

PERANCANGAN SISTEM DIGITAL DENGAN ALTERA UP2 CPLD BOARDS dan VHDL (Very high speed integrated circuits Hardware Description Language)

Oleh : *Meicsy E. I. Najoran, ST. MT. **

Abstrak.

Penelitian ini membahas perancangan sistem digital dengan Altera UP2 CPLD Boards dan VHDL. Suatu perancangan yang dimulai dengan merancang dari skematik atau dengan bahasa modular Hardware Description Language yang selanjutnya hasil disain ini dikompilasi dengan perangkat lunak dan menghasilkan simulasi hasil disain. Hasil dari perancangan ini kemudian dievaluasi untuk mencocokkan hasil disain dengan hasil rancangan. Perangkat lunak berupa Altera Max+Plus II digunakan sebagai compiler dan dengan menggunakan Board Altera UP2 CPLD yang terhubung ke port parallel dari PC hasil dari proses kompilasi tadi dapat di muat (download) ke board Altera untuk diimplementasikan. Hasil dari download tadi dapat diuji coba langsung pada perangkat keras. Untuk perancangan sistem digital pada tugas akhir ini menggunakan rangkaian sederhana berupa gerbang OR aktif low.

Kata kunci : VHDL, Altera CPLD, Perancangan Digital.

I. Pendahuluan.

Perkembangan teknologi digital semakin pesat, dengan dikembangkannya perangkat atau alat yang dapat membantu kinerja manusia yang lebih praktis, efisien dan berdaya guna tinggi. Alat-alat tersebut diciptakan mulai dari tahap perancangan. Perancangan merupakan suatu proses kreativitas untuk menentukan fungsi, bentuk, kapasitas dan yang pada akhirnya untuk keperluan suatu proses dan selanjutnya menjadi suatu alat yang siap digunakan. Keputusan selama fase perancangan ini berdampak pada mutu dan fungsi kinerja dari alat tersebut. Dalam merancang suatu sistem digital seringkali mengalami kesalahan sehingga menyebabkan fungsi dari sistem yang dirancang tidak sesuai.

* *Staf Pengajar Fakultas Teknik,
Jurusan Teknik Elektro Unsrat.*

Para perancang sistem biasanya menguji sistem yang sudah jadi, dan bila terjadi kesalahan maka alat tersebut tidak dapat digunakan lagi. Bila sistem yang dirancang akan diproduksi dalam bentuk yang banyak, pasti akan meningkatkan biaya produksi.

Salah satu alat yang di gunakan sebagai development board yang dibuat oleh *Altera* berupa UP2 CPDL Boards dapat digunakan untuk pengujian tahap akhir dari sistem yang dibuat dan dengan VHDL (*Very high speed integrated circuits Hardware Description Language*) digunakan untuk merancang sistem digital yang dikehendaki.

II. Ruang Lingkup Permasalahan.

Adapun ruang lingkup pembahasan sebagai berikut.

- sistem yang dirancang hanya dalam bentuk rangkaian digital yang sederhana.

- menggunakan program VHDL untuk disain sistem dan MAX+plus II sebagai compiler.
- menggunakan *UP2 Development board* untuk mendapatkan fungsi sistem sebenarnya.

III. Tujuan dan Manfaat Penelitian.

Adapun tujuan penelitian ini merancang suatu sistem digital yang bisa menghasilkan suatu hasil yang baik, efisien bahkan lebih presisi.

Manfaat dari penelitian ini, bisa bermanfaat sebagai sarana belajar untuk merancang sistem digital..

IV. Tinjauan Pustaka.

4. 1. Rangkaian Logik Digital

Informasi biner yang direpresentasi dalam sistem digital berupa besaran fisik yang disebut dengan *signal*. Signal-signal listrik seperti tegangan listrik dikelompokkan dalam dua keadaan yang mewakili biner "0" dan biner "1". Misalnya suatu sistem digital menggunakan signal berupa tegangan 3V untuk merepresentasikan biner "1" dan 0.5V untuk biner "0".

Informasi biner dimanipulasi atau diolah oleh rangkaian logik yang disebut dengan gerbang-gerbang (*gates*). Gerbang-gerbang berupa blok-blok perangkat keras yang akan menghasilkan signal "0" atau "1" bila logika masukan diberikan. Tiga operasi dasar gerbang logika seperti *AND*, *OR* dan *NOT*, yang kemudian dapat dirangkaikan membentuk gerbang-gerbang yang lain seperti *Inverter*, *buffer*, *NAND*, *NOR*, *exclusive-OR (XOR)*, dan *exclusive-NOR*. Setiap gerbang ini mempunyai masing-masing bentuk simbol rangkaian digital, fungsi aljabar dan tabel kebenarannya.

