

# uPHP Reference

uPHP functions have an identical syntax to PHP functions in most cases. Below is a list of all the functions that have been implemented.

Click on the function name for further details:

FUNCTION NAME	PARAMETER(S)	RETURN	DESCRIPTION
<a href="#">adc_read</a>	<a href="#">int</a> <a href="#">channel</a>	<a href="#">int</a> ADC value	Read an onboard ADC <a href="#">channel</a>
<a href="#">array</a>	<a href="#">mixed</a> <a href="#">values</a> ...	<a href="#">array</a>	Create an <a href="#">array</a> , with optional <a href="#">values</a>
<a href="#">array_key</a>	<a href="#">array</a> , <a href="#">int</a> <a href="#">index</a>	<a href="#">string</a> key	Return the key for an <a href="#">array</a> <a href="#">index</a>
<a href="#">array_keys</a>	<a href="#">array</a> with key/value pairs	<a href="#">array</a> of keys	Return keys for an <a href="#">array</a> that has key/value pairs
<a href="#">base64_decode</a>	<a href="#">string</a> <a href="#">base64</a>	<a href="#">string</a> decoded or <a href="#">int</a> 0	Decode a base64-encoded <a href="#">string</a>
<a href="#">base64_encode</a>	<a href="#">string</a> to encode	<a href="#">string</a> base64 or <a href="#">int</a> 0	Return the base64-encoded version of a <a href="#">string</a>
<a href="#">call_user_func</a>	<a href="#">string</a> <a href="#">function_name</a> , <a href="#">mixed</a> <a href="#">parameters</a> ...	<a href="#">mixed</a> result	Call a user defined function with optional <a href="#">parameters</a>
<a href="#">charat</a>	<a href="#">string</a> , <a href="#">int</a> <a href="#">index</a>	<a href="#">int</a> ASCII code	Return the ASCII code for a character in a <a href="#">string</a> at an <a href="#">index</a>
<a href="#">chdir</a>	<a href="#">string</a> <a href="#">directory</a>	<a href="#">int</a> 0=OK	Change the current directory
<a href="#">chr</a>	<a href="#">int</a> <a href="#">code</a>	<a href="#">string</a> 1 character	Return the character for an ASCII <a href="#">code</a>
<a href="#">cos</a>	<a href="#">number</a> <a href="#">radian_angle</a>	<a href="#">float</a> cosine	Return cosine of a <a href="#">radian_angle</a>
<a href="#">debug</a>	<a href="#">string</a> <a href="#">output</a>		Print to debug output
<a href="#">debugout</a>	<a href="#">int</a> 0 or 1		Enable or disable debug messages
<a href="#">die</a>			Kill the script
<a href="#">disk_free_space</a>		<a href="#">int</a> KiloBytes	Return free space on microSD card
<a href="#">disk_total_space</a>		<a href="#">int</a> KiloBytes	Return total space on microSD card
<a href="#">error_reporting</a>	<a href="#">int</a> <a href="#">verbosity</a>		Set the debug output level
<a href="#">exec</a>	<a href="#">string</a> <a href="#">script</a> , <a href="#">int</a> <a href="#">delay</a>		Run a <a href="#">script</a> with an optional <a href="#">delay</a>
<a href="#">exec_action</a>	<a href="#">mixed</a> <a href="#">action</a>	<a href="#">int</a> 1=OK	Triggers a manually executable <a href="#">action</a> by id or name
<a href="#">explode</a>	<a href="#">string</a> , <a href="#">string</a> <a href="#">delimiter</a>	<a href="#">array</a>	Turn a <a href="#">string</a> into an <a href="#">array</a>
<a href="#">f485open</a>	<a href="#">int</a> <a href="#">baud</a> , <a href="#">int</a> <a href="#">parity</a>	<a href="#">int</a> handle or 0	Open the RS-485 port at the specified <a href="#">baud</a> rate and <a href="#">parity</a>
<a href="#">fclose</a>	<a href="#">int</a> <a href="#">handle</a>		Close a file, stream or socket
<a href="#">feof</a>	<a href="#">int</a> <a href="#">handle</a>	<a href="#">int</a> 1 or 0	Test if no more data is available in a file, stream or socket
<a href="#">fgets</a>	<a href="#">int</a> <a href="#">handle</a> , <a href="#">int</a> <a href="#">size</a>	<a href="#">string</a> or <a href="#">int</a> -1	Return a single line from a file, stream or socket, with optional <a href="#">size</a> limit

