자료의 위치를 사용자가 마음대로 편집하는 것이 어려웠다. 하지만 AJAX를 활용할 경우에는 이미 출력된 HTML 문서의 그림이나 글씨 위치를 마우스로 끌어다놓기로 손쉽게 바꿀 수 있다. 물론 프로그램 추가 없이 일반 브라우저 화면에서 이루어지는 AJAX의 기능이며 표 1은 AJAX 기술의 장점이다[9,10].

표 1. AJAX 기술의 장점

Table. 1 The advantages of AJAX technology

기술	장점
XMLHttpRequest	<ul style="list-style-type: none"> *본래의 AJAX로서 페이지에 대한 요청을 처리할 수 있다. *모든 HTTP 헤더를 설정하고 얻을 수 있다. *어떤 타입(GET, POST, PROPFIND 등)을 사용한 HTTP 요청도 가능하다. *모든 데이터 인코딩을 허용하는 POST 요청에 대한 모든 컨트롤을 지원한다.
IFrame	<ul style="list-style-type: none"> *POST와 GET 방식의 HTTP 요청을 처리할 수 있다. *모든 최신 브라우저를 지원한다. *비동기 파일 업로드를 지원한다.
Cookies	<ul style="list-style-type: none"> *많은 브라우저를 지원한다. *브라우저간에 구현 방법이 거의 차이가 없다.

XMLHttpRequest가 지원되지 않는 경우가 많이 있는데, 가장 흔한 경우는 오래된 버전의 브라우저를 사용하는 경우이다. 이런 경우 작업이 힘들어지는데, 이는 AJAX를 대체하는 방법이 없기 때문이 아니라 응용프로그램 내에서 특히 DOM을 조작하는 것이 잘 동작하지 않기 때문이다. 또 다른 문제의 경우는 사용하는 브라우저가 XMLHttpRequest를 제외한 필요한 모든 기능을 지원하는 경우이다. 이 문제는 IE가 ActiveXObject를 사용할 수 없는 모드일 경우나 또는 7.6 이전 버전의 Opera를 사용하는 경우에 발생한다. 만약 완전하게 래핑된 XMLHttpRequest를 사용하고 동기 호출을 사용하지 않는다면 쉬운 대체 방법을 사용할 수 있다. XMLHttpRequest 대신에 다른 접근 방법을 사용할 때 기억해야 할 것은 호환성에 있어서 커다란 도약을 얻게 되는 것은 아니라는 것이다.

이미 대부분의 브라우저들은 XMLHttpRequest 지원하고 있다. 이 기능을 지원하는 경우 자바스크립트를 사용하지 않게 설정한 브라우저는 이 기능을 지원하지 않는 브라우저를 실행하는 것처럼 동작하는 것이 아니라 AJAX 응용프로그램을 실행할 수 없는 브라우저로 인식을 하게 된다. 표 2은 AJAX 기술의 단점이다[11,12].

표 2. AJAX 기술의 단점
Table. 2 The disadvantages of AJAX technology

기술	단점
XMLHttpRequest	<ul style="list-style-type: none"> *IE5, IE6에서는 ActiveX를 필요로 한다. *Opera와 Safari의 경우 최신 버전에서만 가능하다. *브라우저간의 구현 방법이 약간 차이가 있다.
IFrame	<ul style="list-style-type: none"> *동기 요청을 할 수 없다. *IFrame 요청을 처리 할 수 있도록 서버 페이지를 구성해야 한다. *브라우저간에 구현 방법이 다르다. *브라우저 히스토리에 추가 기록을 남길 수 있다.(브라우저 기능에 따라). *모든 요청 데이터는 URL 인코딩되고 TJ 요청 크기가 증가한다.
Cookies	<ul style="list-style-type: none"> *비동기 요청을 할 수 없다. *많은 양의 요청 또는 결과를 처리할 수 없다. *쿠키 요청을 처리할 수 있도록 서버 페이지를 구성해야 한다. *클라이언트에서 쿠키 사용이 가능해야 한다. *GET HTTP 요청만 가능하다.

