PLA3_4. Designing arithmetic circuits using VHDL hierarchical plan C2

We like to invent a binary adder and subtractor circuit to be used with 9-bit two's complement (2C) signed integers as shown in Fig. 1a. Adders are fundamental building blocks in digital design, they were produced ever since as MSI classic and standard chips and now they are integrated by the millions in FPGA or ASICs lattices. For instance, the 4-bit binary adder for radix-2 numbers 74HCT283 represented in Fig. 1b.

We like to build the prototype in a DE115 board using buttons, switches, LED and 7-segment displays as represented Fig. 2. This prototyping is step 6 in our VHDL design flow sequence and optional. Previously we must design our project as usual 1-2-3-4-5, adding this time step 5 on gate-level simulation of the technology view schematic and calculating the circuit's speed.

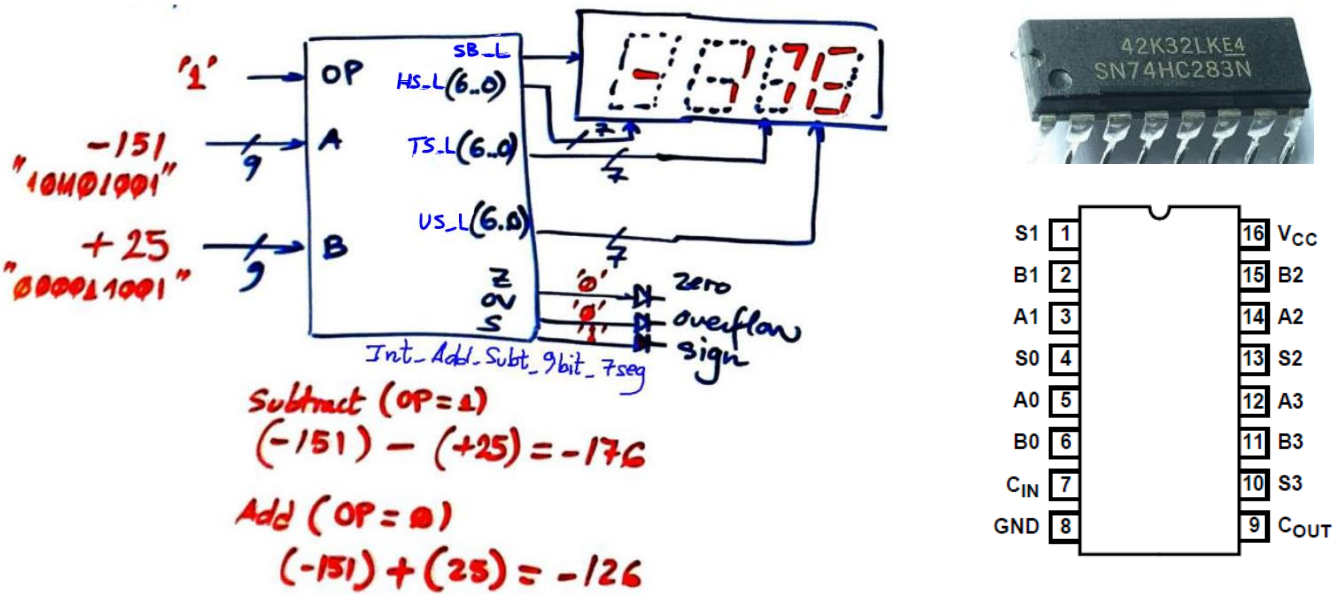

Fig. 1 Symbol for the *Int_Add_Subt_9bit_7seg* chip to design. Chip 74HCT283 is a classic 4-bit adder, and chip 74HCT181 is a classic 4-bit ALU capable of performing 32 logic and arithmetic operations.

