



dscal
DIGITAL SYSTEMS & COMPUTER ARCHITECTURE LABORATORY

Εργαστήριο Λογικής Σχεδίασης

Παράδειγμα ανάπτυξης Project στο Vivado

Βασιλόπουλος Διονύσης

Ε.ΔΙ.Π Τμήματος Πληροφορικής & Τηλεπικοινωνιών - ΕΚΠΑ

VHDL - Παράδειγμα

Πραγματικό πρόβλημα

Σε ένα Computer Room υπάρχουν δύο (2) αισθητήρες θερμοκρασίας (**Sensor_1** και **Sensor_2**) και δύο (2) κλιματιστικά (**AirCond_1** και **AirCond_2**). Το πρώτο κλιματιστικό (AirCond_1) λειτουργεί εάν τουλάχιστον ένας από τους δύο αισθητήρες ανιχνεύσουν θερμοκρασία άνω των 35 βαθμών στο computer room. Το δεύτερο κλιματιστικό (AirCond_2) λειτουργεί εάν και οι δύο αισθητήρες ανιχνεύσουν θερμοκρασία άνω των 35 βαθμών στο computer room. Θεωρείστε ότι κάθε αισθητήρας δίνει σήμα ('1') μόνο όταν η θερμοκρασία που ανιχνεύει γίνει μεγαλύτερη των 35 βαθμών (>35). Σε άλλη περίπτωση ο αισθητήρας στέλνει την τιμή '0'. Σχεδιάστε και υλοποιήστε το λογικό κύκλωμα που περιγράφει το ανωτέρω πρόβλημα. Το όνομα του Vivado Project θα είναι Lab_1, της οντότητας θα είναι CR_AC ενώ το όνομα της αρχιτεκτονικής Dataflow.

Σχεδιάστε και υλοποιήστε το λογικό κύκλωμα που περιγράφει το ανωτέρω πρόβλημα.

VHDL - Παράδειγμα

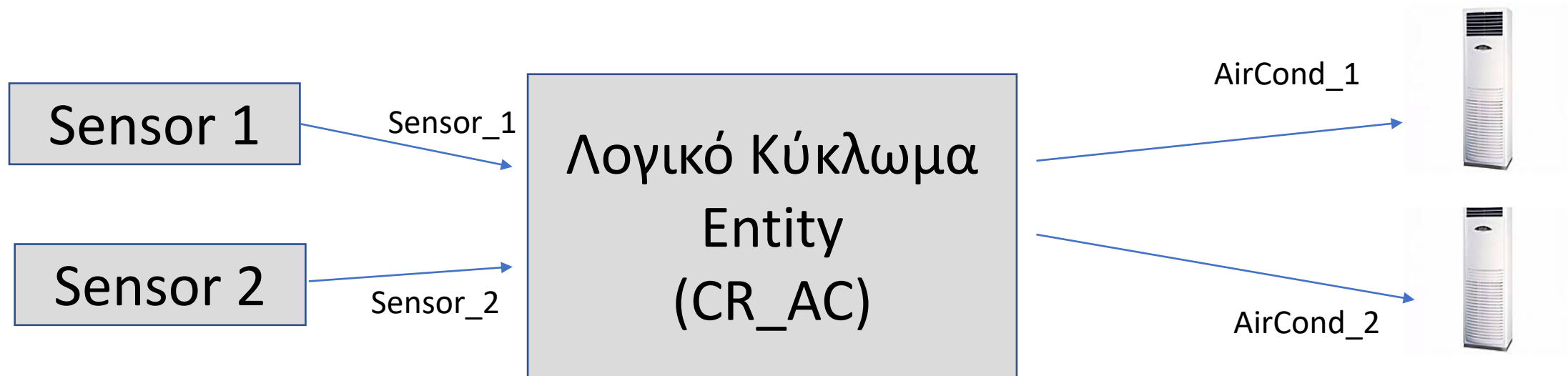
Πραγματικό πρόβλημα – Βήματα επίλυσης

1. Δημιουργία νέου project
2. Δημιουργία Entity – Εντοπισμός Input/Output του συστήματος
3. Δημιουργία Architecture – Θα έχετε (τουλάχιστον) τόσες εντολές όσες είναι και οι έξοδοι του συστήματος. Κάθε μία εντολή αντιστοιχεί σε μία έξοδο.
4. Δημιουργία RTL αναπαράστασης
5. Προσομοίωση

