Field-Programmable Gate Arrays (FPGAs) untuk Otomasi Industri, Kendali, dan Robotika

Jazi Eko Istiyanto, Ph.D
Laboratorium Elektronika dan Instrumentasi
Jurusan Fisika FMIPA UGM
e-mail:jazi@ugm.ac.id
http://jazi.staff.ugm.ac.id

Krisis Design

Pada 1989, Tsugio Makimoto (penasehat korporat Sony) menyatakan bahwa aplikasi silikon utama akan berubah setiap 10 tahun (*mainstream silicon application is switching every 10 years*.) (http://www.kressarray.de). Kalau kita perhatikan sejarah dunia ”silikon” maka kita temukan bahwa sebelum 1957 tradisi silikon adalah perancangan ”custom”. Perioda 1957-1967 diwarnai dengan perancangan berbasis TTL (transistor-transistor logic). Kemudian 1967-1977 mulai ramai perancangan berbasis LSI(large-scale integration), dan MSI (medium-scale integration). Tahun 1997 menandai terjadinya krisis design yang pertama yang memunculkan teknologi mikroprosesor dan memori pada perioda 1977-1987. Tahun 1987-1997 berkembang rancangan ASIC(application-specific integrated circuit). Tahun 1997 merupakan krisis design yang kedua karena ASIC memerlukan fabrication foundries, memerlukan waktu cukup lama sementara tuntutan time-to-market makin memendek dan hanya ekonomis bila volume produksi di atas 10000 unit Tahun 1997-2007 adalah era reconfigurable computing yang muncul karena adanya teknologi field-programmable gate arrays. Kini, banyak kalangan ”silikon” sudah mulai yakin bahwa perioda 2007-2017 nanti adalah perioda system-on-a-chip(SOC) di mana teknologi sudah sedemikian maju sehingga dapat dirancang sebuah sistem besar di dalam sebuah chip.

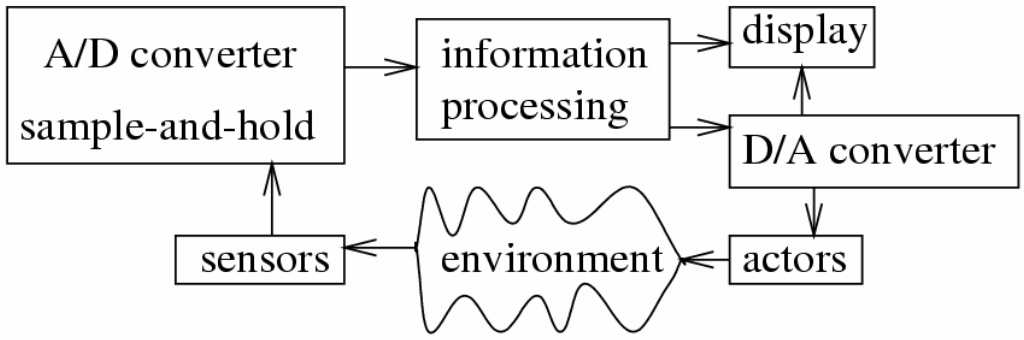
Sistem Instrumentasi dan Kendali

Sistem instrumentasi dan kendali dapat dilukiskan tersusun atas beberapa komponen (lihat gambar 1): (1) sensor (sering juga dinamakan transduser) adalah system yang mengukur besaran lingkungan (misalnya suhu, tekanan, kelembaban, dsb.) dan mengkonversikannya menjadi besaran listrik (misalnya LM35 mengkonversi suhu menjadi tegangan listrik). (2) analog-to-digital converter adalah system yang mengkonversikan besaran listrik analog menjadi besaran listrik digital (3) information processing adalah system yang memproses besaran digital menjadi besaran digital yang lain (4) display adalah system yang menampilkan suatu besaran digital, misalnya layar monitor, LCD display, 7-segment display, dsb. (5) digital-to- analog converter adalah system yang mengkonversikan besaran listrik digital menjadi besaran listrik analog (6) actor atau actuator adalah system yang mengkonversikan besaran listrik analog menjadi besaran lainnya misalnya kecepatan putaran, dsb.