Project location L:\CSD\P4\Int_Add_Subt_9bit_7seg*(project files)*

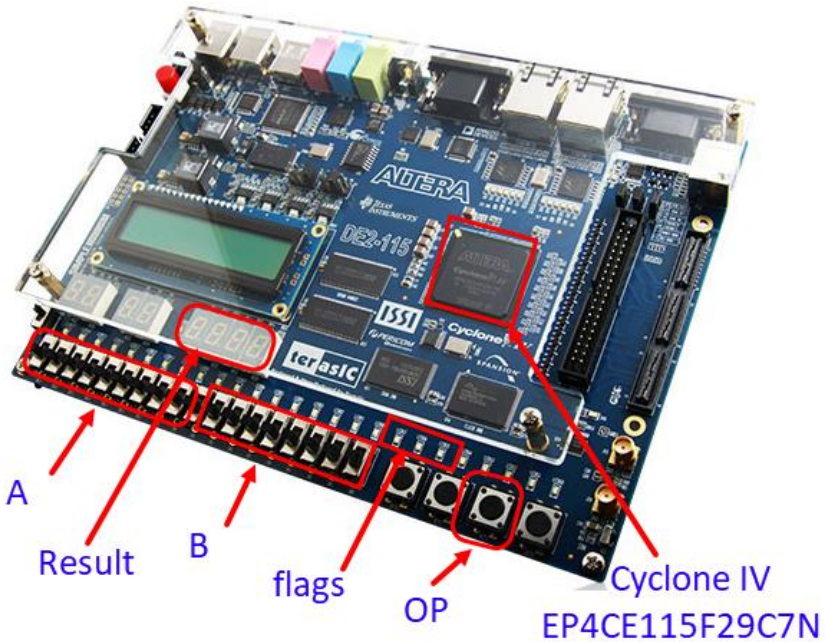

Fig. 2 Terasic -DE2-115 representing how switches and buttons are used to input data and LED and 7-seg display to show results.

Let us organise how to solve the project that in some way includes content from P1 to P4.

1. Specifications

a. Solve the following operations using our Fig. 1a circuit:
   - (+25) + (-51) ; (-256) – (-151) ; (-106) + (-156) ; (0) – (0) ; (+106) – (+156)

b. Write circuit's truth table using some examples.

c. Draw an example sketch of a timing diagram to be later at section 4 translated to a VHDL testbench file.

d. Explain how push-buttons and switches work in DE115 board to generate input signals

2. Planning

We will discuss the plan C2 for *Int_Add_Subt_9bit_7seg* represented in Fig. 3. In principle, it looks like that five chips are required: Chip1 is an arithmetic operator, Chip2 is a code converter, and Chip3, Chip4 and Chip5 are adapters with active-low outputs to drive four of the 7-segment displays in DE115 board.

