

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ



ΕΡΓΑΣΤΗΡΙΟ

ΣΧΕΔΙΑΣΗΣ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ - VHDL

8^ο Εργαστηριακό Μάθημα
2^ο Εργαστήριο

Βασιλόπουλος Διονύσης

ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ
2023-2024

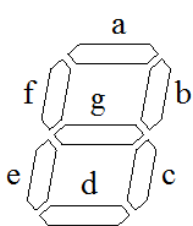
Άσκηση: Αποκωδικοποιητής BCD σε 7-segment

Στην άσκηση αυτή, θα χρησιμοποιήσουμε την αριθμητική κωδικοποίηση *δυναμικά κωδικοποιημένων δεκαδικών (binary coded decimal – BCD)*. Εάν θεωρήσουμε ένα μόνο δεκαδικό ψηφίο, οι δέκα πιθανές τιμές είναι 0, 1, 2, 3, 4, 5, 6, 7, 8 και 9. Χρειαζόμαστε τουλάχιστον 4 bit σε ένα δυαδικό κώδικα και ο κώδικας BCD που είναι ο πιο συνηθισμένος, και έχει τις ακόλουθες κωδικές λέξεις:

0: 0000 1: 0001 2: 0010 3: 0011 4: 0100
5: 0101 6: 0110 7: 0111 8: 1000 9: 1001

Αν έχουμε περισσότερα από ένα δεκαδικά ψηφία για αναπαράσταση, απλά χρησιμοποιούμε τετράδες bit, όπου καθεμία παριστάνει ένα δεκαδικό ψηφίο. Για παράδειγμα, ένα σύστημα που ασχολείται με τριψήφιους αριθμούς θα χρησιμοποιούσε έναν κώδικα με 12 bit. Ο αριθμός 123 τότε θα είχε την κωδικοποίηση 0001 0010 0011.

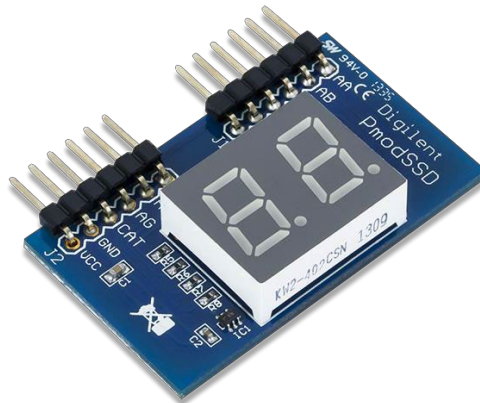
Πολλά ψηφιακά συστήματα εμφανίζουν τους δεκαδικούς αριθμούς με χρήση οθονών 7 τμημάτων (7-segment displays). Κάθε ψηφίο της οθόνης αποτελείται από επτά ξεχωριστά φώτα, που τοποθετούνται όπως φαίνεται στην παρακάτω εικόνα. Αν έχουμε ένα ψηφίο κωδικοποιημένο με BCD και πρέπει να δείξουμε το ψηφίο σε μια οθόνη 7 τμημάτων, χρειαζόμαστε έναν *αποκωδικοποιητή 7 τμημάτων (7-segment decoder)*. Μιλώντας αυστηρά, θα το ονομάζαμε «μετατροπέα κώδικα 7 τμημάτων» αφού μετατρέπει μια είσοδο με κώδικα BCD σε μια έξοδο με κώδικα 7 τμημάτων. Ωστόσο, ο όρος «αποκωδικοποιητής 7 τμημάτων» χρησιμοποιείται ευρύτατα. Υποθέτοντας ότι ένα τμήμα ανάβει αν η είσοδός του είναι 1, χρειαζόμαστε έναν κώδικα των 7 bit για την αναπαράσταση των ψηφίων από 0 μέχρι 9. Η κωδική λέξη για κάθε ψηφίο έχει ένα bit 1 για κάθε τμήμα που είναι αναμμένο και ένα bit 0 για κάθε τμήμα που δεν είναι αναμμένο. Τότε, ένας αποκωδικοποιητής 7 τμημάτων μετατρέπει μεταξύ του BCD και αυτού του κώδικα των 7 bit. Ένας πιθανός κώδικας φαίνεται στην παρακάτω εικόνα, με τα bit να αντιστοιχούν από αριστερά προς τα δεξιά στα τμήματα g μέχρι a (g, f, e, d, c, b, a).