FUNCTION NAME	PARAMETER(S)	RETURN	DESCRIPTION
<a href="#">file_exists</a>	<a href="#">string filename</a>	<a href="#">int</a> 1 or 0	Check if a file exists
<a href="#">filesize</a>	<a href="#">string filename</a> or <a href="#">int handle</a>	<a href="#">int</a> bytes	Return the size of a file, or the number of unread bytes in a stream or socket
<a href="#">findfirst</a>	<a href="#">string pattern</a> , <a href="#">int attributes</a>	<a href="#">array</a> first file found	Start searching the current folder for files matching a <a href="#">pattern</a> and <a href="#">attributes</a>
<a href="#">findnext</a>		<a href="#">array</a> next file found	Return next matching file information (after a <a href="#">findfirst</a> )
<a href="#">firmwareupdate</a>			Initiate a firmware update sequence and reboot the device
<a href="#">floatval</a>	<a href="#">mixed value</a>	<a href="#">float</a> value or <a href="#">int</a> 0/1	Return the <a href="#">float</a> value of a <a href="#">number</a> or <a href="#">string</a>
<a href="#">flush</a>			Flush current output to the browser
<a href="#">fopen</a>	<a href="#">string filename</a> , <a href="#">string mode</a>	<a href="#">int</a> handle or 0	Open a file for reading or writing
<a href="#">fread</a>	<a href="#">int handle</a> , <a href="#">int bytes</a>	<a href="#">string</a> or <a href="#">int</a> 0	Read <a href="#">bytes</a> from a file, stream or socket
<a href="#">freemem</a>		<a href="#">int</a> bytes	Return free memory space
<a href="#">freestack</a>		<a href="#">int</a> bytes	Return free stack space
<a href="#">fseek</a>	<a href="#">int handle</a> , <a href="#">int offset</a> , <a href="#">int whence</a>		Position the file pointer in an open file
<a href="#">fseropen</a>	<a href="#">int baud</a> , <a href="#">int blocking</a> , <a href="#">int invert</a> , <a href="#">int parity</a>	<a href="#">int</a> handle or 0	Open the serial port at the specified <a href="#">baud</a> rate with optional parameters
<a href="#">fsockopen</a>	<a href="#">string host</a> , <a href="#">int port</a> , <a href="#">int timeout</a>	<a href="#">int</a> handle or 0	Open an internet socket connection with optional <a href="#">timeout</a>
<a href="#">ftell</a>	<a href="#">int handle</a>	<a href="#">int</a> position	Return the current position of a file read/write pointer
<a href="#">function_exists</a>	<a href="#">string function_name</a>	<a href="#">int</a> 1 or 0	Check if a function exists (custom or native)
<a href="#">fwrite</a>	<a href="#">int handle</a> , <a href="#">mixed data</a>	<a href="#">int</a> bytes written or -1	Write <a href="#">data</a> to a file, stream or socket
<a href="#">get3gstat</a>		<a href="#">array</a>	Get cellular data connection status information
<a href="#">getcwd</a>		<a href="#">string</a> path	Get the current directory
<a href="#">getethstat</a>		<a href="#">array</a>	Get Ethernet connection status information
<a href="#">getmac</a>		<a href="#">string</a> MAC	Get the Wattmon's MAC address
<a href="#">getusbstat</a>		<a href="#">array</a>	Get USB host status information
<a href="#">header</a>	<a href="#">string header_data</a>		Add to HTTP header
<a href="#">htmlspecialchars</a>	<a href="#">string data</a>	<a href="#">string</a> converted	Convert special characters for display in HTML
<a href="#">ieee754toint</a>	<a href="#">float value</a>	<a href="#">int</a> representation	Convert a <a href="#">float value</a> to an IEEE-754 encoded <a href="#">integer</a> (32 bit)
<a href="#">implode</a>	<a href="#">array</a> , <a href="#">string delimiter</a>	<a href="#">string</a>	Turn an <a href="#">array</a> into a <a href="#">string</a>