(Αναλυτικές οδηγίες στον Οδηγό Vivado στο eclass)

VHDL - Παράδειγμα

Απλοποιημένη μορφή κυκλώματος



VHDL - Παράδειγμα

Περιγραφή Οντότητας (entity)

```
library IEEE;  
use IEEE.STD_LOGIC_1164.all;  
  
entity CR_AC is  
port (  
    Sensor_1      :in STD_LOGIC;  
    Sensor_2      :in STD_LOGIC;  
    AirCond_1     :out STD_LOGIC;  
    AirCond_2     :out STD_LOGIC  
);  
end entity CR_AC;
```

VHDL - Παράδειγμα

Λογική κατανόηση της άσκησης

- AirCond_1:** Ανάβει όταν δίνει σήμα τουλάχιστον ένας αισθητήρας =>
 Ανάβει όταν δίνει σήμα οποιοσδήποτε αισθητήρας
- AirCond_2:** Ανάβει όταν δίνουν σήμα και οι δύο αισθητήρες

VHDL - Παράδειγμα

Πίνακας Αληθείας

Είσοδοι (Inputs)		Έξοδοι (outputs)	
Sensor_1	Sensor_2	AirCond_1	AirCond_2
0	0	0	0
0	1	1	0
1	0	1	0
1	1	1	1

VHDL - Παράδειγμα

Πίνακας Αληθείας - Ελαχιστόροι

Είσοδοι (Inputs)		Έξοδοι (outputs)	
Sensor_1	Sensor_2	AirCond_1	AirCond_2
0	0	0	0
0	1	1	0
1	0	1	0
1	1	1	1

AirCond_1= $Sensor_1' * Sensor_2 + Sensor_1 * Sensor_2' +$
 $Sensor_1 * Sensor_2 =$
 $Sensor_2 + Sensor_1 * Sensor_2' =$
Sensor_2 + Sensor_1

AirCond_2= $Sensor_1 * Sensor_2$

***=AND**

+ =OR

VHDL - Παράδειγμα

Υλοποίηση Αρχιτεκτονικής (Dataflow)

```
architecture Dataflow of CR_AC is  
begin
```

```
    AirCond_1<=Sensor_1 or Sensor_2;
```

```
    AirCond_2<=Sensor_1 and Sensor_2;
```

```
end architecture Dataflow;
```

VHDL - Παράδειγμα

Τελική έκδοση προγράμματος

```
library IEEE;  
use IEEE.STD_LOGIC_1164.ALL;
```

```
Entity CR_AC is
```

```
Port (
```

```
Sensor_1 :in std_logic;
```

```
Sensor_2 :in std_logic;
```

```
AirCond_1 :out std_logic;
```

```
AirCond_2 :out std_logic
```

```
);
```

```
end entity CR_AC;
```

```
architecture Dataflow of CR_AC is
```

```
begin
```

