

Programming the Wattmon

Programming and scripting will be used interchangeably throughout the document. Wattmon is built on a scripting language that is syntactically identical to PHP - we call it uPHP for Micro-PHP. It has a small but powerful subset of the functions and features found in the PHP language.

Why uPHP?

First perhaps it would be good to explain a bit about how and why we chose PHP. The goal of Wattmon is to provide a highly flexible platform that can be customized for pretty much any application imaginable keeping in mind very low memory availability. It became clear that a scripting language would be required to implement this, since having custom firmwares for various applications would be very tedious. The question then became: what would be the *best* scripting language that is well known today and can be deployed on a microcontroller with such limited resource? The choices are actually very small. Most modern scripting languages require quite a bit of RAM. So, we decided to roll our own scripting language from scratch and base it off a well known language.

The Wattmon interface is primarily browser based, and generates HTML pages on the fly from uPHP scripts. The methodology of HTML-PHP integration is well understood by most programmers and hence it allows you to become more productive quickly.

Challenges

The biggest challenge with the scripting language is its memory limitation. The WattmonPRO and MINI processor both have a total of 128kB (yes, kilobytes!) of RAM available to run a multitasking OS with a TCP/IP and USB stack, modbus polling engine, scheduler and more. During normal operation scripts have a maximum of about 30 kB of memory for variables, which means that as a programmer you need to be very conscious of memory allocation. We have introduced some special non-PHP compatible variable types to help with this. The [indexed_array\(\)](#) function lets you create arrays of a very specific type (int, byte, etc) which will reduce the amount of resources compared to hash arrays created with the [array\(\)](#) function.

Benefits

Having a scripting language that lets you modify and edit your program remotely is a very powerful thing. It lets you provide remote support to your clients and fix issues without needing to visit the site, and also makes your applications future-proof as you can at any time update the code base and integrate new features or protocols.

[Language Basics](#)

[uPHP Function Reference](#)

Last update:
2021/09/13 05:56

u.php:programming_the_wattmon http://wattmon.com/dokuwiki/u.php/programming_the_wattmon?rev=1486896060

From:
<http://wattmon.com/dokuwiki/> - **Wattmon Documentation Wiki**

Permanent link:
http://wattmon.com/dokuwiki/u.php/programming_the_wattmon?rev=1486896060

Last update: **2021/09/13 05:56**

