f_x calcFireTarget(f: SuperimposedFrame): FireTarget

f_x superimpose(rgb: RgbFrame, d: Frame, t: Frame): SuperimposedFrame

▶ forall i: nat, j: nat @ (result.contents)[i,j].rgb==rgb[i,j]/\(result.contents)[i,j].d==d[i,j]/\(result.contents)[i,j].t==t[i,j]

FireDetect

- ® MLX90640
- ® RealSenseD435i
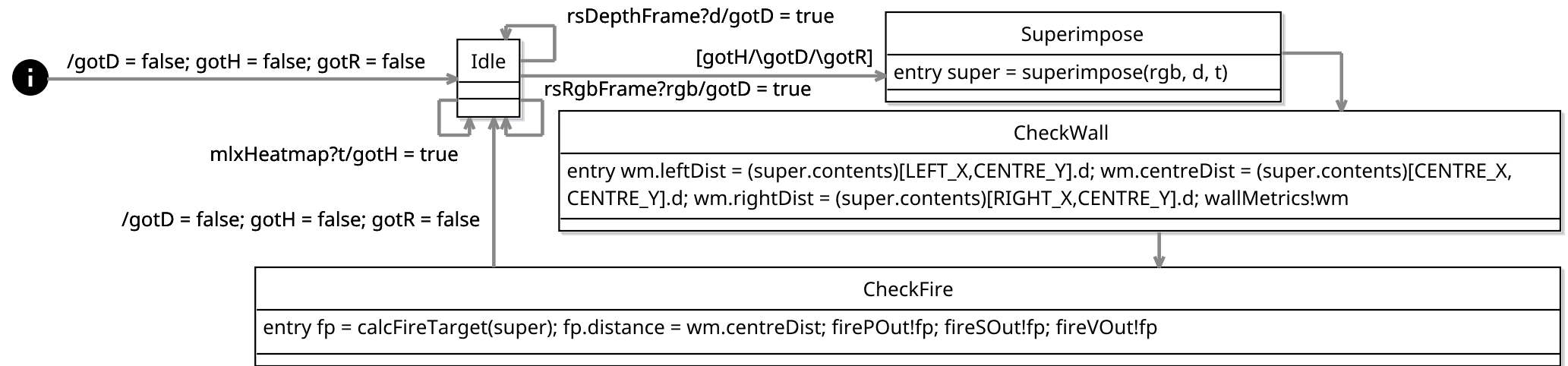- ✗ fp: FireTarget, fs: FireStatus, wm: WallMetrics, checkDone: boolean
- ✗ gotD: boolean, gotH: boolean, gotR: boolean
- ✗ rgb: RgbFrame, d: Frame, t: Frame, super: SuperimposedFrame
- π LEFT_X: nat, CENTRE_X: nat, RIGHT_X: nat, CENTRE_Y: nat
- ⓘ MLX90640Events
- ⓘ RealSenseD435iEvents

/gotD = false; gotH = false; gotR = false

Idle

rsDepthFrame?d/gotD = true

[gotH/\gotD/\gotR]

rsRgbFrame?rgb/gotD = true

mlxHeatmap?t/gotH = true

/gotD = false; gotH = false; gotR = false

Superimpose

entry super = superimpose(rgb, d, t)

CheckWall

entry wm.leftDist = (super.contents)[LEFT_X,CENTRE_Y].d; wm.centreDist = (super.contents)[CENTRE_X, CENTRE_Y].d; wm.rightDist = (super.contents)[RIGHT_X,CENTRE_Y].d; wallMetrics!wm

CheckFire

entry fp = calcFireTarget(super); fp.distance = wm.centreDist; firePOut!fp; fireSOut!fp; fireVOut!fp

wallMetrics: WallMetrics

firePOut: FireTarget   fireSOut: FireTarget   fireVOut: FireTarget   mlxHeatmap: Frame   rsRgbFrame: RgbFrame   rsDepthFrame: Frame