

```

formula update_bw = ... // update bandwidth of all camera
module NetworkManager
[] (rm = rm_init) -> 1 : (rm' = rm_calc_bw); // initialization
[end] (rm = rm_end) -> 1 : (rm' = rm_end); // self-absorbing
[man_inter] (rm = rm_calc_bw) & (!end)-> 1 :
    (rm' = rm_alloc_bw) & (bw' = update_bw);
// bw_allocated: allocating bandwidth to camera and waiting,
// end if reached the maximum number of frames
[bw_allocated] (rm = rm_alloc_bw) & (frames < max_frames) &
    (!end) -> 1 : (rm' = rm_wait) & (frames' = frames + 1);
[bw_allocated] (rm = rm_alloc_bw) & (frames >= max_frames) &
    (!end) -> 1 : (end' = true) & (rm' = rm_end); // final
// last_cam_sent: check if recalculation is needed/requested
// or not and switch to corresponding state
[last_cam_sent] (rm = rm_wait) & (want_rm) & (!end) ->
    1 : (rm' = rm_calc_bw); // recalculation
[last_cam_sent] (rm = rm_wait) & (!want_rm) & (!end)->
    1 : (rm' = rm_alloc_bw); // no recalculation
endmodule

```