

```
float utilization = (float)busyTime / totalTimeMs * 100.0
```

printed = 0

i = 0 to fifoTail -1 && printed < producedCount

False

pieces[i].exitTime != 0

Print each piece's details Only for finished pieces

True

normalize all times relative to when the system started.

pieces[i].id

pieces[i].entryTime - systemStartTime) / 1000.0
pieces[i].exitTime - systemStartTime) / 1000.0
pieces[i].cycleTime / 1000

printed++

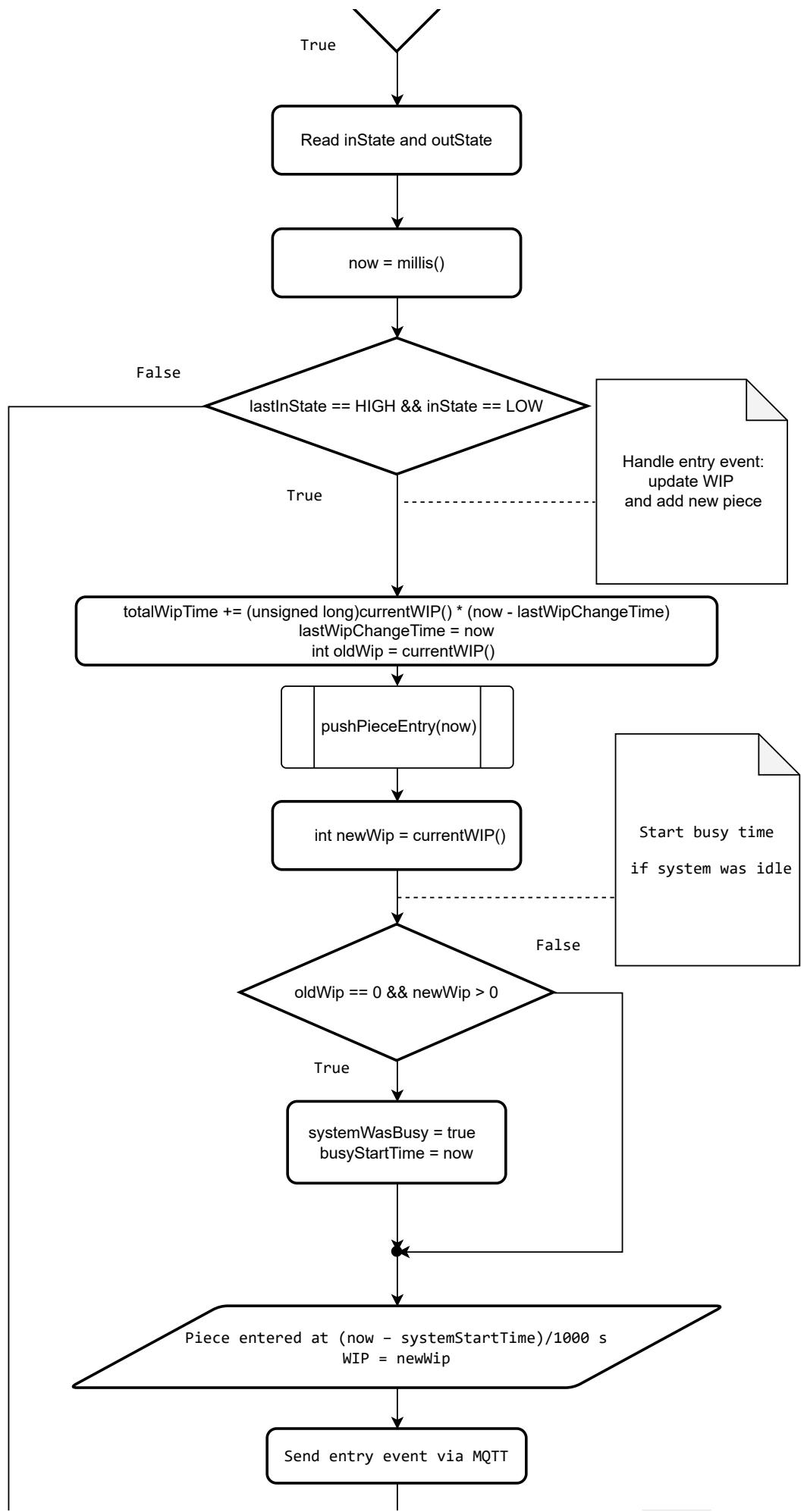
Build JSON summary to send via mqtt

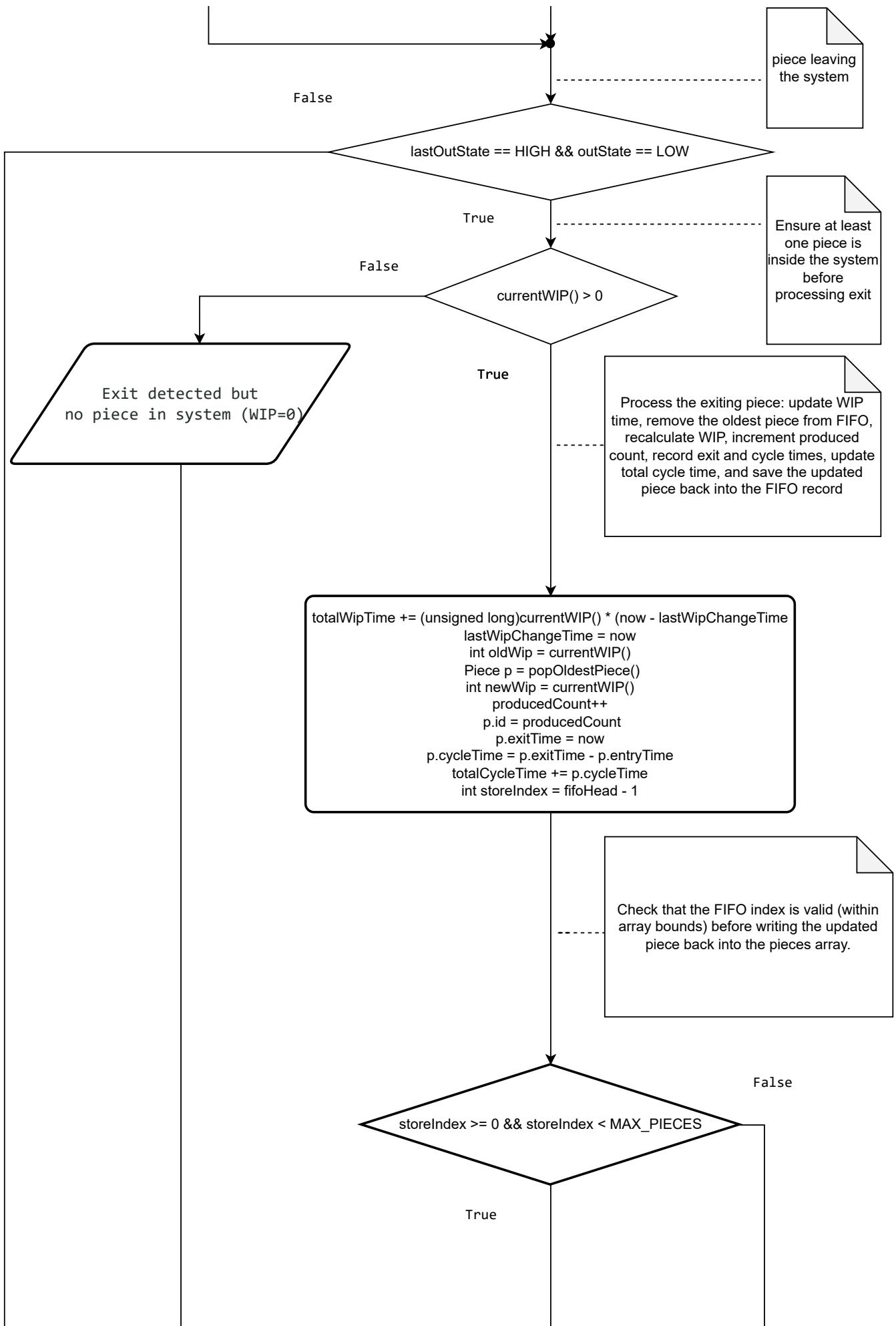
lastButtonState = currentButtonState