III. 모니터링 시스템 구현

3.1. 태양광 인버터 모니터링 구성 환경

사용된 MCU는 Luminary Micro社에서 개발된 Stellaris Family 8000계열중에 최상위 모델이다. 이것은 ARM@Cortex-M3 Core를 채택한 v7 architecture 기반으로 내부 PLL50MHz 성능을 보유한 프로세서이다. 이 MCU는 50MHz, 256KB single cycle FLASH memory, 64KB single cycle SRAM, 10bit ADC 4채널과, 아날로그 비교기, UART, SPI, I2C, MAC, CAN, RTC 등으로 구성되어 있다. 이 MCU의 가장 큰 특징은 10/100 Ethernet Controller(PHY, MAC)가 내장되어 있고, CAN(Controller Area Network) 2.0 A/B 1채널이 있는 것이다. 그리고 Motion Control PWM (Pulse Width Modulation) 3채널이 있다. LM3S8962는 산업자동화 분야나 자동차 분야에서 유용하게 사용이 가능한 많은 장점을 내포한 프로세서이다. 또한 이 MCU는 간단한 임베디드 Ethernet을 구현하기 좋다.

본 논문에서 모니터링을 위해 사용된 태양광 인버터의 사양은 표 3과 같다.

표 3. 태양광 인버터 사양
Table. 3 Photovoltaic inverter specification

모델	다쓰텍	DSP - 123K2
입력	PV Power	3kW
출력	동작전압	DC 280-500V
	정격출력	3KVA
	주파수 변동률	계통 60Hz±0.3Hz
	역률	0.95
	효율	96%
	출력 전류	14A
	독단 운전 방지	0.5초 이내
외부I/F	과부하	110%
	입출력장치	LCD 16*2 LED : 4EA Key SW : 4EA
외부I/F	기본	RS232C/RS485
	옵션	Ethernet

태양광 인버터를 모니터링 하기 위한 구성도는 그림 3과 같다.

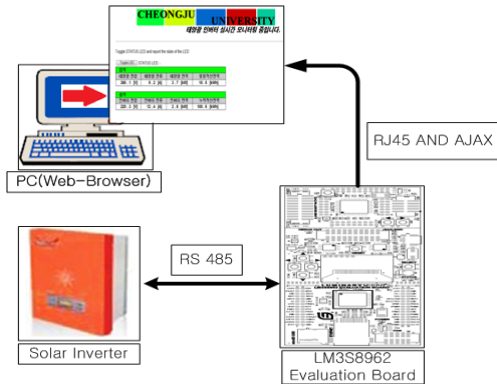


그림 3. 태양광 인버터 모니터링 구성도
Fig. 3 Solar inverter monitoring configuration

인버터와 LM3S8962(EVM)간의 통신 방법은 RS485로 구현하였고, LM3S8962와 PC는 RJ45를 사용하여 연결하였다.

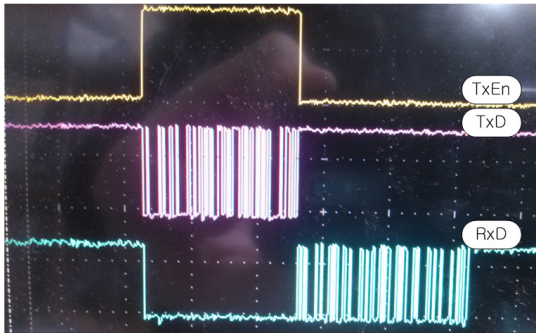


그림 4. RS485 통신 파형
Fig. 4 RS485 communication waveform

그림 4는 태양광 인버터와 EVM이 RS485 통신을 이용하여 전압에 해당하는 프로토콜 데이터를 인버터로부터 받기위해 통신을 하는 그림이다. 그림 4에서 TxD는 EVM이 태양광 인버터로 전압에 해당하는 프로토콜을 보내주고 있다.

