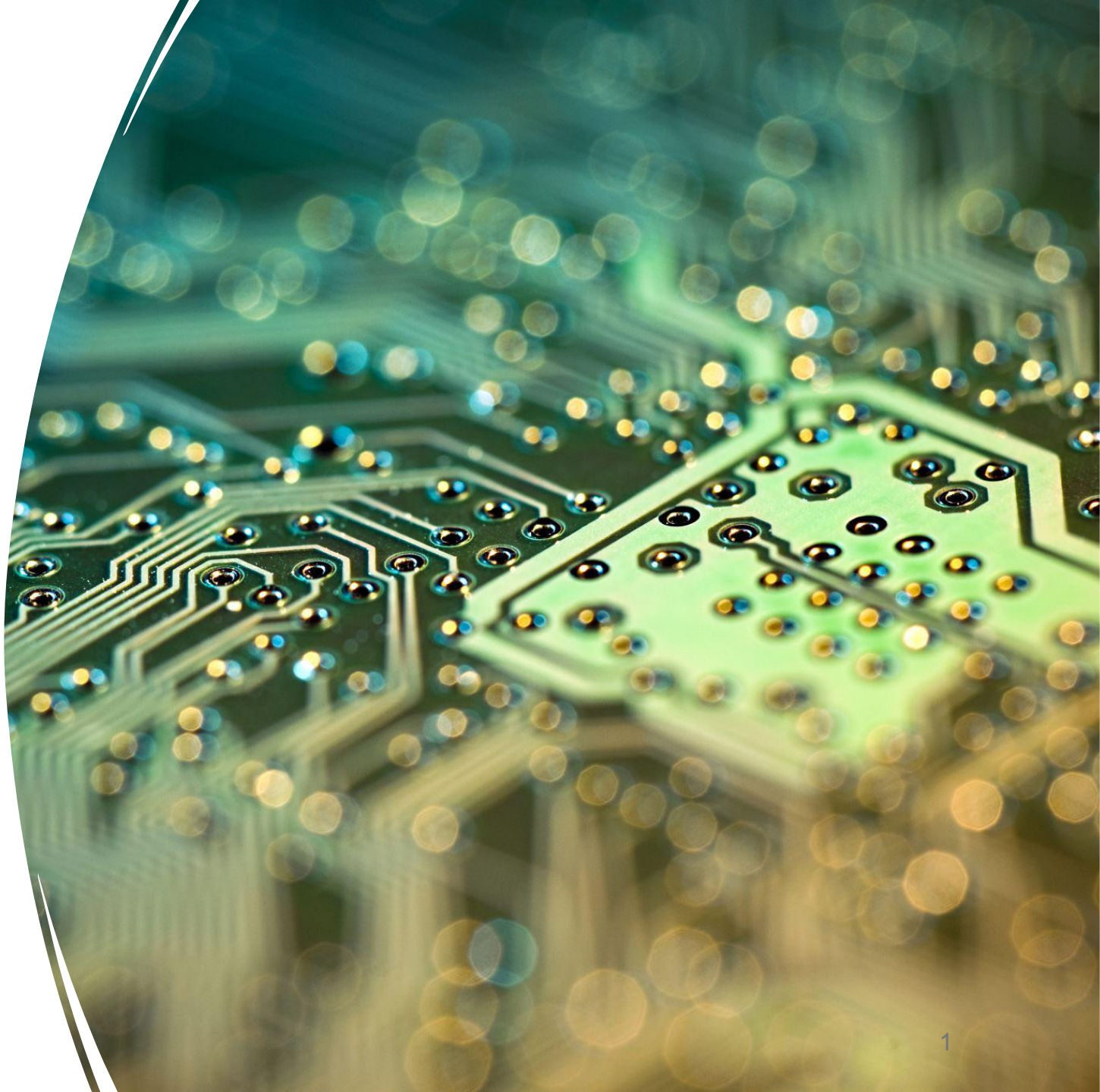