Dalam melakukan perancangan sistem digital, ada beberapa model analisa yang dapat digunakan guna menghasilkan hasil yang benar, efektif dan efisien, seperti :

- Aljabar Boolean

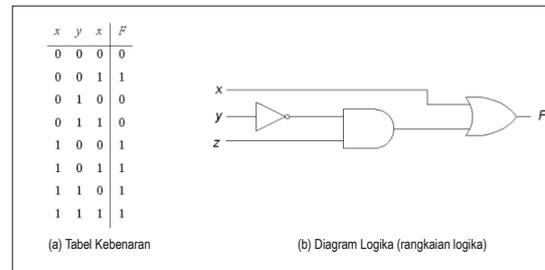
Aljabar Boolean berkaitan dengan variabel-variabel biner dan operasi-operasi logika. Variabel-variabel yang ditulis dengan huruf seperti *A*, *B*, *x* dan *y*. Fungsi Boolean adalah suatu bentuk ekspresi aljabar yang terdiri dari variable-variabel biner, simbol-simbol operasi logika, tanda kurung dan tanda sama dengan. Untuk nilai-nilai yang diberikan pada variabel-variabel, fungsi Boolean dapat menghasilkan nilai "1" atau "0".

Sebagai contoh, fungsi Boolean berikut ini :

$$F = x + y'z \dots\dots\dots (1)$$

Dimana fungsi *F* akan sama dengan 1 jika *x* = 1 atau jika *y'* dan *z* sama dengan 1; dan *F* akan bernilai 0 jika sebaliknya.

Fungsi Boolean diatas dapat ditransformasikan dari ekspresi aljabar kedalam diagram logika yang terdiri dari gerbang-gerbang *AND*, *OR* dan *inverter*, seperti pada gambar 1.



Gambar 1. Tabel kebenaran dan diagram logika untuk $F = x + y'z$

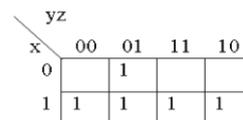
- Penyederhanaan Peta

Kompleksitas dari diagram logika yang diimplementasikan dari fungsi Boolean akan ditentukan dari ekspresi aljabar. Penyederhanaan dari ekspresi aljabar dapat dilakukan dengan fungsi dasar dari aljabar Boolean. Bentuk penyederhanaan lain adalah dengan menggunakan *eta Karnaugh*. Untuk tiap-tiap kombinasi dari variable pada tabel kebenaran disebut dengan *minterm*. Pada peta dibuat diagram berbentuk bujursangkar, dimana tiap-tiap sel direpresentasikan dengan satu minterm.

Sebagai contoh tabel kebenaran pada Gambar 1. dapat diekspresikan seperti :

$$F(x, y, z) = \sum (1, 4, 5, 6, 7) \dots\dots\dots (2)$$

Huruf dalam tanda kurung adalah variable biner, dan tanda \sum adalah jumlah dari minterm (*nilai "1" pada kolom F*) yang terdapat dalam tanda kurung. Peta karnaugh dapat ditunjukkan seperti pada gambar 2.

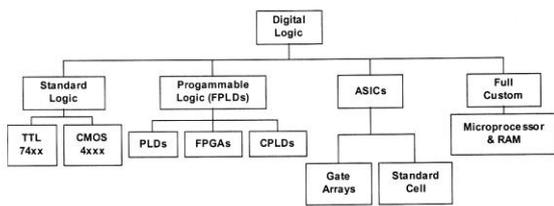


Gambar 2. Karnaugh Map untuk $F(x, y, z) = \sum (1, 4, 5, 6, 7) = x + y'z$

4.2 Peralatan Logik yang dapat diprogram (*Programmable Logic Device / PLD*)

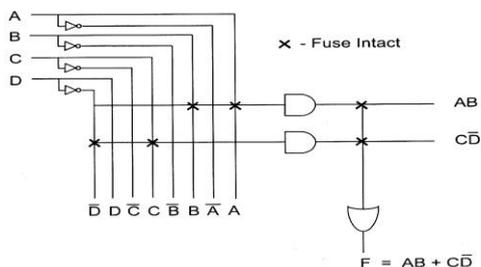
Rangkaian logik digital yang dirancang dapat diimplementasikan dengan *integrated circuit*