인버터의 프로토콜은 총 3개의 프로토콜로 구성되어 있고 ST1, ST2, ST4의 정보를 인버터에 전송할 때 해당 프로토콜의 정보가 응답되는 프로토콜로 구성되어 있다. 즉, ST1을 인버터에 전송하면 PV전압, 전류, 전력량이 응답되어 진다. 태양광 인버터에 대한 모니터링용 프로토콜을 표 4에서부터 표 6에 나타내었다[13].

표 4. 솔라셀의 전압, 전류 프로토콜
Table. 4 protocol of PV voltage using and current

송신	^PaaaST1	aaa	ID number
수신	^D117aaa,bbb,ccc,ddd	aaa	ID number
		bbb	PV voltage[V]
		ccc	PV current[0,1A]
		ddd	PV generated power[0.1kW]

표 5. 발생전력 및 총전력량 프로토콜
Table. 5 Protocol of generated power and total generated power

송신	^PaaaST4	aaa	ID number
수신	^D416aaa,bbb,c ccccc	aaa	ID number
		bbb	generated power [0.1kW]
		ccc	total generated power[kWh]

표 6. 인버터의 전압, 전류, 주파수 프로토콜
Table. 6 Protocol of inverter voltage, current and frequency

송신	^PaaaST2	aaa	ID number
수신	^D219aaa,bbb,ccc,ddd,eee	aaa	ID number
		bbb	A line voltage[V]
		ccc	B line voltage[V]
		ddd	C line voltage[V]
		eee	line frequency[0.1Hz]
송신	^PaaaST3	aaa	ID number
수신	^D318aaa,bbbb,cccc,dddd	aaa	ID number
		bbbb	A line voltage[V]
		cccc	B line voltage[V]
		dddd	C line voltage[V]

3.2. 응답 처리

서버에 요청이 접수되면, 서버는 이에 합당한 작업을 수행 하여 응답을 보내려고 할 것이다. INVPOWERF 객체에 콜백 핸들러 함수를 할당한 것이 이에 해당한다. INVPOWERF은 단순히 응답 도착 시점을 알린다고 생각할 수 있지만, 사실 INVPOWERF은 응답이 시작되어 끝나기까지 여러 지점에서 응답의 상태를 알린다.

onreadystatechange라는 콜백 핸들러를 할당하였다. 이 함수는 INVPOWERF 객체의 준비 상태에서 최소한

한번은 호출될 것이다.

onreadystatechange에서 readyState 속성을 점검할 필요가 있으며, 이렇게 하는 이유는 요청이 현재 진행되고 있는지, 그리고 최종 결과를 처리 할 수 있는지를 확인하기 위해서다. readyState 속성을 점검 할 필요가 있으며, 이렇게 하는 이유는 요청이 현재 진행되고 있는지, 그리고 최종 결과를 처리할 수 있는지를 확인하기 위해서다. readyState에는 미리 정의되어 있는 표 7의 값 중 하나가 들어갈 것이다.

표 7. readyState 값
Table. 7 readyState Value

값	상태	설명
0	초기화되지 않았음	open()이 호출되지 않음
1	로드되고 있음	open()이 실행
2	로드되었음	send()가 실행
3	상호작용	서버가 데이터를 반환
4	완료	요청 완료, 서버도 데이터 전송을 완료

readyState가 4인지 점검하면 된다. readyState가 4라는 것은 요청이 끝났다는 것을 의미한다. 따라서 전형적인 콜백 함수는 다음과 같다.

```
function INVPOWER1()
{
var INVPOWERF = false;
function INVPOWERFComplete()
{
if(INVPOWERF.readyState == 4)
{
if(INVPOWERF.status == 200)
{
document.getElementById("INVPOWERF").innerHTML =
"<div>" + INVPOWERF.responseText + "</div>";
}
}
}
if(window.XMLHttpRequest)
{
INVPOWERF = new XMLHttpRequest();
}
else if(window.ActiveXObject)
```