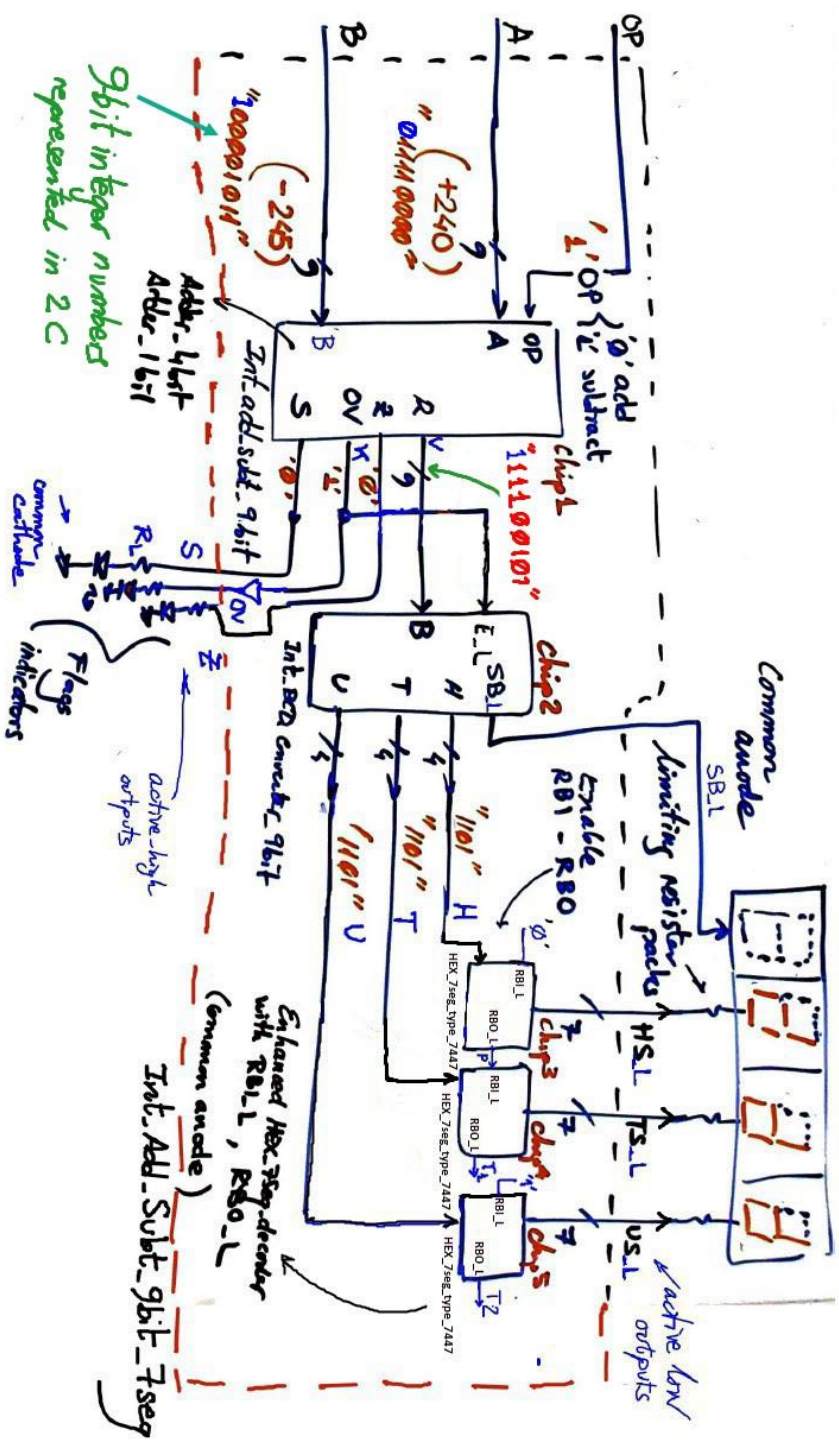

Fig. 3. Schematic plan C annotated ready for translation to VHDL.

a. Redraw circuit in Fig. 3 in your own style. Discuss why such circuit may work correctly. Apply (+25) + (-51); (-256) – (-151); (-106) + (-156); (0) – (0); (+106) – (+156), and explain signal values in every pin to show how the chip is processing numbers.

b. Redraw circuit in Fig. 4 in your own style. Discuss why such circuit may work correctly. Write circuit's truth table using some examples and a sketch an example of timing diagram.