chip (Chip IC) seperti SSI dan MSI TT, dimana Chip ini akan melakukan fungsi operasi yang tetap sesuai pabrikan. Seorang perancang menghubungkan beberapa tipe chip berbeda untuk membuat rangkaiannya. Peralatan-peralatan yang tersedia untuk implementasi perancangan digital dapat dilihat pada gambar 3. Fungsi operasi internal dari chip yang dapat didefinisikan oleh perancang seperti ASICs (*Application Specific Integrated Circuits*), FPGAs (*Field Programmable Gate Arrays*) dan CPLDs (*Complex Programmable Logic Devices*).



Gambar 3. Teknologi Logic Digital

Programmable Logic Device adalah salah satu teknologi programmable logic dimana operasional fungsi internal IC dapat didefinisikan sendiri oleh perancang (user) dengan terlebih dahulu memprogram operasi-operasi yang diinginkan. Salah satu contoh dari Programmable Logic Device adalah *programmable logic array (PLAs)* yang diimplementasikan pada *Sum Of Product*, seperti pada gambar 4 berikut ini.



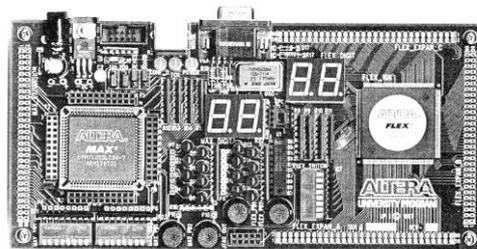
Gambar 4. PLA diimplementasikan pada *Sum Of Product*

Persamaan logic pertama diminimize dan ditempatkan dalam bentuk Sum Of Product (SOP). PLA mempunyai empat input, A, B, C dan D dimana tiap input dihubungkan ke inverter. Masing-masing titik diimplementasikan dengan AND gate dengan input, dan output gerbang AND dimasukkan ke gerbang OR. Pada ujung dari input gerbang AND diletakkan fuses pada input A dan B yang akan

menghasilkan perkalian antara A dan B. Hal yang sama juga untuk gerbang AND yang bawah. Jadi dengan adanya teknologi ini, dapat dibuat juga dengan cara yang sama untuk perancangan digital yang sudah lebih rumit.

4.3. ALTERA UP2 CPLD BOARDS

Altera University Program 2 (UP 2), CPLD disain board seperti pada gambar 2.13. Boards ini mendukung peralatan MAX EPM 7128S 2.500 gerbang, P-term based peralatan CPLD yang berada pada sisi kiri dan FLEX EPF10K20 20.00 gerbang, SRAM based peralatan CPLD yang berada pada sisi kanan.



Gambar 5. Altera UP2 CPLD Board

Kedua peralatan yang terdapat pada board ini dapat diprogram menggunakan kabel JTAG ByteBlaster yang terhubung ke PC melalui printer port. Jumper pada board harus dipilih sesuai dengan tipe peralatan yang akan digunakan.

4.4. Perancangan VHDL

Perancangan VHDL terdiri dari beberapa perancangan unit secara terpisah, masing-masing melalui proses kompilasi dan disimpan dalam suatu library. Ada 4 unit yang dapat dikompilasi, yaitu :

- **Entity unit** menjelaskan signal antarmuka dan merepresentasikan blok dasar yang dibangun sesuai perancangan.
- **Architecture unit** menjelaskan perilaku. Satu entity dapat memiliki beberapa architecture, perilaku atau struktur.
- **Configuration unit** memilih bentuk variasi perancangan dari disain library.
- **Package unit**, paket-paket yang tersimpan dan dapat dipakai berulang-ulang. Ini mirip dengan subprogram pada bahasa pemrograman lainnya. Paket ini dapat menjadi toolbox yang digunakan untuk perancangan.