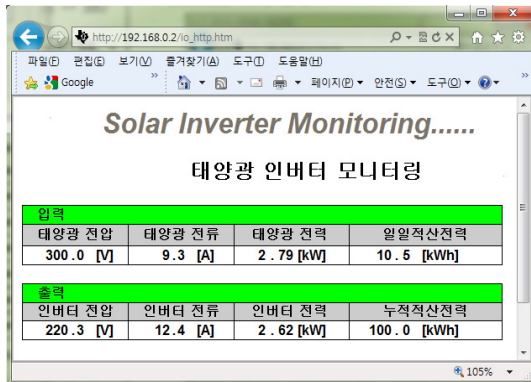
```
{
INVPOWERF = new
ActiveXObject("Microsoft.XMLHTTP");
}
if(INVPOWERF)
{
INVPOWERF.open("GET", "/InvPower1?id=" +
Math.random(), true);
INVPOWERF.onreadystatechange = INVPOWERFComplete;
INVPOWERF.send(null);
}
}
```

즉, readyState 속성의 값이 4인지를 확인하고, 값이 4라는 것은 응답이 완전히 도달하였으며 파싱 될 수 있다는 것을 의미한다. 값이 4일 경우, 파싱함수(parsing function)로 넘어간다. status 속성에는 요청의 HTTP 상태가 포함된다. 요청된 URL이 제대로 처리되었으면 유효한 HTTP GET이나 POST는 대개 200을 반환한다. URL이 없으면 404를 반환한다. 일반적으로 결과 코드가 200과 299사이라면 성공을 나타내고, 이외의 다른 코드는 오류이거나 브라우저에서 추가 액션이 이루어짐을 나타낸다. readyState와 status를 결합하면 요청이 제대로 완료되었는지 확인할 수 있다.

3.3. Web-Browser 모니터링 화면

Web-Browser에 Ajax와 HTML를 활용하여 태양광 인버터의 전압, 전류, 전력, 일일적산전력 등을 모니터링 할 수 있도록 간단한 홈페이지를 만들어 보았다. LM3S8962에서 태양광 인버터로 0.5초마다 전압, 전류, 전력 등에 해당하는 프로토콜을 보내게 된다. 인버터는 EVM 으로부터 받은 프로토콜을 분석하여 해당하는 값을 피드백 하여 준다. 인버터에서 EVM 으로 보내준 프로토콜은 특정 버퍼에 저장하게 되고, 버퍼에 저장된 프로토콜 값은 0.1초마다 기존에 저장된 인덱스 값과 비교하여 변경된 경우 변경된 값만 리플레쉬(Refresh) 하게 된다.

그림 5에서 웹브라우저는 0으로 초기화 되어 있고, 인버터가 동작을 시작하고 웹 서버와 RS485 통신을 하면 해당하는 태양광 전압, 전류 등이 표시되어 지는데 전류 등을 모니터링 할 수 있다.



입력			
태양광 전압	태양광 전류	태양광 전력	일일적산전력
300.0 [V]	9.3 [A]	2.79 [kW]	10.5 [kWh]

출력			
인버터 전압	인버터 전류	인버터 전력	누적적산전력
220.3 [V]	12.4 [A]	2.62 [kW]	100.0 [kWh]

그림 5. 태양광 인버터 모니터링 화면
Fig. 5 Solar inverter monitoring screen

IV. 결 론

AJAX는 브라우저와 서버의 통신을 추가 소프트웨어 없이 가능케 하기 때문에 액티브X가 가진 강력한 기능과 보안 성능을 모두 구현할 수 있고, 비동기식 Javascript 언어와 XML 등의 기타 웹 표준을 종합적으로 사용하는 기술로써 Web 2.0의 핵심 기술이다. 또한, 기존의 방식이 새로운 데이터를 불러오려면 전체 페이지를 다시 로딩해야 하는 것과 달리 AJAX는 SOAP (Simple Object Access Protocol), XML(eXtensible Markup Language)등 소프트웨어 통신 프로토콜을 이용해 브라우저의 사용자 화면 구성에 필요한 서비스만을 서버에 호출하며 웹 서버의 응답을 처리하기 위해 클라이언트 쪽에서는 자바스크립트를 이용해 그 결과를 화면에 적용할 수 있다.