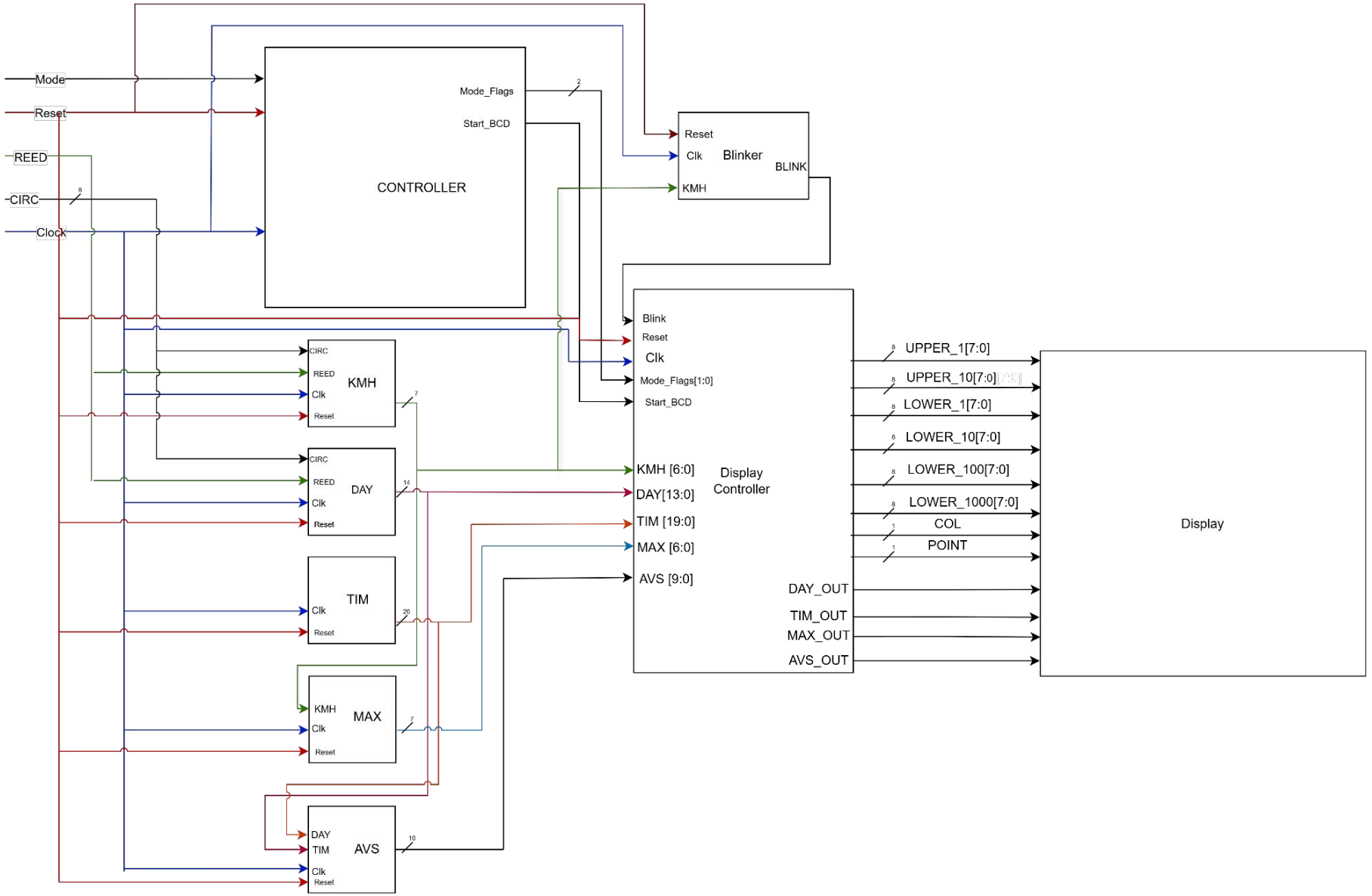
Project Integrated Circuit Design