Contoh dari perancangan unit adalah sbb:

```

PACKAGE my_units IS          -- Package --
    CONSTANT unit_delays; TIME :=1 NS;
    END my_units

ENTITY compare IS           -- Entity --
    PORT      (a, b : in bit ;
               c : out bit);
    END compare

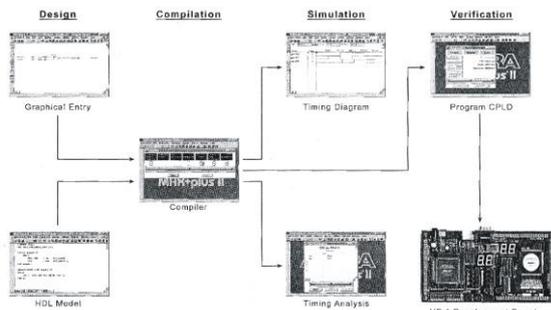
LIBRARY my_library;        -- Architecture --
USE my_library.my_units.all;

ARCHITECTURE first OF compare IS
BEGIN
    c <=NOT (a XOR b) after unit_delay;
END first
    
```

V. Implementasi Sistem Digital Dengan Altera UP2 CPLD Boards Dan VHDL.

5. 1. Disain, Syntesis, Simulasi dan Implementasi Sistem Digital Sederhana

Disain dimulai dengan menangkap skematik (gerbang or) atau menggunakan hardware description language (HDL) seperti VHDL, AHDL atau Verilog. Disain ini juga dimungkinkan untuk kombinasi blok-blok dengan metode berbeda kedalam satu disain. Gambar 6 memperlihatkan tools yang dapat digunakan untuk mensimulasikan, menghitung waktu penundaan (*timing delays*), synthesise logic, dan memprogram disain implementasi hardware kedalam CPLD.

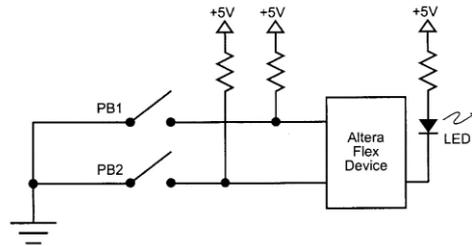


Gambar 6.

Proses Disain dengan Skematik atau VHDL

Pada Altera UP2 Board terdapat dua CPLD, yakni chip Altera MAX 7128S berisi 2500 gerbang dan chip FLEX 10K20 berisi 20.000 gerbang. Pada disain ini akan menggunakan FLEX dimana untuk tombol input PB1 dan PB2 akan dihubungkan secara

langsung ke FLEX chip pada pin 28 dan 29. Koneksi antara tombol PB1, PB2 dan LED serta Altera FLEX device dapat dilihat pada gambar 7



Gambar 7.

Koneksi antara PB1, PB2, LED dan Altera FLEX

Definisi masalah dari disain simple logic ini adalah :

$$LED = PB1 + PB2$$

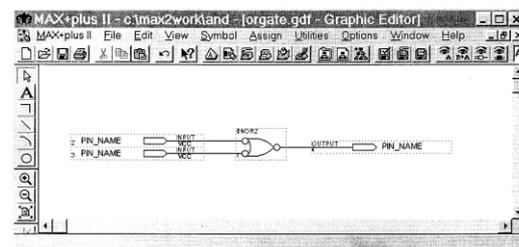
Untuk menyelesaikan kita perlu membuat rangkaiannya pada grafik editor dan implementasi pada VHDL. Karena pada tombol PB1 dan PB2 serta LED memerlukan signal balik (inverter) untuk aktif, maka kita dapat menggambarkan rangkaian logic OR seperti pada gambar 8 berikut ini :



Gambar 8.

Rangkaian Ekuivalen untuk OR dengan Input dan Output aktif Low

Grafik editor akan digunakan sebagai format masukan. Pada program MaxPLUS ada grafik editor yang akan menyimpan file dalam bentuk *.gdf. Bagian skematik terlihat seperti pada gambar 9.

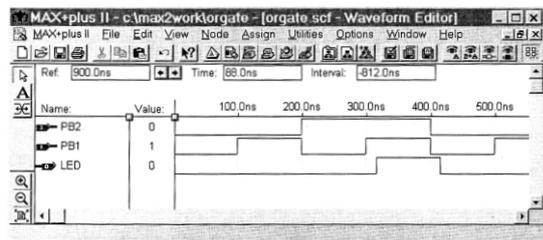


Gambar 9

Aktif low skematik gerbang OR dengan nomor pin yang ditetapkan

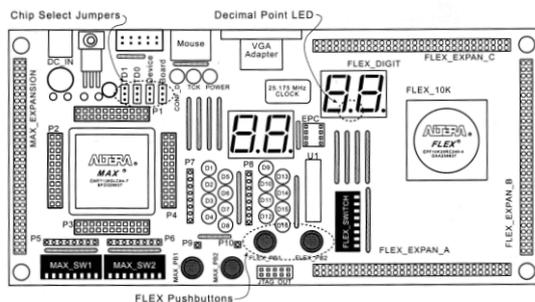
Tahap selanjutnya adalah proses kompilasi hasil untuk mengecek terjadi kesalahan, synthesise disain logic, informasi pewaktu yang dihasilkan untuk simulasi, kecocokan disain pada CPLD yang dipilih