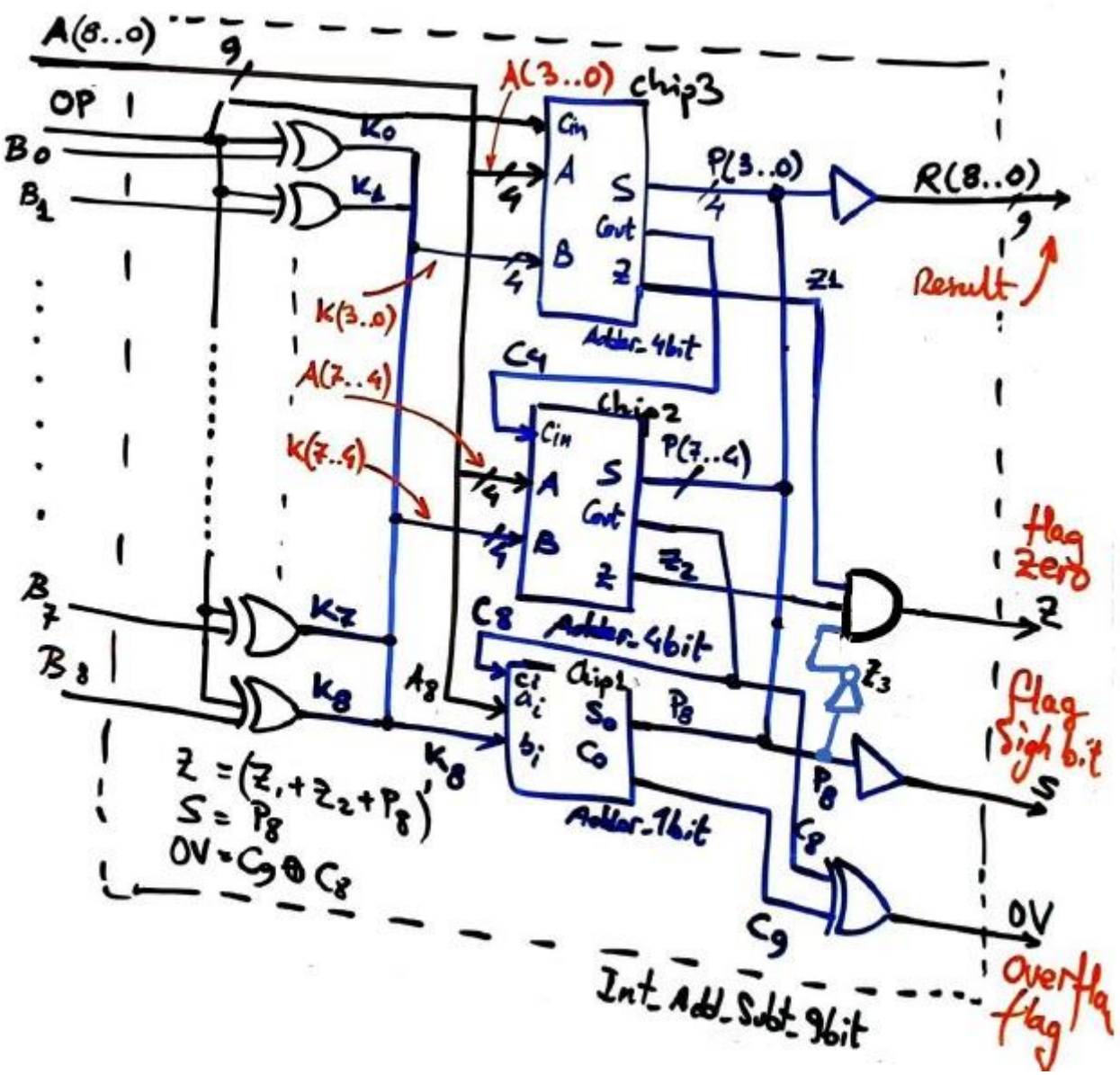

Fig. 4. Schematic plan C annotated ready for translation to VHDL.

c. Redraw circuit in Fig. 5 in your own style. Discuss why such circuit may work correctly. Write circuit's truth table using some examples and a sketch an example of timing diagram. Redraw circuit in Fig. 6 in your own style. Discuss why such circuit may work correctly. Write circuit's truth table using some examples and a sketch an example of timing diagram.