Favio Acosta

Enes Öncel



System Overview



Controller

Inputs: CLK, MODE, RESET

Outputs: Mode_Flags[1:0], Start_BCD

Functionality:

- Holds Mode value to provide the display controller (though Mode_Flags)
- Controls blinking of mode indicators if speed > 65 km/h (Blinker module)
- Ensures that START_BCD timing is synchronized for display updates

Instantaneous Speed Module

Inputs: CLK, REED, CIRC[8:0], RESET

Operation:

- A counter starts at the first REED pulse and stops at the second, counting clock pulses (COUNT)
- CONSTANT = (CIRC {cm} * 2048 * 3600) / 100000
- AVS {km/h} = CONSTANT / COUNT

Resolution:

- 1 km/h
- There's no need to address decimal point

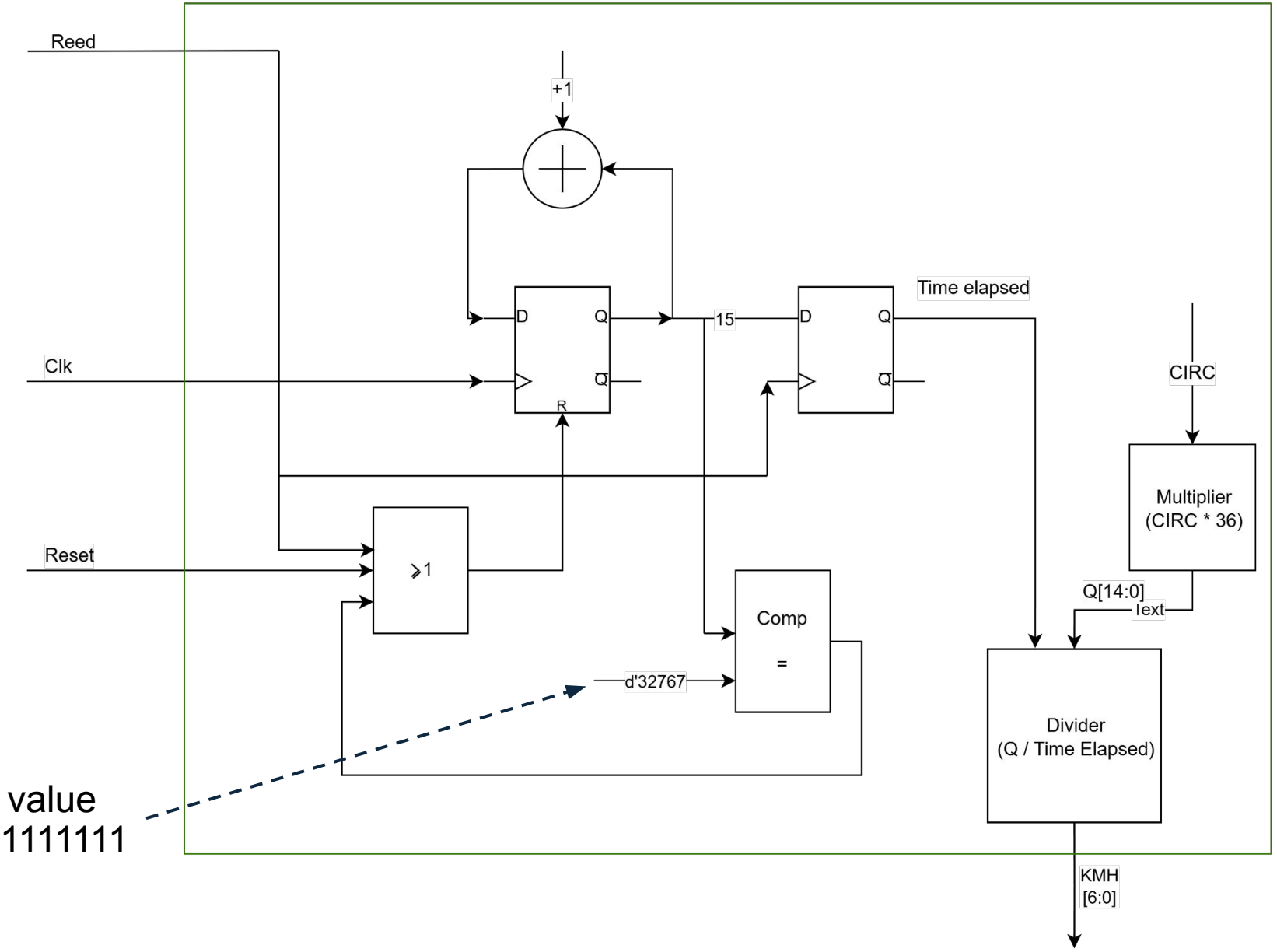
Output: KMH [6:0]