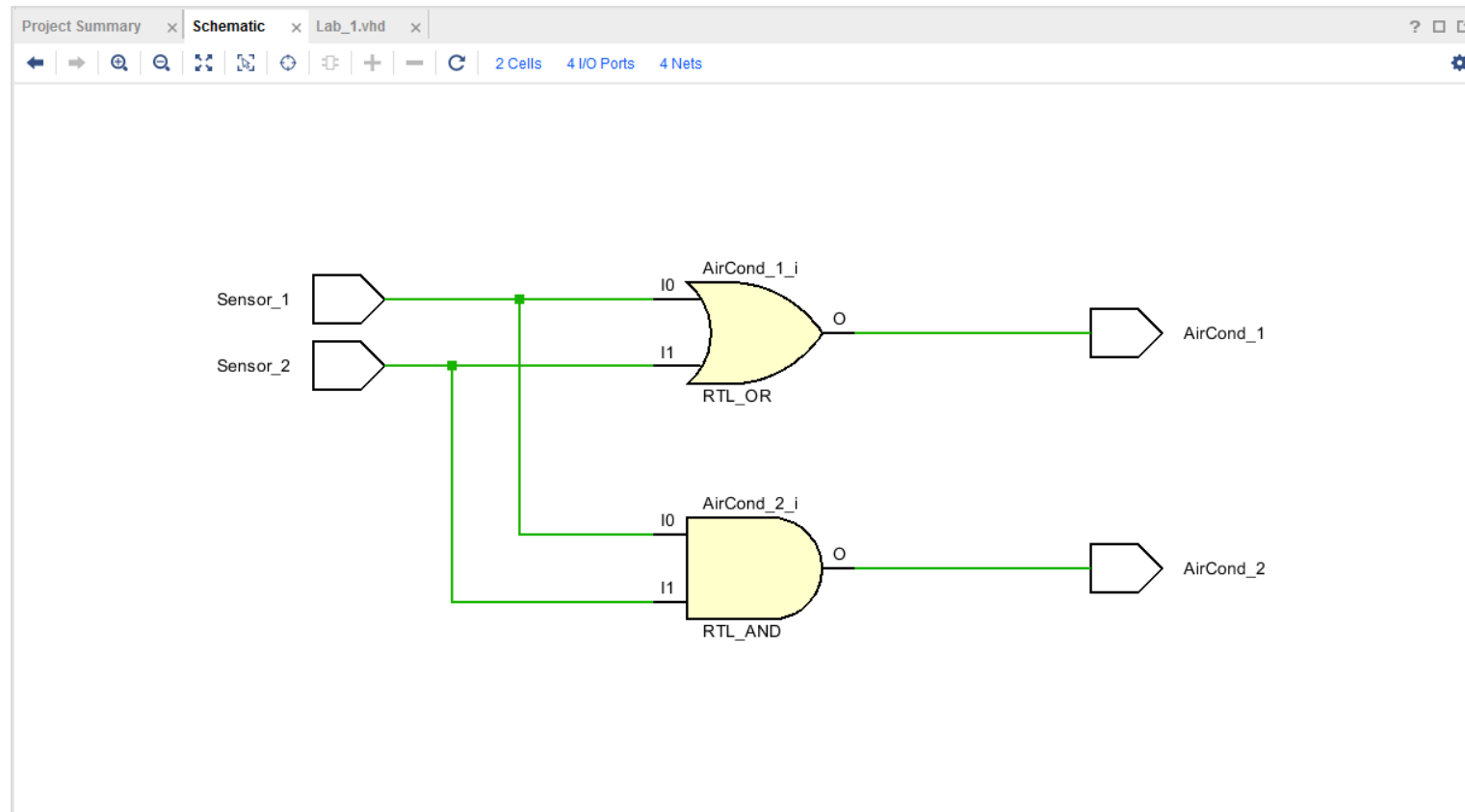
```
AirCond_1<=Sensor_1 or Sensor_2;
```

```
AirCond_2<=Sensor_1 and Sensor_2;
```

```
end architecture Dataflow;
```