Robotika

Robot didefinisikan sebagai sebuah automaton, yakni suatu piranti mekanik yang cerdas. Menurut Robotics Industry Association (1985), robot didefinisikan sebagai
“*A re-programmable, multi-functional manipulator designed to move material, parts, tools, or specialized devices for the performance of various tasks*” yakni suatu

manipulator banyak-fungsi yang dapat diprogram-ulang yang dirancang untuk memindahkan material, komponen, perkakas, atau piranti khusus untuk meningkatkan kinerja berbagai tugas. Robot juga didefinisikan sebagai *a machine able to extract information from its environment and use knowledge about its world to act safely in a meaningful and purposeful manner* (Ron Arkin, 1998), yakni sebuah mesin yang mampu mengekstrak informasi dari lingkunganny dan menggunakan pengetahuan tentang lingkungannya untuk beraksi secara selamat dengan cara yang sesuai yang diinginkan oleh pemrogrammnya.



Gambar 1 Blok diagram system instrumentasi dan kendali

Gambar 2 dan 3 adalah masing-masing robot buatan Sony dan Honda, yang sesuai dengan gambaran umum tentang robot. Gambar 4 adalah robot pelubang PCB yang memenuhi definisi robot menurut Robotics Industry Association, yakni meningkatkan kinerja berbagai tugas. Pelubang PCB (Sofyan dan Istiyanto, 2005) dikendalikan oleh computer yang membaca sebuah gambar PCB dan koordinat titik-titik yang akan dibor. Proses pengeboran dipercepat dengan mengimplementasikan algoritma nearest-neighbourhood heuristics (NNH), yakni suatu cara penyelesaian masalah traveling-salesman problem. Algoritma NNH dilukiskan pada listing 1.

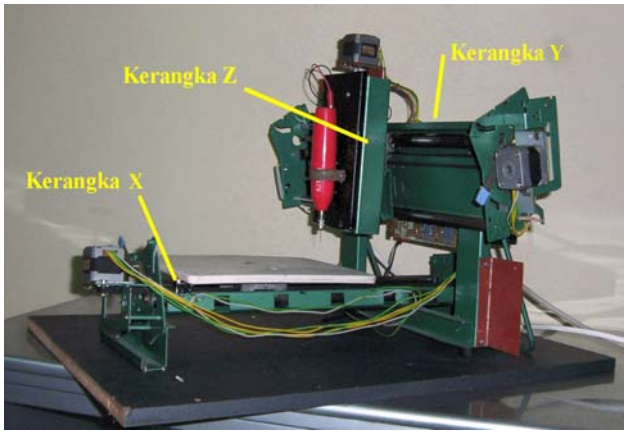
```
tour *tsp(int **A, int nrl, int nrh, int ncl, int nch){
  int i, bar, lastbar,  count, min;
  tour *t;

    count = 1;bar = 1; t = gen_tour(1); count++;
     while (count <= nrh) {
      min = 1;
     for (i = 1; i <= nch; i++) if (A[bar][min] < A[bar][i])  min = i;
     for (i= ncl; i <= nch; i++)
       if ((A[bar][min] > A[bar][i]) && (A[bar][i] != 0) && (in_tour(t, i)!= 1) )
            min = i;A[0][bar] = min;
            t= append_item(t, min);lastbar = bar;bar = min; count++;
    }
  t= append_item(t, 1);
  return t;
}
```

Listing 1. Algoritma NNH dalam GNU C

Gambar 2 Robot Aibo dari Sony

Gambar 3 Robot Asimo dari Honda



Gambar 4 Prototipe Bagian Mekanik Penggerak Mesin Pelubang PCB
(Sofyan dan Istiyanto, 2005)

Bila dicermati definisi-definisi robot di atas, maka gambar 1 yang merepresentasikan komponen-komponen suatu sistem instrumentasi dan kendali juga ditemukan pada robot.

Robotika adalah studi mengenai robot, yakni sistem berdikari (autonomous) yang berinteraksi dengan lingkungan fisika di sekitarnya. Oleh karena itu, walaupun serupa, sistem instrumentasi dan kendali hanya sebagian kecil saja dari studi robotika.

Robotika cakupannya lebih luas yakni menyangkut pula persepsi (bagaimana robot memandang suatu situasi), interaksi (bagaimana robot berinteraksi dengan lingkungan), dan aksi (bagaimana robot bertindak terhadap perubahan lingkungan).