Example:

Constant = (220 (CIRC {cm}) * 2048 * 3600) / 100000
= 16220.16
(int) 16220

16220/1000
= (int) 16 km/h

Instantaneous Speed Module



Saturation value
b’111111111111111

Blinker Module

Purpose: Toggles the signals DAY, AVS, TIM or MAX if the KMH is greater than 65 km/h

Inputs: CLK, RESET, KMH[6:0]

Operation:

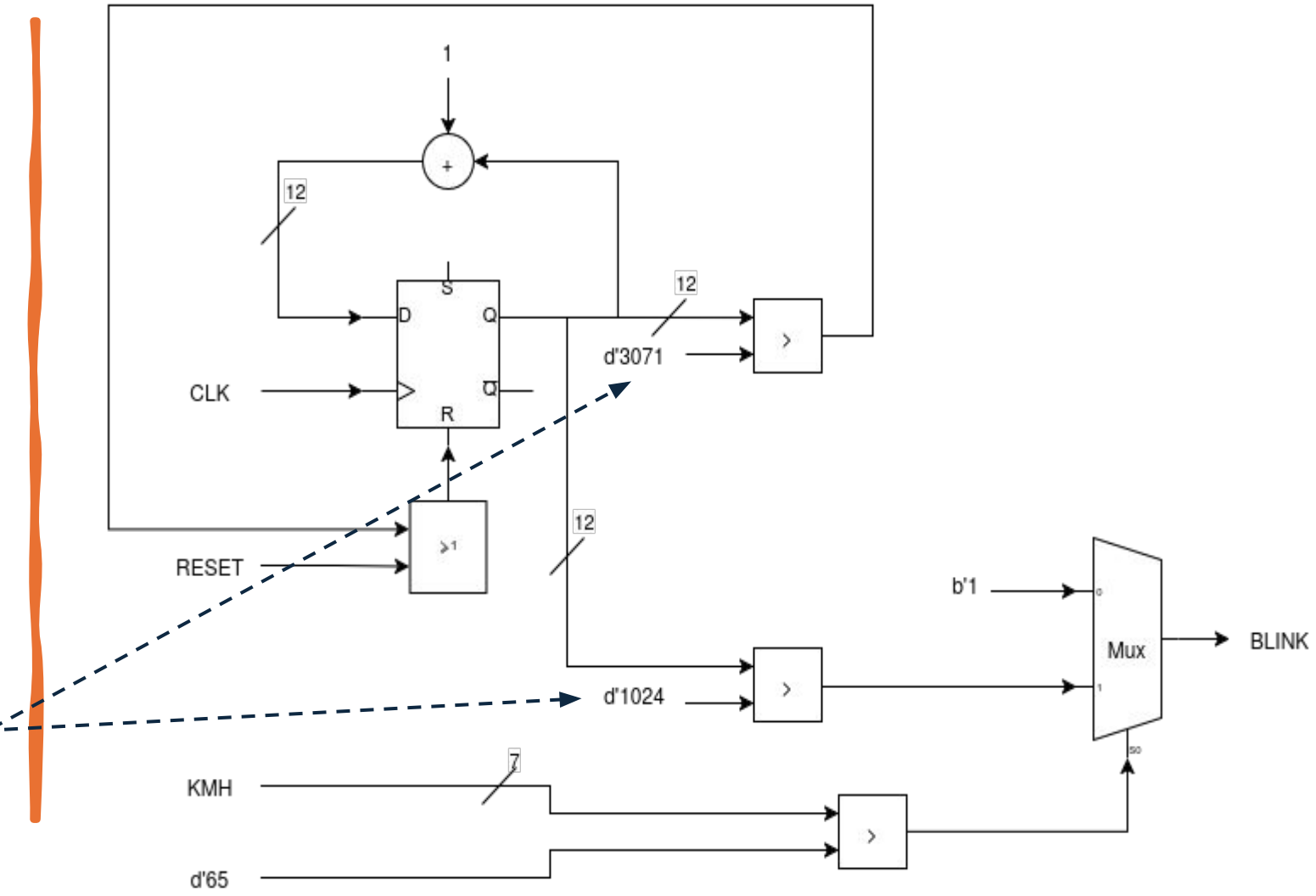
- Uses a comparator module between the input KMH {km/h} and the limit speed of 65 km/h
- If KMH is lower, then the output is ‘1’
- If KMH is greater, then the output blinks (0.5 seconds ‘0’, 1 second ‘1’)