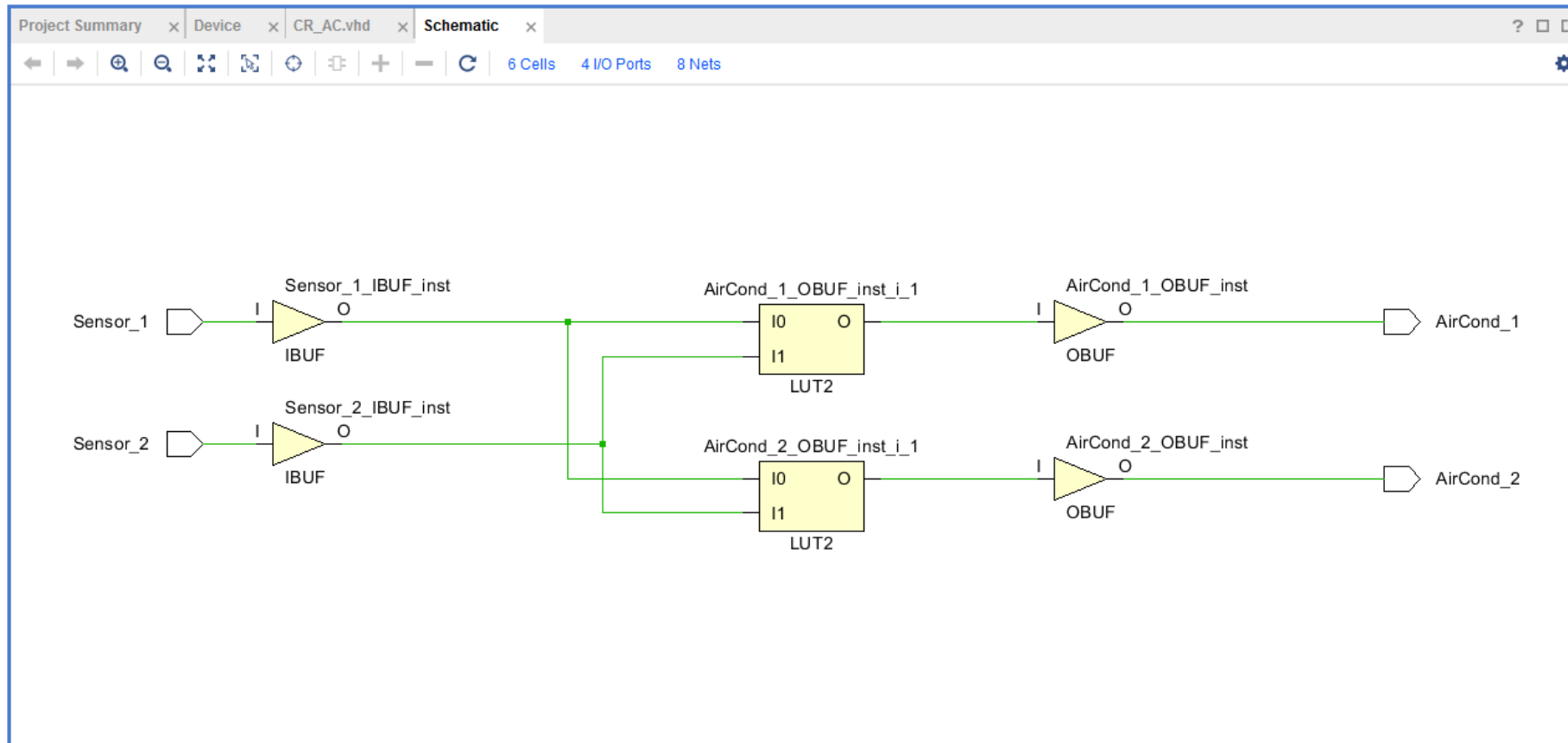
VHDL - Παράδειγμα

Λογικό κύκλωμα: Αναπαράσταση RTL



VHDL - Παράδειγμα

Λογικό κύκλωμα: Φάση Σύνθεσης – LUT(LookUp Table), υλοποιούν Πίνακες Αληθείας (προγραμματιζόμενα μέρη της κάρτας FPGA)



VHDL - Παράδειγμα

Λογικό κύκλωμα: Φάση Σύνθεσης – LUT(LookUp Table), υλοποιούν Πίνακες Αληθείας (προγραμματιζόμενα μέρη της κάρτας FPGA)

Nets->Σύρματα

Cells->Ψηφιακά στοιχεία

The screenshot shows the Vivado 2019.2 interface. The left sidebar contains the Project Manager and Synthesis sections. The main window displays a schematic diagram of a logic circuit. The circuit includes two sensors (Sensor_1 and Sensor_2) connected to inverters (IBUF). The outputs of these inverters are connected to two LUT2 cells (AirCond_1_OBUF_inst_i_1 and AirCond_2_OBUF_inst_i_1). The outputs of the LUT2 cells are connected to inverters (OBUF) and then to the final outputs (AirCond_1 and AirCond_2). The Truth Table for the LUT2 cell is shown in the bottom right, with the following data:

I1	I0	O=I0+I1
0	0	0
0	1	1
1	0	1
1	1	1

LUT: Πίνακας Αληθείας

VHDL - Παράδειγμα

Προσομοίωση (Πρόγραμμα testbench)

```
library IEEE; use IEEE.STD_LOGIC_1164.ALL;
```

```
entity CR_AC_tb is  
end CR_AC_tb;
```

```
architecture Behavioral of CR_AC_tb is
```

```
component CR_AC is  
Port (  
Sensor_1 :in std_logic;  
Sensor_2 :in std_logic;  
AirCond_1 :out std_logic;  
AirCond_2 :out std_logic);  
end component CR_AC;
```

```
-- Internal Signals: One for each port  
signal Sensor_1_tb :std_logic; signal Sensor_2_tb :std_logic;  
signal AirCond_1_tb :std_logic; signal AirCond_2_tb :std_logic;
```

```
begin
```

```
-- create entity into testbench entity  
test_entity: CR_AC port map (Sensor_1=>Sensor_1_tb,  
Sensor_2=>Sensor_2_tb, AirCond_1=>AirCond_1_tb,  
AirCond_2=>AirCond_2_tb);
```

```
enter_test_values: process is  
begin
```