FPGAs dan Programmable Logic Devices

Field-Programmable Gate Arrays (FPGAs) adalah salah satu piranti yang termasuk dalam kelompok programmable logic devices. FPGAs berbeda dari general-purpose mikroprosesor (misalnya Intel) dalam hal fleksibilitas logic-nya. Mikroprosesor mempunyai hardware yang tetap. Assembly programmer memprogram suatu komputasi dengan keterbatasan pada tetapnya banyaknya register, siklus fetch-decode-execute, serta fungsi-fungsi ALU(arithmetic and logic unit) dan pada banyaknya bit suatu register. FPGAs berbeda dari mikrokontroler (misalnya ATMEL), karena mikrokontroler pada prinsipnya adalah mikroprosesor yang diprogram dengan bahasa assembly dan dirancang sebagai pengendali bukan untuk komputasi. Mikroprosesor dan mikrokontroler mengimplementasikan suatu komputasi pada hardware yang tetap. Hardware pada FPGAs diserahkan sepenuhnya pada design engineer untuk memprogramnya. Sebelum diprogram, FPGAs hanyalah tersusun atas blok-blok yang belum dikonfigurasikan dan interkoneksi yang belum disusun dan difungsikan. Oleh karena itu, istilah yang lebih tepat adalah merekonfigurasikan FPGAs, bukan memprogramnya. Chip FPGAs yang sama dikonfigurasikan dengan data yang berbeda akan mengimplementasikan hardware yang berbeda.

FPGAs berbeda dari PLAs (Programmable Logic Arrays) maupun PALs(Programmable Array Logic). PLAs dan PALs tersusun atas larik gerbang OR dan larik gerbang AND. PLAs mengimplementasikan bentuk sum-of-product (SOP) dari logika kombinasi, sedangkan PALs mengimplementasikan bentuk product-of-sum (POS) dari logika kombinasi. PALs dan PLAs hanya dapat mengimplementasikan rangkaian kombinasi, FPGAs dapat mengimplementasikan rangkaian kombinasi maupun rangkaian sekuensial.

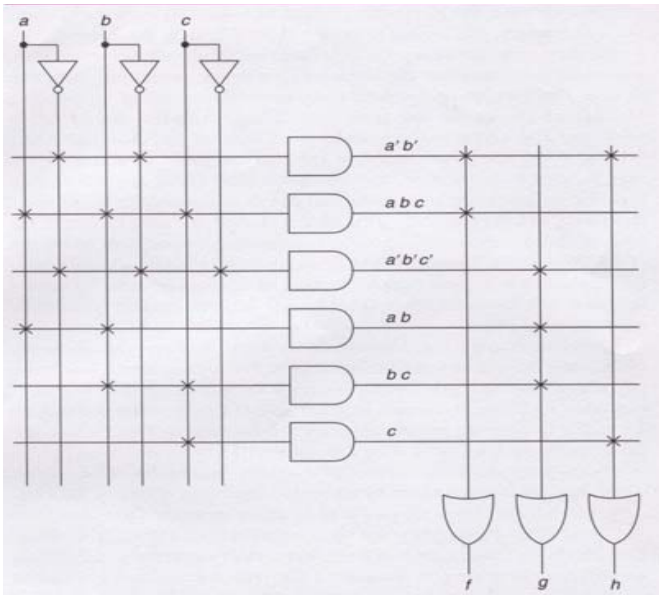
Rangkaian kombinasi dapat diimplementasikan menggunakan ROM (Random-Access Memory). ROM8x1 dapat mengimplementasikan sembarang fungsi Boole yang mempunyai sebuah output 1-bit dan mempunyai 3-input masing-masing 1-bit. Banyaknya variasi fungsi Boole yang dapat diimplementasikan dalam sebuah ROM8x1 adalah 2^2^3 = 256 buah. ROM juga termasuk programmable logic device tetapi khusus untuk rangkaian kombinasi. Kelemahan ROM adalah tingginya pemborosan silikon bila banyak isinya adalah logic 0. Misalnya bila diimplementasikan F = A.B.C pada ROM8x1, maka semua word pada ROM bernilai 0 kecuali pada address 7 (111 biner). PLAs dikembangkan untuk mengatasi pemborosan pada ROM ini. Namun PLAs mempunyai keterbatasan dibandingkan ROM.

Bila pada ROMkx1 sembarang fungsi Boole yang mempunyai sebuah output 1-bit dan mempunyai log2(k)-input masing-masing 1-bit dapat implemntasikan, maka pada PLAs hanya fungsi yang dapat diungkapkan dalam bentuk SOP menggunakan maksimum p suku saja yang dapat diimplementasikan. Suatu PLAs nxm dangan p suku hasil-kali tersusun atas p gerbang AND masing-masing 2n input dan m gerbang OR masing-masing p-input.