Output: BLINK

- Used then in the display controller to blink the corresponding signals.

Blinker Module

0.5 Seconds → 1024 Pulses
1.5 Seconds → 3072 Pulses



Trip Distance Module

Purpose: Tracks trip distance by accumulating wheel rotations.

Inputs: CLK, REED, CIRC[8:0], RESET

Operation:

- DISTANCE {cm} += CIRC (with every REED)
- Converts to km:
  - DAY {km*10} = DISTANCE / 10000

Resolution:

- 0.1 km/h
- DAY is calculated as tens of km. For example:
  - DAY = 505
  - DISTANCE = 50.5 {km}

Output: DAY [13:0]

- Converted from tens of km to km in display controller

Trip Time Module

Purpose: Tracks trip time when bicycle starts moving

Inputs: CLK, RESET

Operation:

- Requires 2048 pulses to achieve 1 {s}
  - TIM {s} += 1 every 2048 ticks

Output: TIM [19:0] in number of seconds

- Converted to MM:SS or HH:MM in Display controller

Maximum Speed Module

Purpose: Tracks maximum speed reached during trip

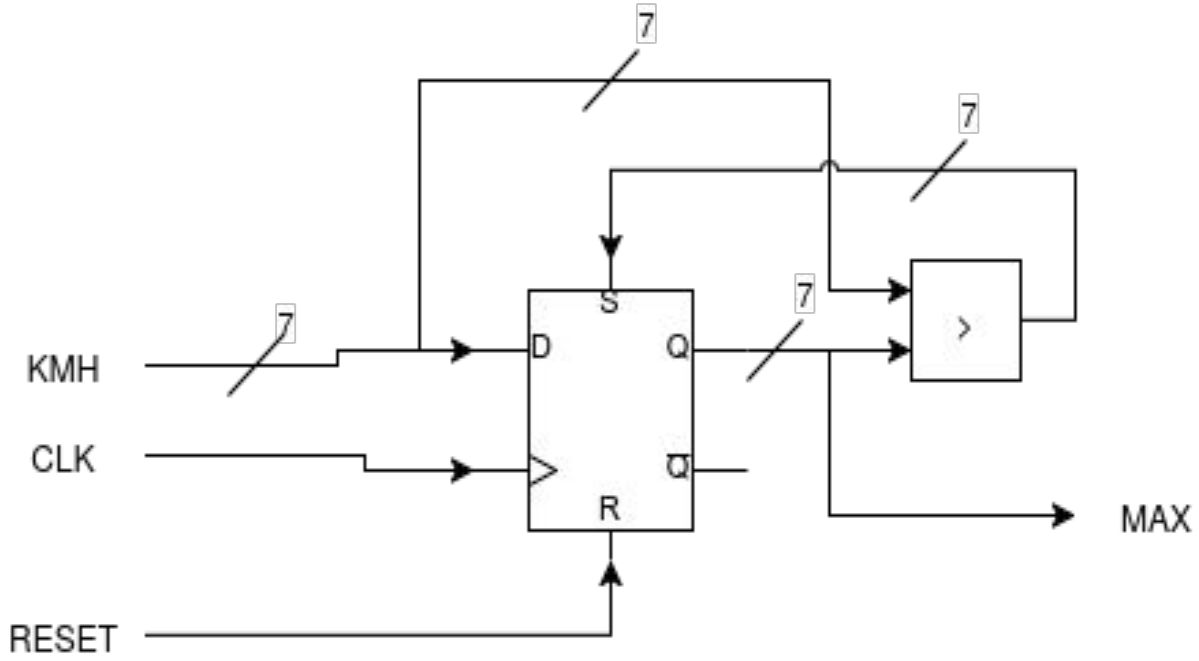
Inputs: CLK, KMH[6:0], RESET

Operation:

- Compares previous value with current value each CLK cycle
- Takes the greater value
- Unit is {km/h}

Output: MAX [6:0]

Maximum Speed Module



Average Speed Module

Purpose: Calculates average speed from distance module and trip time module

Inputs: CLK, DAY[13:0], TIM[19:0], RESET

Operation:

- AVS {km*10/h} =

  (DAY {km*10} / (TIM {s}) * 3600

Resolution:

- 0.1 km/h
- AVS is calculated as tens of km/h. For example:
  - AVS = 505
  - SPEED = 50.5 {km/h}

Output: AVS [9:0]

- Converted from tens of km/h to km/h in display controller

Display Controller (1/3)