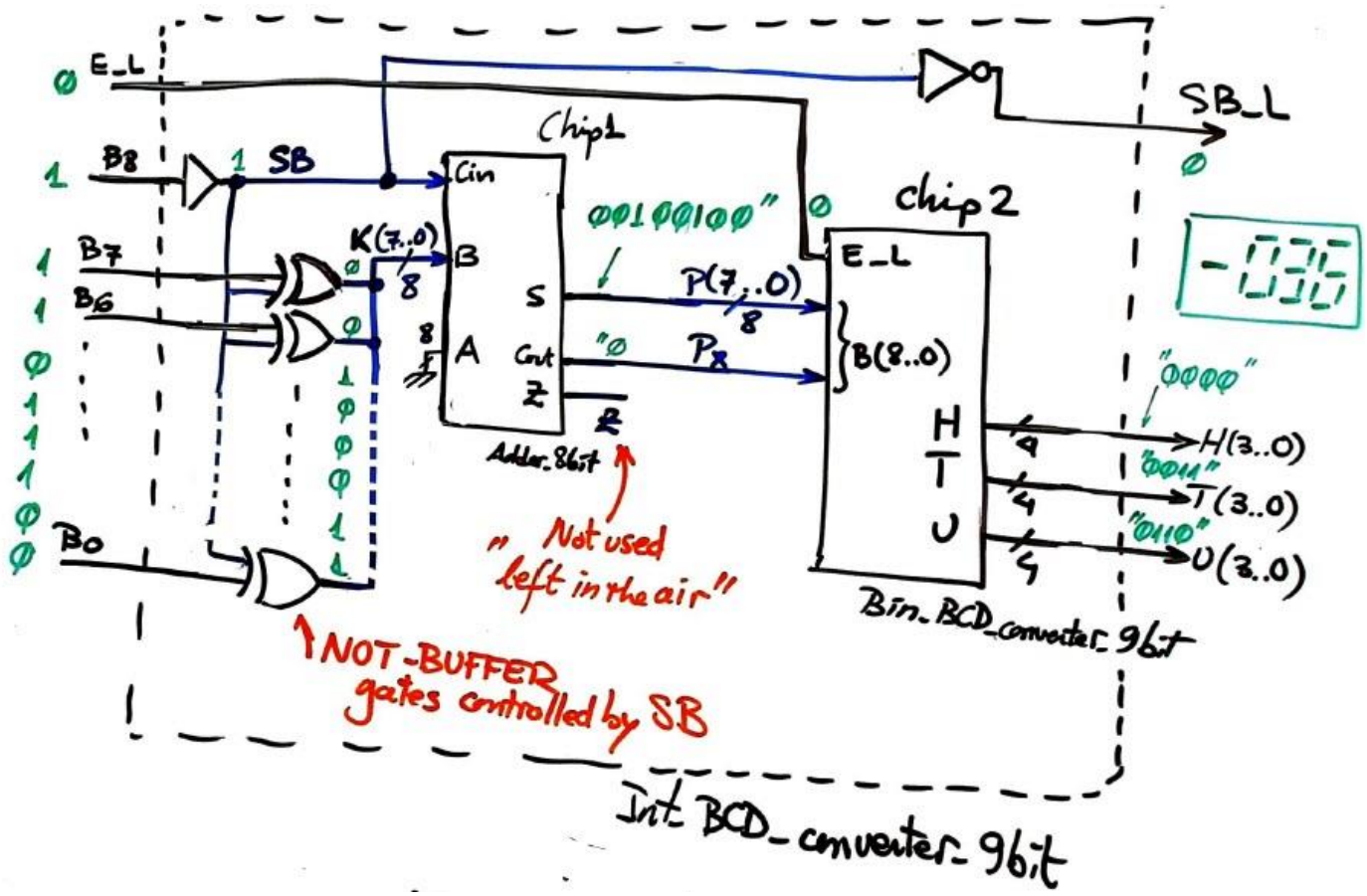

Fig. 5. Schematic plan C annotated ready for translation to VHDL.

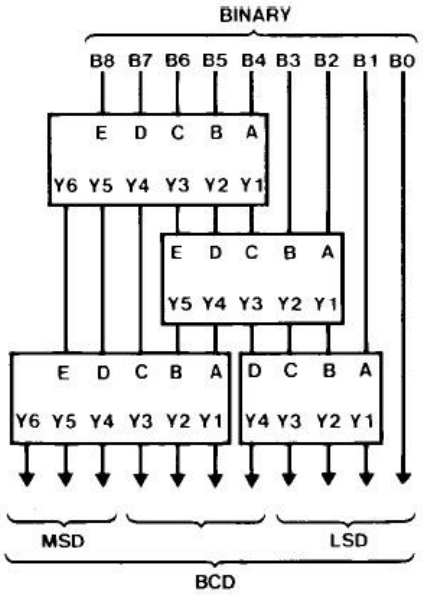

FIGURE 6. 9-Bit Binary-to-BCD Converter

MSD—Most significant decade

LSD—Least significant decade

Note A: Each rectangle represents a DM74185A.

Note B: All unused E inputs are grounded.

