**New Scheduler – logic flow**

New scheduler replaces the old/original sprinklers\_pi scheduler with a lot simpler, cleaner logic.

Original sprinklers\_pi scheduler uses intermediate list of “events”. “Events” are generated out of the schedule, and are executed by an execution engine. Sprinklers\_pi “events” are actually two-layer – they may include references to schedules, as well as direct zone start/stop commands.

New scheduler does not use “events” and has a lot simpler logic. Key principles:

* New scheduler allows execution of one schedule at a time (this constrain may be lifted in the future by running multiple instances of scheduler).
  + If two or more schedules overlap, only one of them will be executing.
* New scheduler runs one zone at a time (this constrain may be lifted in the future by running multiple instances of scheduler).
* New scheduler can co-exist with manual zones control and other scheduler instances (or other schedulers) – it starts/stops zones according to the schedule but does not interfere with other zones start/stop operations.

**Design**

New scheduler class has few global/state variables:

* Currently running schedule (or -1 if no schedule is running)
* Currently running zone (or -1 if no zone is running)
* Timestamp of the currently active zone start in millis()
* Timestamp of the currently active schedule start in millis()
* Currently running zone set time, in minutes

Schedule can be one of the configured schedules in EEPROM, or an ad-hoc, “quick schedule”.

Timestamps are stored and processed in millis() to avoid impact of the time changes (e.g. due to time sync) while schedules are running.