dan membangkitkan file yang dibutuhkan untuk memuat (download) program. Bila proses kompilasi tidak terjadi kesalahan dilanjutkan dengan simulasi terlebih dahulu sebelum dimuat ke CPLD. Simulasi akan berjalan dan waveform untuk LED akan terlihat seperti pada gambar 10.



Gambar 10. Simulasi Diagram Pewaktu untuk gerbang OR aktif low

Tahap selanjutnya adalah memuat (download) hasil disain ke UP2 Development board. Pada tahapan ini perlu diatur koneksi UP2 board dengan komputer serta pemilihan chip serta mengetahui lokasi tombol dan decimal point LED (lihat gambar 11).



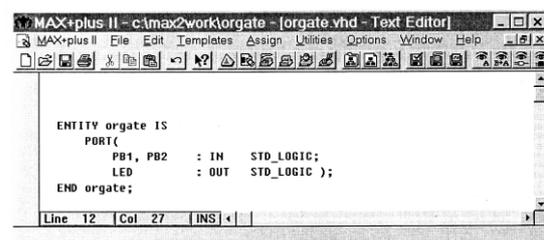
Gambar 11. Altera UP2 board dengan pengaturan jumper dan lokasi PB1, PB2 dan LED

Pengujian hasil disain. Setelah didownload ke UP2 Board, maka coba menekan tombol PB1 atau PB2 dan Keluaran akan ditampilkan dibagian seven-segment.

5. 2. Disain dengan VHDL

Cara lain selain menggunakan skematik yaitu dengan disain menggunakan *hardware description language (HDL)* seperti VHDL, Verilog atau AHDL. Untuk disain-disain yang lebih besar, lebih baik menggunakan pemrograman *HDL*, karena dapat meningkatkan produktifitas dan mengurangi period waktu perancangan. Berikut ini adalah tahapan-tahapn untuk mendisain gerbang-OR dengan menggunakan VHDL.

- Gunakan template untuk memulai proses pembuatan. Pilih *File* → *New*, pilih *Text Editor File* dan *Ok*. Tempatkan kursor dalam area text, klik kanan dan pilih *VHDL Template*. Pilih *ENTITY*, ini adalah deklarasi untuk komponen dan variabel masukan dan keluaran.
- Simpan File. Untuk menyimpan file pilih *File* → *Save As*. Ubah *Automatic Extension* ke *.vhd* (VHDL) dan simpan file sebagai *orgate.vhd* – klik *Ok*.
- Ganti komentar pada kode VHDL. Ganti *_entity_name*, dengan *orgate*.
- Deklarasi Pin I/O. Pin Input dan Output masing-masing PB1, PB2 dan LED perlu untuk dideklarasikan pada deklarasi PORT. Karena tidak ada input vector, bi-directional I/O pin, atau GENERIC deklarasi pada disain ini, hapus semua baris tersebut. File sumber akan terlihat seperti gambar 12.



Gambar 12. Entity Deklarasi Text pada VHDL

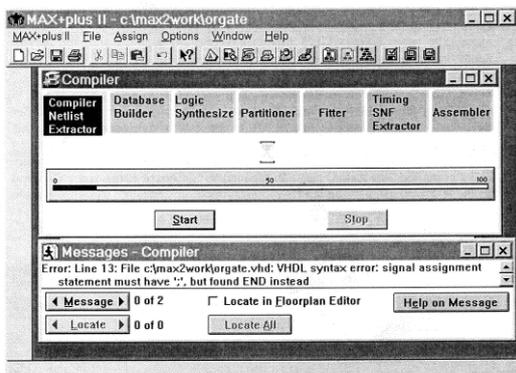
- Setting up Architecture Body. Klik mouse pada bagian bawah text (akan menyisipkan template lain). Ikuti procedure untuk memilih template VHDL, pilih *Architecture Body*. Sintax untuk *Architecture Body* terlihat pada jendela text setelah text yang lain.
- Edit Architecture Body. Ubah nama entity pada perintah *ARCHITECTURE* ke *orgate*. Hapus deklarasi dari dua baris signal karena tidak sesuai dengan signal internal dan sisipkan *LED <= NOT (NOT PB1 OR NOT PB2)* sebagai satu baris. (perintah ini salah, nanti akan ditemukan pada saat kompilasi). Sisipkan dua baris dibawah ini pada bagian awal dari file text untuk definisi libraries untuk tipe data *STD_LOGIC*.
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.all;
- Sebelum compile, tipe device dan nomor pin perlu ditetapkan. Pilih *Assign* → *Device*. Dibawah *Device Family* pilih *FLEX10K* family. Dibawah *Devices* pilih chip *EPF10K20RC240-x*.

