EtherRain Function Reference
For
HomeSeer 2

The EtherRain Sprinkler Control Interface for HomeSeer2 provides both a HomeSeer2 compatible web-page interface and event functions offering easy systematic setup and control of sprinkler irrigation using HomeSeer2 and EtherRain sprinkler valve controllers. The Interface provides support for HomeSeer's flexible events allowing complete irrigation cycles to be run by defining only a single event.

The web interface consists of a set of html pages which utilize JavaScript and the ASP scripting language that make it easy to set up an irrigation control environment using EtherRain controllers. The html pages call functions that are implemented within a single VB script called EtherRain.vb.

Many of the functions contained in EtherRain.vb are accessible for use in HomeSeer2 events. For example, an irrigation program schedule is commonly built by calling the most frequently used function: "runNamedCycle." See the accompanying users guide for information on how to set up irrigation events.

This document provides a list of the available functions as well as information and parameters used when calling those functions.

The following functions are available for use in HomeSeer2 events

> **runNamedCycle**
>
> **runSeqCycles**
>
> **runCycle**
>
> **stopCycle**
>
> **stopAllCycles**
>
> **disableSystem**
>
> **enableSystem**
>
> **suspendSystem**

These functions are described below.

How to Call a Function in a HomeSeer2 Event

EtherRain interface functions can be called in a HomeSeer2 event by setting up an Action in an Event Trigger. When the event triggers, the action is performed.

The process for creating an event, defining a trigger, and adding an action is described in these steps:

1. From the HS2 Events screen, click "Add Event"
2. The Event Properties screen will be displayed with the "Name" tab highlighted
3. Enter Name "Main Irrigation Cycle" and uncheck all of the option checkboxes.
4. Click the "Trigger" tab
5. Verify the current trigger type of "Absolute Time"
6. Check the days you wish to have the Event trigger.
7. Provide an Event Trigger time.
8. Click "Add Action" and from the list, click "Run Script"
9. Click "Switch to Advanced View" (though it may already be in advanced view)
10. From the field labeled "Existing Script," click the dropdown arrow and select "EtherRain.vb" (If the script does not appear then it was not copied to the correct folder.)
11. In the field labeled "Optional Parameters" enter the following.
    Important: Pay close attention to the formatting characters.

        (Function Call with parameters)

    The function call is always enclosed with parenthesis as shown
    The function call syntax is provided for each function in the function description below
    The exact syntax must be followed for the command to run.

12. Click the update button.
13. On the ensuing screen, click the "save" button
14. Locate the event on the event screen under the "Time Triggered" category.
15. You may wish to create a separate event group for EtherRain, if you do, use the group "EtherRain Events" so your scheduled events will be grouped with the monitoring event.

Accessible Functions

The following pages provide information necessary for using available functions within HomeSeer2 events.

Function: runNamedCycle

This action sends an irrigation command to the named controller based on the named cycle. The named cycle is created in the EtherRain interface pages. This is the most commonly used EtherRain control function. The function first checks the system parameters to determine whether a suspension is in effect before sending the command.   If rain is detected by the local sensor the command will not run and a message will be logged.

Parameters: Controller Name; Cycle Name

Function Call Syntax: ("runNamedCycle","EtherRainName; CycleName")

Where EtherRainName is the name of the controller which will receive the command, and CycleName is the name of the cycle you wish to use. The controller name must be defined in the virtual controller list and the cycle name must be defined on the cycle builder page.

Syntax Checks:

    Be careful to enclose the two parameter strings with the double quotes characters.
    Be careful to use both a beginning parenthesis and an end parenthesis.
    Be careful to separate the controller name from the cycle name with a semi-colon.
    Be careful to separate the "runNamedCycle" method name from the parameter string with a comma.

Function: runSeqCycles

This action runs multiple cycles sequentially through a single event. In installations that have multiple EtherRain controllers, this command allows 16, 24, and 32 zone cycles to be run sequentially through a single event.   This action can also be used for setting up cycle-and-soak type cycles in installations where it is desirable to minimize runoff. With this function, only one event needs to be set up in order to schedule sequential cycles through multiple controllers or multiple cycles through a single controller. The function automatically calculates cycle time, taking into account setbacks, if the setback is applied, so that the cycle runs continuously with a maximum gap between cycles of 1 minute.   This function requires the use of Named Cycles.

In order to use this function a list of controller names and cycle names must be specified in the parameter string. A controller name can be repeated, as can a cycle name. The function first checks the system parameters to determine whether a suspension is in effect before sending the commands.

