|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Current State | Event | | | | Next State |
| MIN\_BPM | p\_rateModulation == e\_off | | | | --- |
| p\_rateModulation == e\_on | k\_currentActivity < p\_activityThreshold | | | --- |
| k\_currentActivity ≥ p\_activityThreshold | After(k\_hysWaitDelay) | | UP\_BPM |
| NOT(After(k\_hysWaitDelay)) | | --- |
| UP\_BPM | p\_rateModulation == e\_off | | | | MIN\_BPM |
| p\_rateModulation == e\_on | k\_current activity < p\_activityThreshold | After(k\_rateChangeDelay) | | SAME\_BPM |
| NOT(After(k\_rateChangeDelay)) | | --- |
| k\_currentActivity ≥ p\_activityThreshold | After(k\_rateChangeDelay) | k\_BPM + p\_modulationSensitivity ≤ p\_upperRateLimit | UP\_BPM |
| k\_BPM + p\_modulationSensitivty > p\_upperRateLimit | MAX\_BPM |
| NOT(After(k\_rateChangeDelay)) | | --- |
| SAME\_BPM | p\_rateModulation == e\_off | | | | MIN\_BPM |
| p\_rateModulation == e\_on | k\_currentActivity < p\_activityThreshold | After(k\_hysWaitDelay) | | DOWN\_BPM |
| NOT(After(k\_hysWaitDelay)) | | --- |
| k\_currentActivity ≥ p\_activityThreshold | After(k\_hysWaitDelay) | | UP\_BPM |
| NOT(After(k\_hysWaitDelay)) | | --- |
| DOWN\_BPM | p\_rateModulation == e\_off | | | | MIN\_BPM |
| p\_rateModulation == e\_on | k\_currentActivity < p\_activityThreshold | After(k\_rateChangeDelay) | k\_BPM - p\_modulationSensitivity ≥ p\_lowerRateLimit | DOWN\_BPM |
| k\_currentActivity ≥ p\_activityThreshold | After(k\_rateChangeDelay) | k\_BPM - p\_modulationSensitivity < p\_lowerRateLimit | MIN\_BPM |
| NOT(After(k\_rateChangeDelay)) | | --- |
| MAX\_BPM | p\_rateModulation == e\_off | | | | MIN\_BPM |
| p\_rateModulation == e\_on | k\_currentActivity < p\_activityThreshold | After(k\_hysWaitDelay) | | DOWN\_BPM |
| NOT(After(k\_hysWaitDelay) | | --- |
| k\_currentActivity ≥ p\_activityThreshold | | | --- |

# MIN\_BPM

## ENTRY:

%Min bpm is lower rate limit

k\_bpm = p\_lowerRateLimit;

# up\_BPM

## ENTRY:

%New bpm is current bpm plus modulation rate

k\_bpm = k\_bpm + p\_modulationSensitivity;

# SAME\_BPM

## ENTRY:

%No change in bpm

# DOWN\_BPM

## ENTRY:

%New bpm is current bpm minus modulation rate

k\_bpm = k\_bpm – p\_modulationSensitivity;

# MAX\_BPM

## ENTRY:

%Max bpm is upper rate limit

k\_bpm = p\_upperRateLimit;

p\_rateModulation {e\_off, e\_on} – is rate modulation enabled or disabled

k\_currentActivity {double} – RMS average of the accelerometer readings

p\_activityThreshold {double} – programmed threshold to initiate rate modulation

k\_hysWaitDelay {uint16} – delay to account for hysteresis effects

k\_rateChangeDelay {uint16} – delay specifying how often to change bpm

p\_modulationSensitivity {uint8} – specifies how much the rate should change per event

p\_lowerRateLimit {uint8} – lower end bpm rate

p\_upperRateLimit {uint8} – upper end bpm rate