

Přílohy:

- sensor_driver.vhd
- tb_sensor_driver.vhd

Popis struktury:

Vyčítání dat ze senzoru je založeno na principu FSM. Probíhá v následujících stavech:

1) `idle` (prodleva mezi jednotlivými měřeními) 2) `trigger` vyslání trigger impulsu délky 100us 3) `tarry` prodleva mezi seběžnou hranou trigger impulsu a náběžnou hranou echo pulzu (vyčkávání na echo response) 4) `counting` čítání délky echo pulzu,, po ukončení čítání se proces opakuje.

Popis problému:

V procesu `sensor_get_data` po ukončení čítání (*state counting; podmínka if (echo_i = '0')*) je obsažené vyhodnocení délky echo responsu. Je zde přepočet na cm, ošetření rozsahu hodnot, a přiřazení výstupního vektoru na port.

Právě zde se vyskytuje problem. Jakoby některé řádky procesu proběhly, jiné ne, nebo se provedou až po dalším zavolání funkce (viz screenshot simulace1 > přiřazení z `s_distance` na výstup proběhne až při dalším zavolání >> na výstupu je vždy hodnota z předchozího měření). K ošetření podmínek také nedojde, na výstupu se vyskytují i čísla mimo rozsah (při přiřazení pak dojde k přeteční výstupního vektoru)

pozn.: Provizorně jsem zkusil problém vyřešit tím způsobem, že jsem přiřazení výstupního vektoru na výstupní port prováděl v každém hodinovém cyklu. Tím se přiřadil výstup o další hodnový tik později (viz screenshot simluace2). I tak ale nedojde k vzhodnocení podmínek ošetření stavů...

Možná je problém s použitím proměných/konstant typu integer. V literatuře jsem k tomuto žádnou připomínku nenašel, proto se obracím na Vás.

Výčet programu case pro state `counting` :

```
when counting =>
  -- wait for fall of echo pulse and count time
  if (echo_i = '0') then
    -- compute distance (in cm)
    s_distance <= (s_counter+1)/5800;
    -- range threatment
    if (s_distance > 255) then s_distance <= 255; end if; -- Max of range
    if (s_distance < 1) then s_distance <= 1; end if; -- Min of range
    -- mazbe TODO threatment of range of sensor itself (2cm-4meters)
    -- TODO threatments are not function
    s_counter <= 0; -- reset counter
    s_state <= idle; -- change state
    -- output assigment
    distance_o <= std_logic_vector(to_unsigned(s_distance, 8));
    -- too far obstacle if echo pulse is too long
  elsif (s_counter >= c_max_echo_time) then
    distance_o <= c_out_msg_too_far; -- set special event message
    s_state <= fault;
```

```
s_counter <= 0;    -- reset counter  
end if;
```

Odpověď mi bude klidně stačit písemně, případně pro domluvení online konzultace se domluvíme, časově se vrámci možností přizpůsobím. Kontakt na mě: mail: xvanek39@vutbr.cz; nebo volejte na 731937719

Děkuji mnohokrát za Vaší pomoc S pozdravem Vaněk Pavel