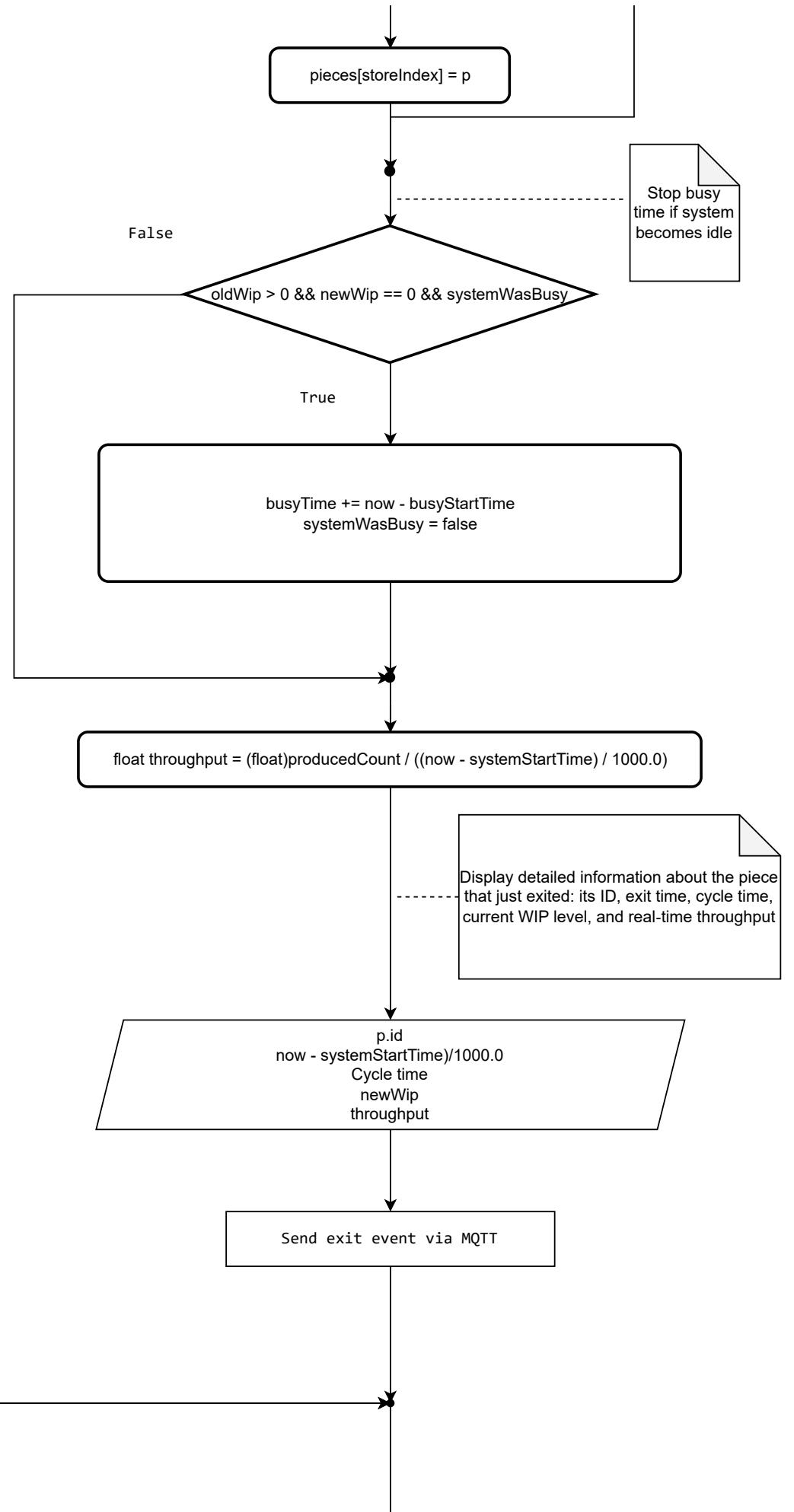
If system is running, check sensors

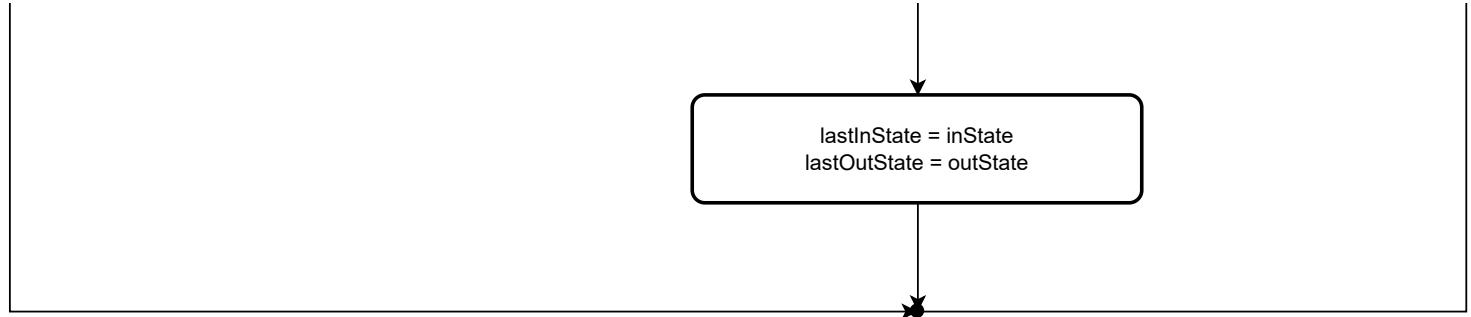
False

systemRunning









```
graph TD; A((lastInState = inState  
lastOutState = outState)) --> B([★])
```

lastInState = inState
lastOutState = outState