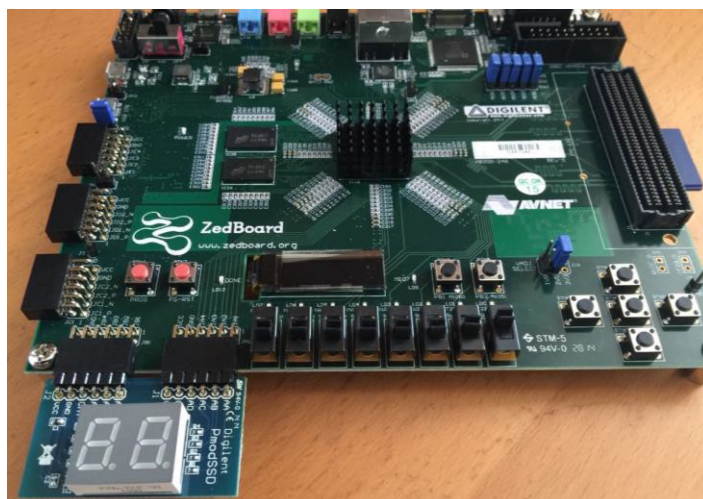
0111111	0000110	1011011	1001111	1100110
1101101	1111101	0000111	1111111	1101111

Να σχεδιάσετε, να προσομοιώσετε και να υλοποιήσετε ένα κύκλωμα αποκωδικοποιητή BCD σε 7-segment που θα δέχεται ως είσοδο στα DIP switches SW3 (MSbit), SW2, SW1, SW0 (LSbit) της αναπτυξιακής κάρτας ένα δεκαδικά κωδικοποιημένο δεκαδικό αριθμό (BCD) και θα οδηγεί (θα το εμφανίζει σε) ένα από τα δύο ψηφία του 7-segment display (Pmod SSD κοινής καθόδου της Digilent) που συνδέεται στην αναπτυξιακή κάρτα ZedBoard μέσω των GPIO σημάτων του Bank 13 (3.3V) και των JA1 και JB1 connectors.

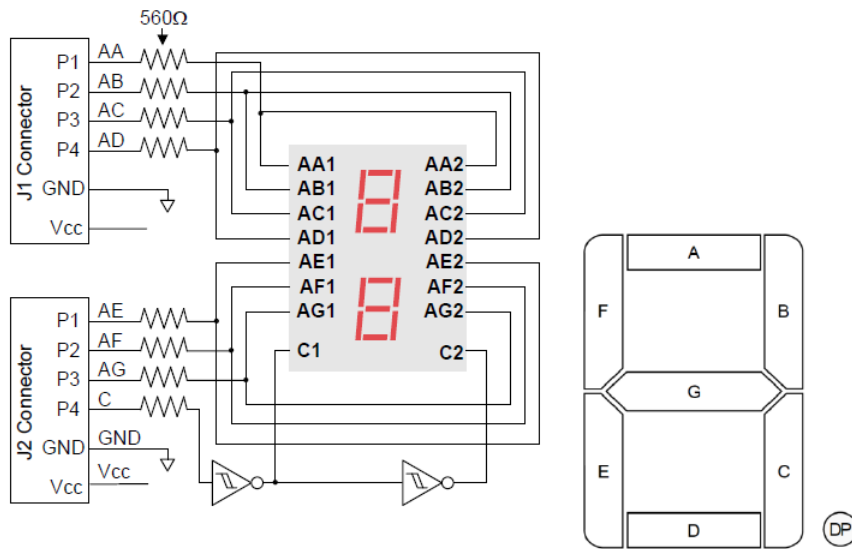
Το Pmod SSD υποστηρίζει δύο ψηφία. Τα segments ανάβουν οδηγώντας με λογικό 1 την αντίστοιχη άνοδο σε όποιο από τα δύο ψηφία έχει επιλεγεί. Ο χρήστης μπορεί να επιλέξει ένα συγκεκριμένο ψηφίο χρησιμοποιώντας την είσοδο DIP switch SW4 και οδηγώντας αντίστοιχα το Digit Selection pin (C) του Pmod SSD σε λογικό 1 (για το αριστερό ψηφίο) ή 0 (για το δεξί ψηφίο). Το datasheet του Pmod SSD βρίσκεται στο e-class.