Tabel 1 ROM8x1 mengimplementasikan F=A.B.C

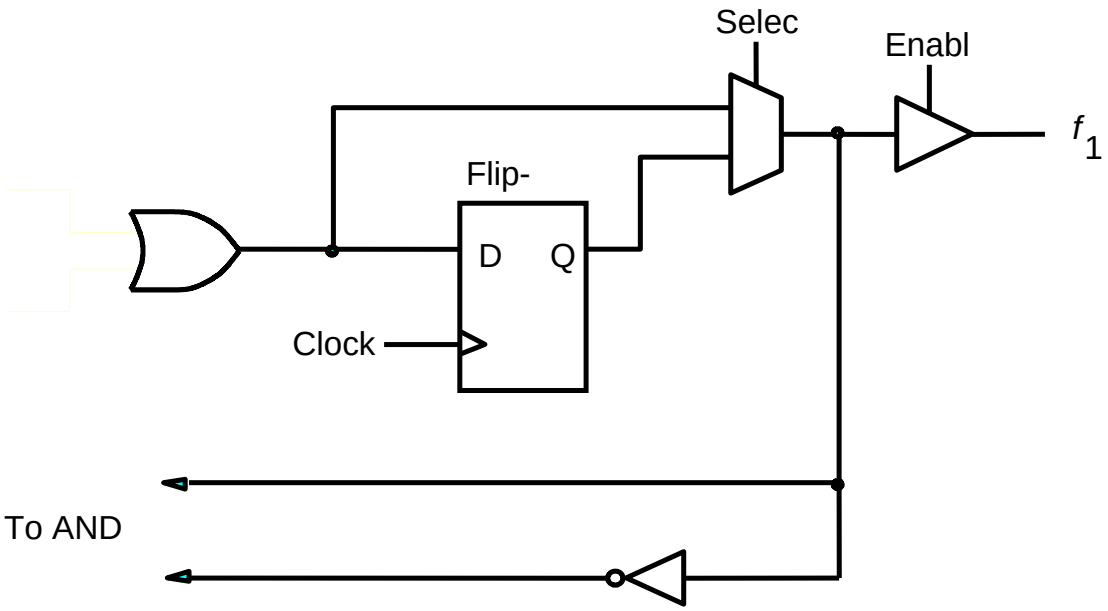
Address (desimal)	Address (biner)	Data ROM8x1
0	000	0
1	001	0
2	010	0
3	011	0
4	100	0
5	101	0
6	110	0
7	111	1

Gambar 5 melukiskan implementasi dari fungsi-fungsi f(a,b,c) = a'b' + abc, g(a,b,c) = a'b'c' + ab + bc, dan h(a,b,c) = c, pada PLA. Terlihat bagaimana suku-suku disusun dengan menghubungkan larik-larik AND, dan bentuk SOP disusun dengan menghubungkan larik-larik OR.



Gambar 5 Implementasi fungsi-fungsi f(a,b,c) = a'b' + abc, g(a,b,c) = a'b'c' + ab + bc, dan h(a,b,c) = c, pada PLA.

PLA dan PAL mempunyai keterbatasan dalam ketidakmampuannya mengimplementasikan rangkaian sekuensial. Oleh karena itu ditemukanlah CPLD (Complex Programmable Logic Device). Gambar 6 melukiskan arsitektur CPLD.