본 논문에서 AJAX를 이용한 모니터링 시스템은 기존의 웹 서버 기술보다 동적인 페이지를 구성하는데 좋은 역할을 하였고, AJAX를 이용함으로써 한 페이지 내에서 전압, 전류, 전력 등을 페이지 전환 없이 모니터링 하기 위해 임베디드 시스템을 구현하였고, 간단한 웹 브라우저를 구현함으로써 태양광 인버터의 모니터링을 웹 브라우저를 통해 실시간으로 모니터링 할 수 있다.

감사의 글

본 연구는 교육과학기술부와 한국연구재단의 지역혁신인력양성사업으로 수행된 연구 결과임.

참고문헌

- [1] L.D Paulson, "Building rich web applications with Ajax", IEEE Computer Society, Vol. 38, No. 10, pp. 14-17, Oct 2005.
- [2] S. Salva, Univ. LIMOS, Aubiere. d'Auvergne, and P. Laurencot, "Automatic Ajax Application Testing", Internet and Web Applications and Services, 2009. ICIW '09. Fourth International Conference on, pp. 229-234, 24-28 May 2009.
- [3] <http://blog.naver.com/jinoxst/140021511970>
- [4] 마이클 마헤모프 "Ajax 디자인 패턴 웹 2.0", 한빛미디어, pp. 45-342, 2007년 7월.
- [5] 김성환, 박장현, 나철훈, "Ajax 인터페이스를 이용한 원격모니터링 임베디드 시스템", 한국지능시스템학회 학술발표논문집, 제 21권, 제 2호, pp.185-186, 2011년 12월.
- [6] 우준용, 김성래, 김은주, "Ajax를 이용한 웹의 응용과 효과", 한국멀티미디어학회, 2008년도 추계학술발표논문집, pp.673-676, 2008년 11월.
- [7] 정경진, 박홍진, 김영찬, "자바 이동 에이전트를 이용한 웹 서버 부하 모니터링" 한국정보과학회, 1999년도 가을 학술발표논문집, pp.489-491, 1999년 10월.
- [8] 정기훈, 염미령, 노삼혁, "아파치 웹 서버의 성능 비교 분석", 한국정보과학회, 2001년도 가을 학술발표논문집, pp.694-696, 2001년 10월.
- [9] <http://blog.daum.net/momnymn/10>
- [10] Joshua Eichorn, "예제로 쉽게 배우는 Ajax", 정보문화사, pp. 21-147, 7월, 2007년.
- [11] 데이브 크레인 and 에릭 파스카렐로, "Ajax 인 액션", 에이콘, pp.243-424, 5월, 2006년.

- [12] Smith, K, "Simplifying Ajax-style Web development",
IEEE Computer Society, Vol. 39, No. 5, pp. 98-101,
May 2006.
- [13] 제현우, 양오, "지그비 통신을 이용한 태양광 인버터의 원격 모니터링 시스템", 한국정보기술학회논문지, 제 10권, 제2호, pp.93-99, 2012년 2월.

저자소개



권효상(Hyo-sang Kwon)

2011년 : 청주대학교 전자공학부
학사 졸업
2011년 ~현재 : 청주대학교
전자공학과 석사과정

※ 관심분야 : 디지털 시스템 설계 및 DSP 응용 제어
시스템, 계통연계형 태양광 인버터 설계



양 오(Oh Yang)

1983년 : 한양대학교 전기공학과
학사 졸업
1985년 : 한양대학교 전기공학과
석사 졸업

1997년 : 한양대학교 전기공학과 박사 졸업
1985년 1월 ~ 1997년 8월 : LG산전 연구소 책임 연구원
1997년 9월 ~ 현재 : 청주대학교 반도체설계공학과
교수

2006년 ~ 2007년 : Texas A&M University 방문교수

※ 관심분야 : 디지털 시스템 설계 및 DSP 응용 제어
시스템 설계, 디지털 통신시스템 설계, 계통연계형
태양광 인버터 설계, ASIC 설계