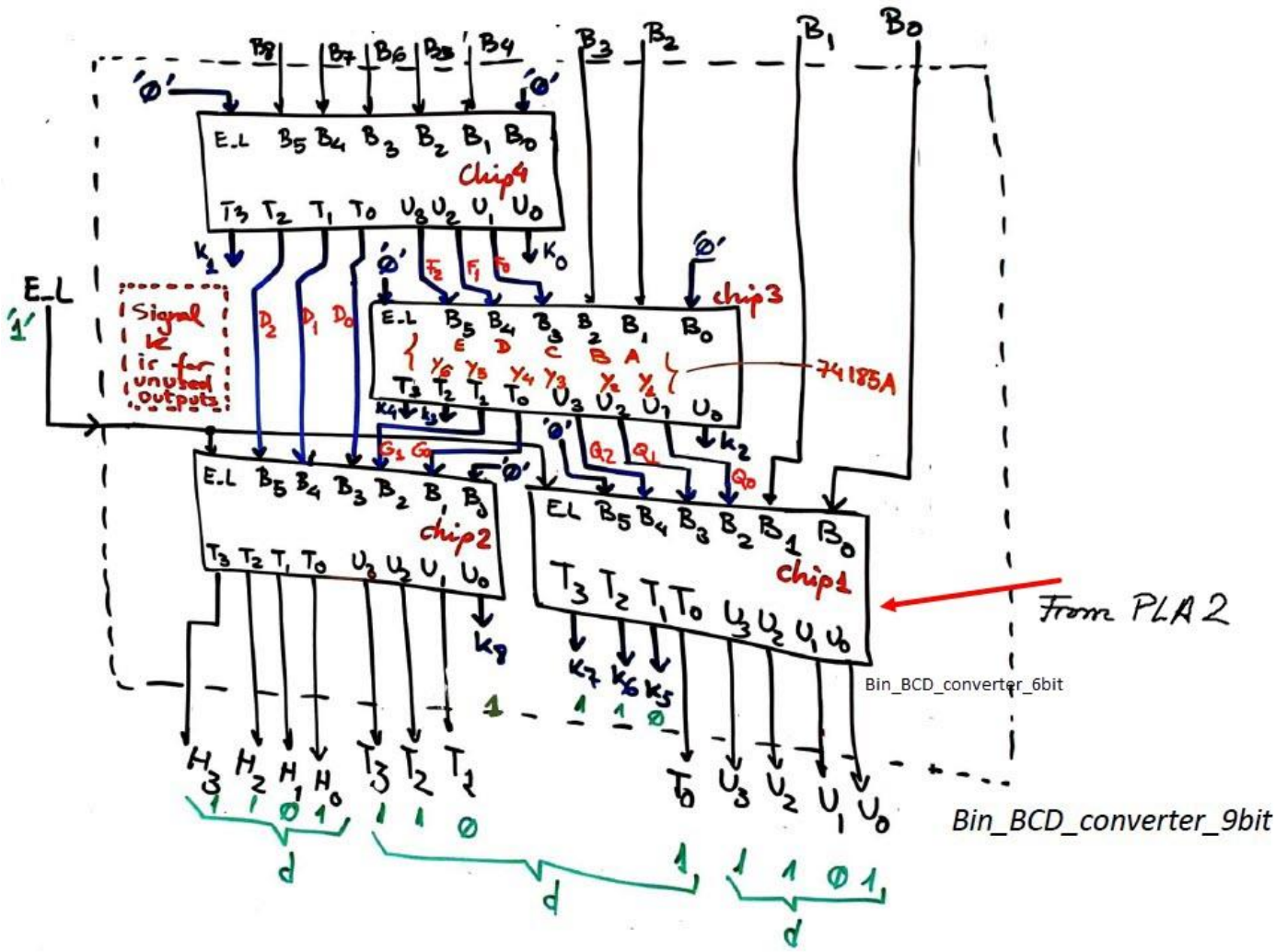

Fig. 6. Circuit from 74185A datasheet and schematic plan C annotated ready for translation to VHDL.

d. How to build the *Hex_7seg_type7447* using our *HEX_7seg_decoder* from P2 as a component? Discuss the logic of the extra ripple blanking output (RBO_L) signal. Control signals LT_T and BI_L are not necessary in this application, thus, there is no need to implement them. What kind of connections are required to blank leading zeros at the 7-segment display? Demonstrate how it works using simulations based on our Proteus circuit. (1) and (2).