FUNCTION NAME	PARAMETER(S)	RETURN	DESCRIPTION
<code>include</code>	<code>string filename</code>		Include a file within the current script at the current location
<code>indexed_array</code>	<code>int type</code> , <code>int size</code>	<code>array</code>	Create an <code>array</code> of a specific <code>type</code> and <code>size</code>
<code>ini_get</code>	<code>string filename</code> , <code>string section</code> , <code>string key</code> , <code>mixed default</code>	<code>mixed</code> value	Get a value from an INI file
<code>ini_get_array</code>	<code>string filename</code> , <code>string section</code>	<code>array</code>	Get a group of parameters from an INI file as an <code>array</code>
<code>ini_put_array</code>	<code>string filename</code> , <code>array data</code> , <code>string section</code>		Write a group of parameters to an INI file from an <code>array</code>
<code>ini_set</code>	<code>string filename</code> , <code>string section</code> , <code>string key</code> , <code>mixed value</code>	<code>int</code> 1=OK	Set a <code>value</code> in an INI file
<code>inttoieee754</code>	<code>int</code> representation	<code>float</code> value	Convert an IEEE-754 encoded <code>integer</code> representation (32 bit) to a <code>float</code>
<code>intval</code>	<code>mixed</code> value	<code>int</code> value	Return the <code>integer</code> value of a <code>number</code> or <code>string</code>
<code>is_array</code>	<code>mixed</code> variable	<code>int</code> 1 or 0	Check if a <code>variable</code> is an <code>array</code>
<code>is_float</code>	<code>mixed</code> variable	<code>int</code> 1 or 0	Check if a <code>variable</code> is a <code>float</code>
<code>is_int</code>	<code>mixed</code> variable	<code>int</code> 1 or 0	Check if a <code>variable</code> is an <code>integer</code>
<code>is_numeric</code>	<code>mixed</code> value	<code>int</code> 1 or 0	Check if a <code>value</code> is numeric ( <code>int</code> , <code>float</code> or numeric <code>string</code> )
<code>is_string</code>	<code>mixed</code> variable	<code>int</code> 1 or 0	Check if a <code>variable</code> is a <code>string</code>
<code>isset</code>	<code>mixed</code> variable	<code>int</code> 1 or 0	Check if a <code>variable</code> exists
<code>json_encode</code>	<code>array</code> , <code>int</code> method	<code>string</code>	JSON encode an <code>array</code> into a <code>string</code> , with optional <code>method</code>
<code>ln</code>	<code>number</code> number	<code>float</code> log <sub>e</sub>	Return the natural logarithm of a <code>number</code>
<code>log</code>	<code>string</code> output, <code>string</code> file		Print to the System Log (or optional <code>file</code> )
<code>log10</code>	<code>number</code> number	<code>float</code> log <sub>10</sub>	Return the base 10 logarithm of a <code>number</code>
<code>mail</code>	<code>string</code> recipient, <code>string</code> subject, <code>string</code> body	<code>int</code> 0 or SMTP error code	Send an email
<code>max_execution_time</code>	<code>int</code> seconds		Set the maximum execution time for the current script
<code>mb_add_dev</code>	<code>int</code> id, <code>int</code> type, <code>string</code> name, <code>int</code> poll_interval, <code>int</code> status	<code>int</code> 0=OK	Add a device to the list of polled devices
<code>mb_delete_device</code>	<code>int</code> id	<code>int</code> 1=OK	Delete a device from the list of active devices
<code>mb_get_dev_by_id</code>	<code>int</code> id	<code>array</code>	Return modbus device details by <code>id</code>
<code>mb_get_dev_by_index</code>	<code>int</code> index	<code>array</code>	Return modbus device details by <code>index</code>
<code>mb_get_dev_by_name</code>	<code>string</code> name	<code>array</code>	Return modbus device details by <code>name</code>