Selanjutnya pilih *Assign* → *Pin/Location/Chip*. Pilih *Pin* dan gulung kebawah mendapatkan pin 28 untuk PB1. Jika nomor pin 28 sudah ada, itu berarti karena sudah didefinisikan dari disain secara skematik. Ulangi langkah yang sama mendapatkan pin 29 untuk PB2 dan pin 14 untuk decimal LED. Sampai disini, kode VHDL secara umum siap untuk dikompilasi, disimulasi dan didownload keyboard.

- Kompilasi VHDL design. Proses kompilasi untuk pengecekan syntax error, synthesizing logic design, informasi waktu yang dihasilkan untuk simulasi, ketepatan design pada CPLD dan membangkitkan file yang dibutuhkan untuk program ke CPLD. Kompilasi ini dilakukan dengan proses tujuh tahap.

Pilih *File* → *Project* → *Set Project to Current File*. Berikut, kompilasi dengan memilih *File* → *Project* → *Save and Compile*.

- Pengecekan untuk Kompilasi Warnings dan Errors. Project ini akan dikompilasi dengan 1 Error. Setelah proses kompilasi dari code VHDL, pada jendela indikasi akan terlihat error. Ini dapat dilihat pada gambar 13 Klik ganda pada baris error dan catat letak baris kesalahan. Kesalahan karena pada akhir perintah tidak ada *semicolon (;)*. Tambahkan kekurangan tersebut dan kompilasi lagi.



Gambar 13. Kompilasi VHDL dengan syntax error.

VI. Penutup / Kesimpulan.

Dari berbagai tahap perancangan sistem digital dengan Altera UP2 CPLD Boards dan VHDL dapat diambil beberapa kesimpulan :

- Rancangan sistem-sistem digital dapat diimplementasikan dalam bentuk perangkat keras pada CPLD Altera FLEX 10K EPF10K70RC24-4

dengan memakai elemen logika yang sesuai dengan hasil rancangan.

- Complex Programmable Logic Device (CPLD) adalah salah satu teknologi programmable logic dimana operasional fungsi internal IC dapat didefinisikan sendiri oleh perancang (user) dengan terlebih dahulu memprogram operasi-operasi yang diinginkan.
- Perancangan sistem-sistem digital dapat dilakukan dengan menggunakan bahasa modular VHDL dimana awal perancangan dapat dilakukan dengan menggunakan skematik dan bahasa penjelasan perangkat keras (*Hardware Description Language*).
- Altera UP2 CPLD Boards sangat tepat digunakan untuk percobaan-percobaan perancangan sistem digital di Laboratorium.
- Altera MAX+PLUS II adalah tools Computer Aided Design (CAD) yang dapat digunakan untuk mendisain, mensimulasi, membangkitkan karakteristik waktu serta memuat dan menjalankan hasil disain pada Altera UP2 CPLD Boards.

Daftar Pustaka:

- [1] Altera, MAX+PLUS II Getting Started, Altera Corporation, 1997.
- [2] David Green, *Modern Logic Design*, Addison-Wesley Publishing, 1986.
- [3] David Pellerin and Douglas Taylor, *VHDL Made Easy*, Prentice Hall PTR, 1997
- [4] James O. Hamblen and Michael D. Furman, *Rapid Prototyping of Digital Systems : a Tutorial Approach*, Kluwer Academic Publishers, 2001
- [5] Louis Nashelsky, *Introduction to digital computer technology*, John Wiley & Sons Inc, 1977
- [6] Richard F. Tinker, *Digital Engineering Design: a Modern Approach*, Prentice-Hall, INC, 1991.
- [7] Zoran Salcic and Asim Smilagic, *Digital Systems Design and Prototyping Using Field Programmable Logic*, Kluwer Academic Publishers, 1997
- [8] <http://www.altera.com>