```
int[0, M] avail_processors = M;
urgent chan run[N];
broadcast chan synch, first_synch;
chan priority first_synch < run;
chan priority synch < run;

bool is_last_segment() {
    return seg_idx ==
        Tasks[id].k - 1;
}

bool job_ready() {
    return queue_len > 0;
}

bool processor_avail() {
    return avail_processors > 0;
}
```