Parameters: A list of Controller Names and Cycle Names

`Function Call Syntax:`
("runSeqCycles","EtherRain1; CycleName1; EtherRain2; CycleName2; EtherRain3; CycleName3")

Where EtherRainN is the name of the controller which will receive the cycle command, and CycleNameN is the name of the cycle from your Named Cycle list that you wish to use. The controller name must be defined in the virtual controller list and the cycle name must be defined on the cycle builder page. The cycle name must follow the controller name in the list. The list can contain up to 8 controller/cycle pairs (16 elements total).

Syntax Checks:

> Be careful to enclose the two parameter strings with the double quotes characters.
> Be careful to use both a beginning parenthesis and an end parenthesis.
> Be careful to separate the list elements (controller names and cycle names) with a semi-colon.
> Be careful to separate the "runSeqCycles" method name from the parameter string with a comma.

Function: runCycle

This action sends an irrigation command to a named EtherRain Valve Controller. A parameter string containing 10 parameters each separated by a semicolon must be provided.   A virtual controller does not need to be set up to use this command.

Parameters: The parameters are:
  Controller name,
  Controller password,
  8 valve "on" times.

All ten parameters are required. If ten parameters are not found in the parameter string, the event terminates with an error written to the log. The parameters m1 through m8 must be whole numbers from 0 to 99. These numbers represent valve "on" times in minutes. For example, if the m1 parameter was 25, then valve 1 would open for 25 minutes. To skip a valve, enter 0 for its valve on time. If rain is detected command will not run and a message will be logged.

Function Call Syntax:

("runCycle"," EtherRainName; password; m1; m2; m3; m4; m5; m6; m7; m8 ")

Syntax Checks:

  Be careful to enclose the two parameter strings with the double quotes characters.
  Be careful to use both a beginning parenthesis and an end parenthesis.
  Be careful to separate the 10 parameters with a semi-colon character.
  Be careful to separate the "runCycle" method name from the parameter string with a comma.

Function: stopCycle

This action sends a stop/reset command to the named controller. A virtual controller does not need to be set up.

Parameters: Controller Name; Controller password

Function Call Syntax: ("stopCycle","EtherRainName; Password")

 Syntax Checks:

  Be careful to enclose the two parameter strings with the double quotes characters.
  Be careful to use both a beginning parenthesis and an end parenthesis.
  Be careful to separate the controller name from the cycle name with a semi-colon.
  Be careful to separate the "stopCycle" method name from the parameter string with a comma.

Function: stopAllCycles

This action sends a stop/reset command to each controller in the virtual controllers list. Useful when called from a HomeSeer weather event.

Parameters: none

Function Call Syntax: ("stopAllCycles")

 Syntax Checks:

Be careful to enclose the parameter string with the double quotes characters.
Be careful to use both a beginning parenthesis and an end parenthesis.

Function: disableSystem

This action sets the global HomeSeer control switch to disable irrigation. No irrigation commands will run through the EtherRain control scripts until the global control switch is re-enabled via manual event or control page button click.

Parameters: none

Function Call Syntax: ("disableSystem")

 Syntax Checks:

Be careful to enclose the parameter string with the double quotes characters.
Be careful to use both a beginning parenthesis and an end parenthesis.

Function: enableSystem

This action sets the global HomeSeer control switch to enable irrigation.

Parameters: none

Function Call Syntax: ("enableSystem")

 Syntax Checks:

Be careful to enclose the parameter string with the double quotes characters.
Be careful to use both a beginning parenthesis and an end parenthesis.

Function: suspendSystem

This action sets the global HomeSeer control switch to suspend irrigation for a specified number of days from 1 to 9. The system is disabled for the number of days specified, and then automatically re-enabled when the suspension period is over. No irrigation commands will run through the EtherRain control scripts during the suspension period unless a the global HomeSeer control switch is re-enabled via the "enableSystem" command or a button click on the control page.

Parameters: number of days to suspend starting from the current date, max of 9 days.

Function Call Syntax: ("suspendSystem","days")

 Syntax Checks:

        Be careful to enclose the two parameter strings with the double quotes characters.
        Be careful to use both a beginning parenthesis and an end parenthesis.
        Be careful to separate the "suspendSystem" method name from the "days" string with a comma.
        Be careful to enter a single digit number for the "days" parameter

Rev 1, August 2011
Rev 1.1, June 2012 (Adds: runSeqCycles, runCycle and runNamedCycle log rain status)