FUNCTION NAME	PARAMETER(S)	RETURN	DESCRIPTION
<a href="#">mb_get_dev_info</a>	<a href="#">int</a> type	<a href="#">array</a>	Return modbus device details by <a href="#">type</a>
<a href="#">mb_get_role_array</a>		<a href="#">array</a>	Return an <a href="#">array</a> of all roles and their values
<a href="#">mb_get_status_by_role</a>	<a href="#">int</a> role	<a href="#">int</a> 1=OK	Return status of the device attached to the <a href="#">role</a>
<a href="#">mb_get_val_by_role</a>	<a href="#">int</a> role	<a href="#">number</a>	Return value of the <a href="#">role</a>
<a href="#">mb_num_devices</a>		<a href="#">int</a>	Return number of devices on the modbus
<a href="#">mb_queue_command</a>	<a href="#">mixed</a> values ...	<a href="#">array</a> of numbers	Queue a sequence of characters to the rs485 bus and get but ignore the reply
<a href="#">mb_scan_complete</a>		<a href="#">int</a> 1=complete, 0=ongoing	Check to see if a modbus scan has completed
<a href="#">mb_scan_percent</a>		<a href="#">number</a> percent completed	Return scan percentage completed
<a href="#">mb_send_command</a>	<a href="#">mixed</a> values ...	<a href="#">array</a> of numbers	Send a sequence of characters to the rs485 bus and get a reply
<a href="#">mb_set_dev_var</a>	<a href="#">string</a> name or <a href="#">int</a> id, <a href="#">string</a> variable, <a href="#">mixed</a> value	<a href="#">int</a> 1=OK	Set a <a href="#">variable</a> on a modbus device
<a href="#">mb_set_val_by_role</a>	<a href="#">int</a> role, <a href="#">number</a> value	<a href="#">int</a> 1=OK	Set a <a href="#">role</a> <a href="#">value</a> on a modbus device
<a href="#">mb_start_scan</a>	<a href="#">int</a> start, <a href="#">int</a> end		Initiate an automatic scan of the modbus
<a href="#">md5</a>	<a href="#">string</a> input	<a href="#">string</a> 32 characters	Calculate the MD5 hash of a <a href="#">string</a>
<a href="#">md5_file</a>	<a href="#">string</a> filename	<a href="#">string</a> 32 characters	Calculate the MD5 hash of a file
<a href="#">mem_dump</a>			Write the current memory map to /dump.txt
<a href="#">mem_usage</a>			Write memory usage to standard output
<a href="#">microtime</a>		<a href="#">int</a> ms	Return the number of milliseconds since boot
<a href="#">mkdir</a>	<a href="#">string</a> pathname	<a href="#">int</a> 0 or error code	Make a directory
<a href="#">mktime</a>	<a href="#">int</a> hour, <a href="#">int</a> minute, <a href="#">int</a> second, <a href="#">int</a> month, <a href="#">int</a> day, <a href="#">int</a> year	<a href="#">int</a> seconds	Return the Linux Timestamp for a given date and time
<a href="#">net_disable3g</a>			Disable 3G support for the dongle
<a href="#">net_enable3g</a>			Enable 3G support for the dongle
<a href="#">netstat</a>		<a href="#">array</a>	Get Ethernet information
<a href="#">number_format</a>	<a href="#">mixed</a> number, <a href="#">int</a> digits	<a href="#">string</a> formatted	Return the <a href="#">string</a> value of a <a href="#">number</a> formatted to a particular precision
<a href="#">nvram_backup</a>	<a href="#">string</a> filename	<a href="#">int</a> bytes written or 0=error	Backup the contents of <u>NVRAM</u> to a file on the SD Card
<a href="#">nvram_defrag</a>			Defragment <u>NVRAM</u> to optimise it