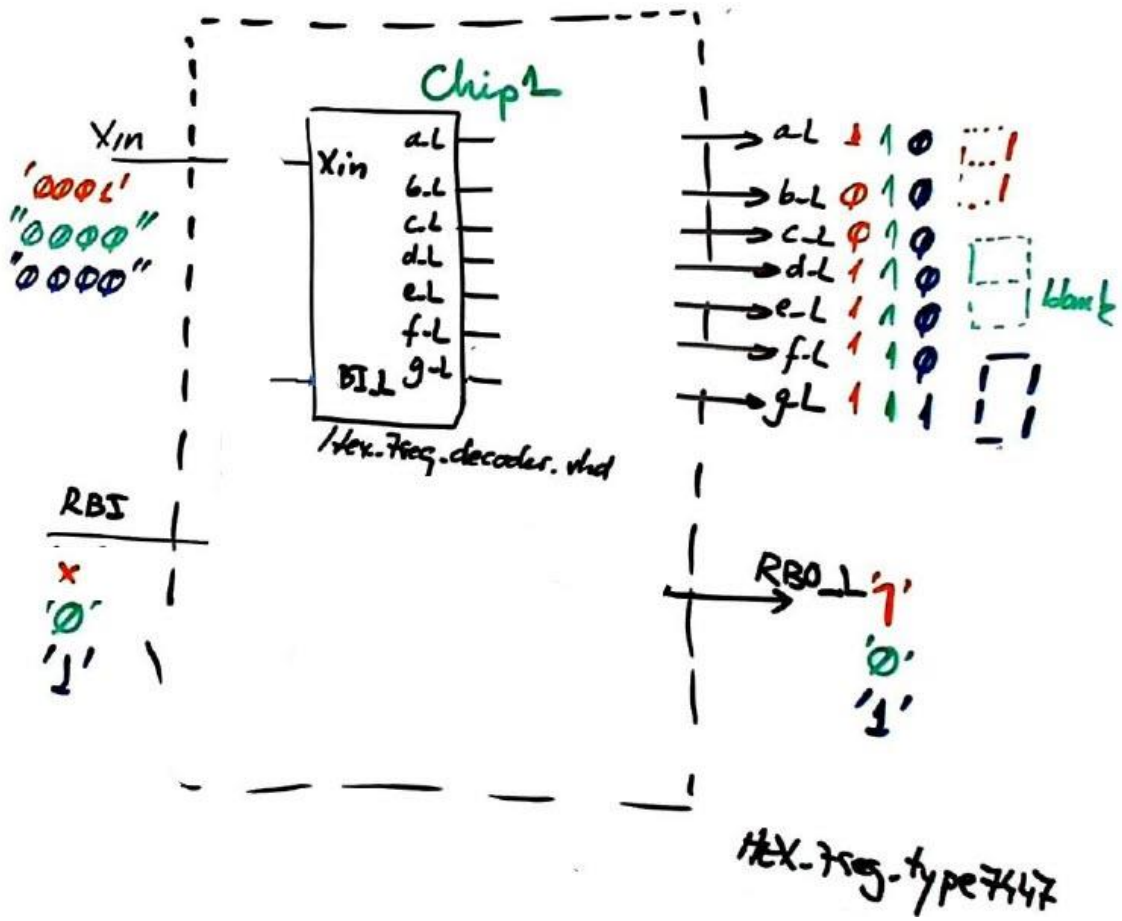

Fig. 7. Idea for implementing Hex_7seg_type7447 using a HEX_7seg_decoder

3. **Development and**
4. **Functional testing**

a. Develop Chip1 and perform a functional test as an independent project. This task means synthesise circuits, discuss results, organise a testbench from initial timing diagram sketches, etc.
b. Develop Chip2 and perform a functional test as an independent project
c. Develop Chip3 and perform a functional test as an independent project.

d. Develop and test the full circuit *Int_Add_Subt_9bit_7seg*.

5. **Gate-level simulation**

Run a gate-level simulation and calculate propagation delay for a given signal transition, worst-case propagation delay and maximum speed of processing. Use ModelSim and Timing Analyser tools to perform measurements and calculations.

NOTE: Solve this section for only one chip

6. **(Optional, demonstration) Prototyping in a target board Terasic -DE2-115**
7. **Report and marking**

Report consisting of scanned figures and handwritten discussions, organised as follows: [1] – [2] – [3-4] – [5].

Section 1 on project specification is the same for both plans, thus solving and reporting it one time is all right. (2p, 0.5p each question)
Sections 2 on planning (2p, 0.5p each question)
Section 3-4 on development and functional testing designed circuits (4p, 1p/chip)

Section 5 on VHDL gate-level simulation and timing analyser calculations (2p)