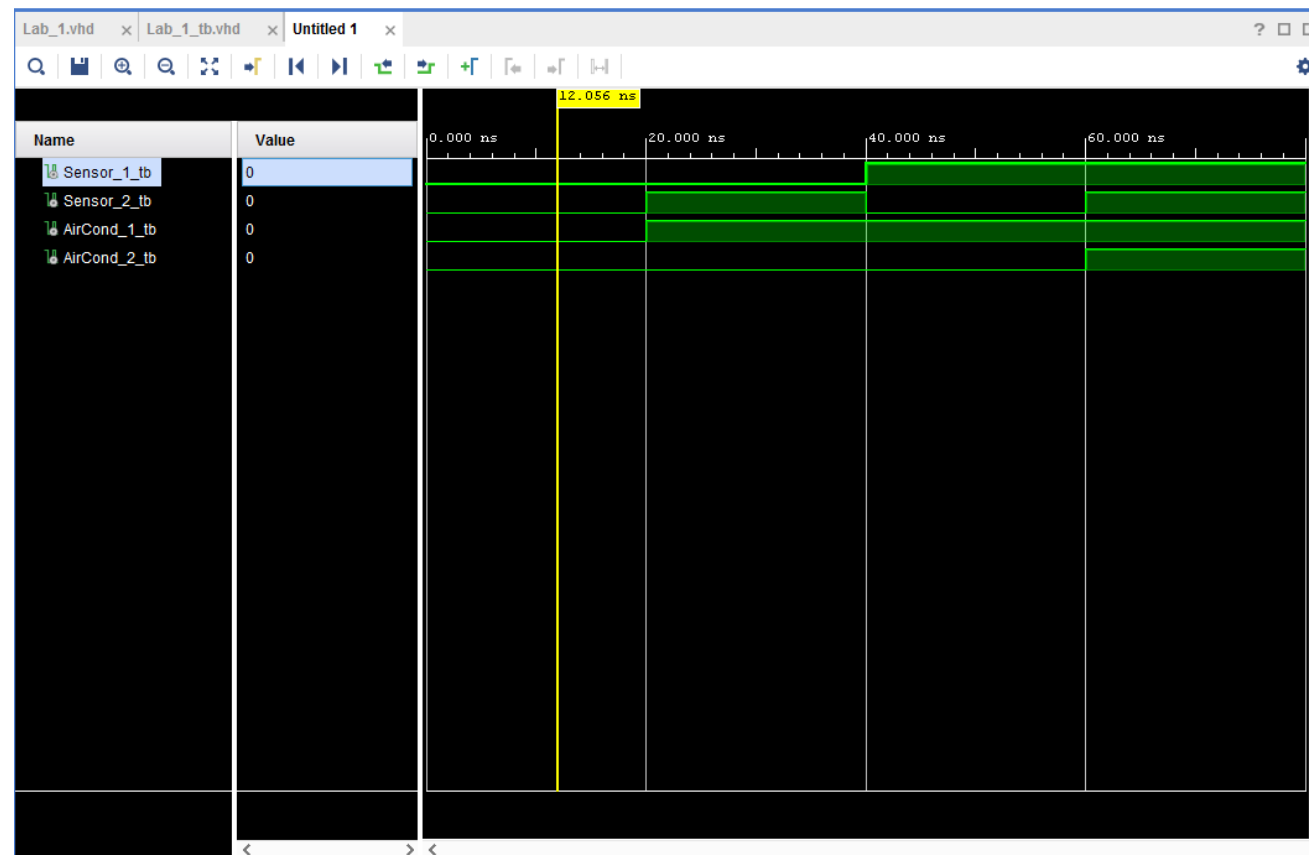
```
Sensor_1_tb<='0';Sensor_2_tb<='0';wait for 20 ns;  
Sensor_1_tb<='0';Sensor_2_tb<='1';wait for 20 ns;  
Sensor_1_tb<='1';Sensor_2_tb<='0';wait for 20 ns;  
Sensor_1_tb<='1';Sensor_2_tb<='1';wait for 20 ns;
```

```
end process enter_test_values;
```

```
end architecture Behavioral;
```

VHDL - Παράδειγμα

Προσομοίωση Behavioral (Χρονοσειρά)



Περίληψη

- Ανάπτυξη βήμα-βήμα μιας απλής εφαρμογής στο Vivado
- RTL->Synthesis
- Προσομοίωση
- LUT