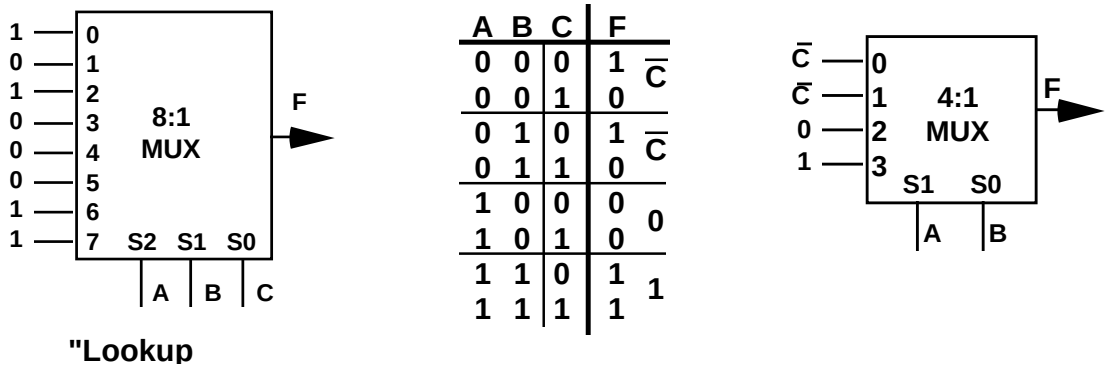


Gambar 6 Complex Programmable Logic Device Block

Multiplexer Sebagai General Purpose Logic Block

Andaikan akan diimplementasikan F(A,B,C) = m0 + m2 + m6 + m7 = A' B' C' + A' B C' + A B C' + A B C = A' B' (C') + A' B (C') + A B' (0) + A B (1). Tabel kebenaran, implementasi dalam MUX8-ke-1, dan implementasi dalam MUX4-ke-1 ditunjukkan pada Gambar 7. Terlihat bahwa MUX dapat mengimplementasikan embarang fungsi kombinasi. Karena F merupakan fungsi dari 3 variabel, maka bila hanya dipunyai MUX4-ke-1 harus dilakukan functional decomposition dahulu.

FPGAs produksi Actel menggunakan MUX sebagai building blocknya, sedangkan Xilinx menggunakan RAM



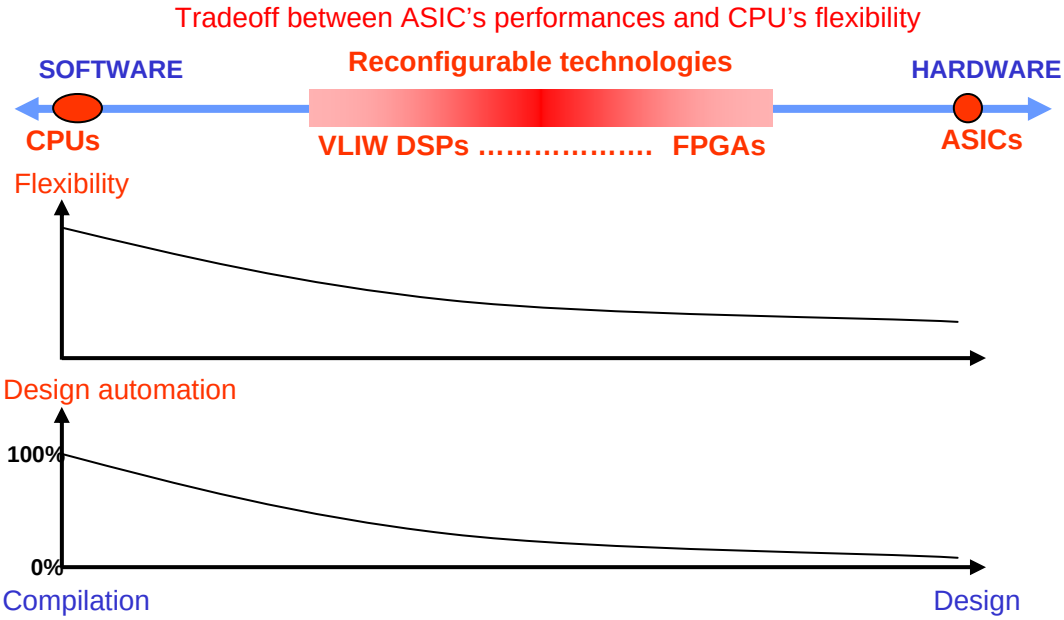
"Lookup

Gambar 7 Implementasi F(A,B,C) = m0 + m2 + m6 + m7 dalam MUX8-ke-1 dan MUX4-ke-1

FPGAs merupakan komponen utama suatu reconfigurable hardware. Apa itu reconfigurable hardware ?. Lihat Gambar 8. Reconfigurable hardware memanfaatkan trade-off antara tingginya kinerja full-hardware (ASIC-Application-Specific Interated Circuit) dengan tingginya fleksibilitas implementasi software pada CPU mikroprosesor. Reconfigurable teknologi didukung oleh FPGAs dan Digital Signal Processor. Gambar 9 dan 10 melukiskan dua buah produk komersial reconfigurable system.

What is Reconfigurable ?

