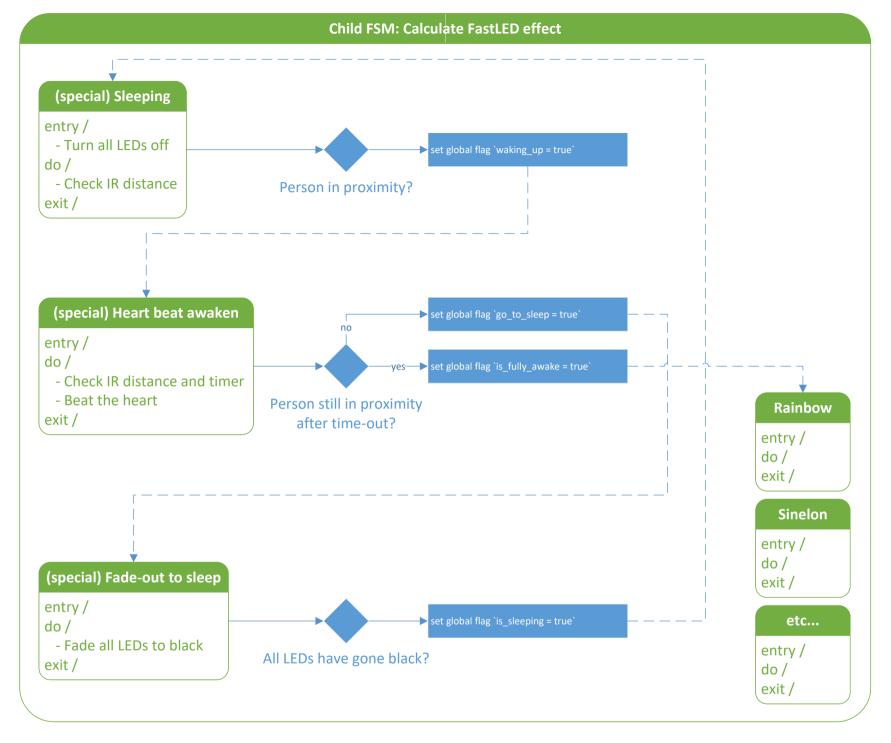
ect special effect `Sleeping` et global flag `is_sleeping = true is sleeping? **START** Show FastLED effect waking up? entry / do / - Check incoming serial commands - Check global flags is_fully_awake? `is sleeping` `waking up` ect special effect `Fade-out to sleep 'is fully awake' 'go to sleep' go to sleep? - Check IR distance and timer - Check button press - Check mode set global flag `go_to_sleep = true` - Invoke FSM `Calculate FastLED effect` - Show effect (call `FastLed.delay()`) is fully awake exit / & person out of proximity for more than 10 minutes? Show modes menu entry / - Double-flash LEDs state__Rainbow, state__Sinelon, etc... - Show current mode - Check button press **Button long pressed** Button long pressed? - Check time-out | timed-out? exit / - Double-flash LEDs Button short pressed? Overall notes: is fully awake Making use of global flags might not be & button short pressed needed. Instead, one could perhaps & (mode== 1 | mode==2)? directly poll the current state of the FSM and decide on that. Likewise, one could EVERY_N_SEC() { next_effect(); perhaps transition to another state from within the current state. CONFIRMED. is fully awake

& mode==1?



Notes: 'Heart beat awaken' effect

Start beating LEDs in red, every beat brighter than the previous. When user is still in proximity after 20 secs then fully awake, else go to sleep again.

Menu will indicate mode by lighting up 1 of the 4 corners of the mirror: L mode 1: show effect from preset list, auto-advance

J mode 2: show effect from preset list, no auto-advance

mode 3: show IR distance demo (hue = IR distance)

r mode 4: ...

Notes: Modes menu