Purpose: Converts selected result to displayable ASCII and manages LCD

Inputs: CLK, RESET, HOLD_DISP, START_BCD, Mode_Flags[1:0], KMH[6:0], BLINK, DAY[13:0], TIM[19:0], MAX[6:0], AVS[9:0]

Operation:

- Selects current module output based on Mode_Flags
  - Activates to ‘1’ the desired mode output (DAY, TIM, MAX, AVS), and set the others to ‘0’
- Converts binary → BCD → ASCII
- Controls POINT symbol:
  - POINT (decimal point) for AVS and DAY
    - First binary type is converted to BCD
    - POINT goes on the 2nd digit from the left, between the tens and units. For example:
      - DAY = d’505 → b’111111001 → BCD’0101 0000 (POINT on) 0101

Display Controller (2/3)

… Operation:

- Controls COL symbol:
  - COL (column time separator) for TIM
    - Just when TIM is activated based on Mode_Flags.
    - Measures 1 second based on CLK and toggles COL output.
- TIM MM:SS or HH:MM calculation steps:
  1. Minutes & seconds
     - MINUTES = TIM ÷ 60
     - SECONDS = TIM mod 60
  2. Hours & minutes
     - HOURS = MINUTES ÷ 60
     - MINUTES_HOUR = MINUTES mod 60
  3. Choose format
     - If TIM < 3600 → MM:SS
     - Else → HH:MM
  4. BCD convertion → Map to display

Display Controller (3/3)

… Operation:

- Controls blinking of outputs DAY_OUT, TIM_OUT, MAX_OUT, AVS_OUT based on input BLINK
  - Given the 3 signals that are not active, based on Mode_Flags, an AND gate is established with the BLINK input, in order to make these outputs to toggle according to the requirements.

Outputs:

- UPPER10 [7:0], UPPER1 [7:0]
- LOWER1000 [7:0] … LOWER1 [7:0]
- COL, POINT
- DAY_OUT, TIM_OUT, MAX_OUT, AVS_OUT

Thanks