FUNCTION NAME	PARAMETER(S)	RETURN	DESCRIPTION
<code>nvrump_dump</code>			Dump the contents of <code>NVRAM</code> to standard output
<code>nvrump_free</code>		<code>int</code> bytes	Return the number of bytes available in <code>NVRAM</code>
<code>nvrump_get</code>	<code>string</code> key	<code>mixed</code> value	Get a value from <code>NVRAM</code>
<code>nvrump_restore</code>	<code>string</code> filename		Restore the contents of <code>NVRAM</code> from a file
<code>nvrump_set</code>	<code>string</code> key, <code>string</code> value	<code>int</code> 1=OK	Set a <code>key</code> and <code>value</code> in <code>NVRAM</code>
<code>nvrump_unset</code>	<code>string</code> key	<code>int</code> 1=OK	Clear a <code>key</code> from <code>NVRAM</code>
<code>ord</code>	<code>string</code> character	<code>int</code> ASCII code	Return the ASCII code for a <code>character</code>
<code>ow_first</code>		<code>array</code> or <code>int</code> 0	Initiate a OneWire bus scan and return the address of the first device found
<code>ow_next</code>		<code>array</code> or <code>int</code> 0	Return the address of the next OneWire device found (after an <code>ow_first</code> )
<code>ow_read</code>		<code>int</code> value or 0	Read a byte from the OneWire bus
<code>ow_read_temp</code>	<code>array</code> device_id	<code>float</code> degrees Celsius	Read a temperature from a device on the OneWire bus
<code>ow_reset</code>			Reset the OneWire bus
<code>ow_write</code>	<code>int</code> value		Write a byte to the OneWire bus
<code>phpinfo</code>		<code>string</code>	Return information about the system
<code>pin_configure</code>	<code>int</code> pin_index, <code>int</code> pin_type, <code>int</code> counter_type		Configure an IO pin as a digital input, output, or analog input
<code>pin_get</code>	<code>int</code> pin_index, <code>int</code> pin_type	<code>int</code> value	Return the value of an IO pin
<code>pin_set</code>	<code>int</code> pin_index, <code>int</code> value		Set a digital output to <code>value</code> 1 or 0
<code>ping</code>	<code>string</code> host	<code>array</code>	Send an ICMP ping and place the result in an <code>array</code>
<code>power</code>	<code>number</code> base, <code>number</code> exp	<code>number</code> base <sup>exp</sup>	Return <code>base</code> raised to the power of <code>exp</code>
<code>print</code>	<code>string</code> data		Print <code>data</code> to the current output stream such as a web page or terminal
<code>print_r</code>	<code>array</code>		Dump the contents of an <code>array</code> to the current output
<code>printf</code>	<code>string</code> format, <code>mixed</code> values ...		Print a formatted <code>string</code> to standard output
<code>process_kill</code>	<code>int</code> pid		Send a kill request to a process
<code>process_list</code>		<code>array</code>	Return an <code>array</code> of the currently running scripts
<code>rand</code>	<code>int</code> min, <code>int</code> max	<code>int</code>	Return a random <code>integer</code> between <code>min</code> and <code>max</code>
<code>reboot</code>			Reboot the processor