Εικόνα 1. Pmod SSD της Digilent



Εικόνα 2. Σύνδεση του Pmod SSD στα JA1 και JB1 Pmod connectors της ZedBoard



Εικόνα 3. Σχηματικό διάγραμμα του Pmod SSD της Digilent

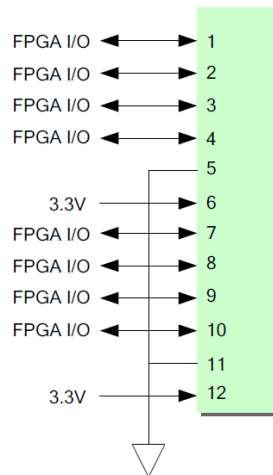
Header J1			Header J2		
Pin	Signal	Description	Pin	Signal	Description
1	AA	Segment A	1	AE	Segment E
2	AB	Segment B	2	AF	Segment F
3	AC	Segment C	3	AG	Segment G
4	AD	Segment D	4	C	Digit Selection pin
5	GND	Power Supply Ground	5	GND	Power Supply Ground
6	VCC	Positive Power Supply	6	VCC	Positive Power Supply

Πίνακας 1. Pinout του Pmod SSD της Digilent

Στα πλαίσια της άσκησης θα πρέπει να γράψετε το κατάλληλο αρχείο .xdc με τα constraints για τα pin assignments λαμβάνοντας υπόψιν α) το σχηματικό διάγραμμα του Pmod SSD (Εικόνα 3), β) τον Πίνακα 1 με το pinout του Pmod SSD γ) τον Πίνακα 2 που περιγράφει τις συνδέσεις των DIP switches με τα pins του Zynq FPGA, δ) στο σχηματικό διάγραμμα των συνδέσεων του Pmod SSD (Εικόνα 4) και ε) τον Πίνακα 3 που περιγράφει τις συνδέσεις των Pmod JA1 και Pmod JA2 με τα pins του Zynq FPGA.

Signal Name	Zynq pin
SW0	F22
SW1	G22
SW2	H22
SW3	F21
SW4	H19
SW5	H18
SW6	H17
SW7	M15

Πίνακας 2. Συνδέσεις των DIP switches στα pins του Zynq FPGA



Εικόνα 4. Σχηματικό διάγραμμα των συνδέσεων του Pmod SSD της Digilent

Pmod	Signal Name	Zyng pin	Pmod	Signal Name	Zyng pin
JA1	JA1	Y11	JB1	JB1	W12
	JA2	AA11		JB2	W11
	JA3	Y10		JB3	V10
	JA4	AA9		JB4	W8
	JA7	AB11		JB7	V12
	JA8	AB10		JB8	W10
	JA9	AB9		JB9	V9
	JA10	AA8		JB10	V8

Πίνακας 3. Συνδέσεις των JA1 και JB1 Pmods στα pins του Zyng FPGA

Στο eclass θα σας δοθεί ένα template πηγαίου κώδικα με την περιγραφή του entity το αντίστοιχο testbench σε VHDL, καθώς και το αρχείο με τα pin constraints.

- Συμπληρώστε το architecture του entity
- Ελέγξτε τη λειτουργία του κυκλώματος τόσο στη προσομοίωση όσο και στο υλικό προγραμματίζοντας το FPGA
- Συμπληρώστε τον παρακάτω πίνακα κοιτάζοντας το report μετά το implementation:

FF:	
LUT:	
I/O	

Σημείωση: Στη σχεδίαση του αποκωδικοποιητή να χρησιμοποιήσετε δομή case ή την if και όχι selected signal assignment (with-select) ή conditional signal assignment (when-else). Επίσης, εάν η BCD είσοδος είναι μεγαλύτερη του εννέα (9) η οθόνη θα πρέπει να εμφανίζει “_”.