FUNCTION NAME	PARAMETER(S)	RETURN	DESCRIPTION
rename	string source, string destination	int 0 or error code	Rename or move a file or directory from <a href="#">source</a> to <a href="#">destination</a>
reset			Reset the processor
rmdir	string pathname, int delete_contents	int 0=OK	Remove a directory, with optional deletion of contents
session_destroy			Clear the current session's data
session_is_new		int	Check if a session was just initiated
session_start			Initiate a new session and send the cookie data for it
set_search_path	string pathname		Set the search path for the telnet client
setethpower	int state		Enable or disable the ethernet controller
setpriority	int priority		Set the <a href="#">priority</a> of the current script
settime	int timestamp, int calibration		Set the system time from a Linux <a href="#">Timestamp</a> , with optional <a href="#">calibration</a>
setusbpower	int state		Enable or disable USB power
sha1	string input	string 40 characters	Calculate the SHA1 hash of a <a href="#">string</a>
sin	number radian_angle	float sine	Return sine of a <a href="#">radian_angle</a>
sizeof	array	int number of elements	Return the number of elements in an <a href="#">array</a>
sleep	int ms		Sleep for specified milliseconds
spi_clearcs			Clear the CS output of the <a href="#">SPI</a> bus
spi_read		int byte	Read a byte from the <a href="#">SPI</a> bus
spi_setcs			Set the CS output of the <a href="#">SPI</a> bus
spi_write	int byte		Write a <a href="#">byte</a> to the <a href="#">SPI</a> bus
sprintf	string format, mixed values ...	string formatted	Return a formatted <a href="#">string</a>
sqr	number number	number squared	Return the square of a <a href="#">number</a>
sqrt	number number	number square root	Return the square root of a <a href="#">number</a>
stats		array	Return system statistics
strftime	string format, int timestamp	string formatted	Format a Linux <a href="#">Timestamp</a> using a <a href="#">format string</a>
strlen	string input	int length	Return the length of a <a href="#">string</a>
strpos	string haystack, string needle	int position or -1	Return the position of the first occurrence of a <a href="#">needle</a> in a <a href="#">haystack</a>
strrpos	string haystack, string needle	int position or -1	Return the position of the last occurrence of a <a href="#">needle</a> in a <a href="#">haystack</a>

FUNCTION NAME	PARAMETER(S)	RETURN	DESCRIPTION
<a href="#">strtolower</a>	<a href="#">string</a> input	<a href="#">string</a> lowercase	Return the lowercase version of a <a href="#">string</a>
<a href="#">strtoupper</a>	<a href="#">string</a> input	<a href="#">string</a> UPPERCASE	Return the UPPERCASE version of a <a href="#">string</a>
<a href="#">strval</a>	<a href="#">mixed</a> value	<a href="#">string</a>	Return the <a href="#">string</a> equivalent of a <a href="#">number</a>
<a href="#">substr</a>	<a href="#">string</a> input, <a href="#">int</a> start, <a href="#">int</a> length	<a href="#">string</a> substring	Return part of a <a href="#">string</a>
<a href="#">tar_finish</a>	number handle		Add the ending header to a TAR file
<a href="#">tar_put</a>	<a href="#">int</a> handle, <a href="#">string</a> src_pathname, <a href="#">string</a> tar_pathname	<a href="#">int</a> 1=OK	Add a file to an open file in TAR format
<a href="#">time</a>		<a href="#">int</a> seconds	Return the current system timestamp
<a href="#">timefromfat</a>	<a href="#">int</a> filetype	<a href="#">int</a> seconds	Convert a FAT <a href="#">filetime</a> to a Linux Timestamp
<a href="#">ucfirst</a>	<a href="#">string</a> input	<a href="#">string</a> Lowercase	Convert a <a href="#">string</a> to Lowercase except for the first character
<a href="#">unlink</a>	string filename	Number (0=OK)	Remove a file (delete it)
<a href="#">untar</a>	string filename, number verbose		Expand a file into the current folder, optionally verbose
<a href="#">uptime</a>		<a href="#">int</a> ms	Return the uptime in milliseconds
<a href="#">usbinfo</a>		array	Get information about the USB state

From:

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

Permanent link:

[http://wattmon.com/dokuwiki/uphp/functions/uphp\\_function\\_reference?rev=1495907315](http://wattmon.com/dokuwiki/uphp/functions/uphp_function_reference?rev=1495907315)

Last update: 2021/09/13 05:56