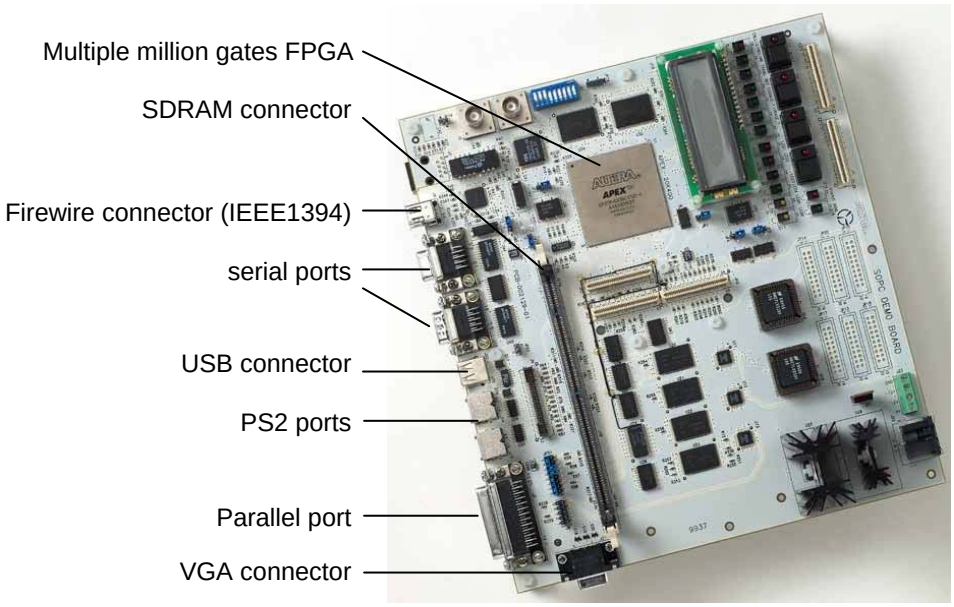
Somewhere between Hardware and Software



Gambar 8 Trade-off Reconfigurable Computing

Existing reconfigurable systems

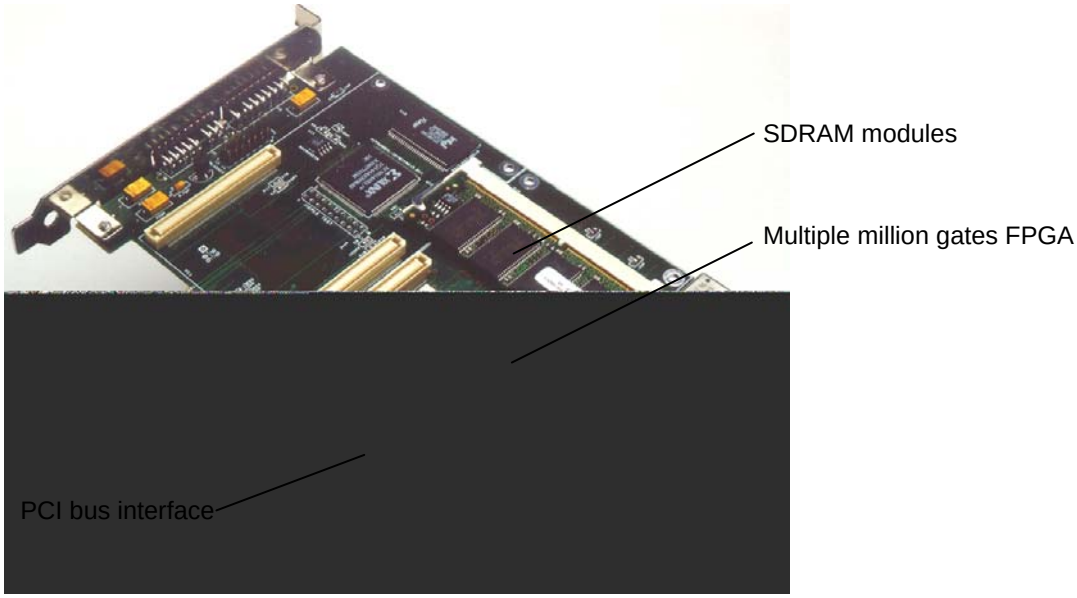
Altera solutions : SOPC (System on a Programmable Chip)



Gambar 9 Reconfigurable system dari Altera

Existing reconfigurable systems

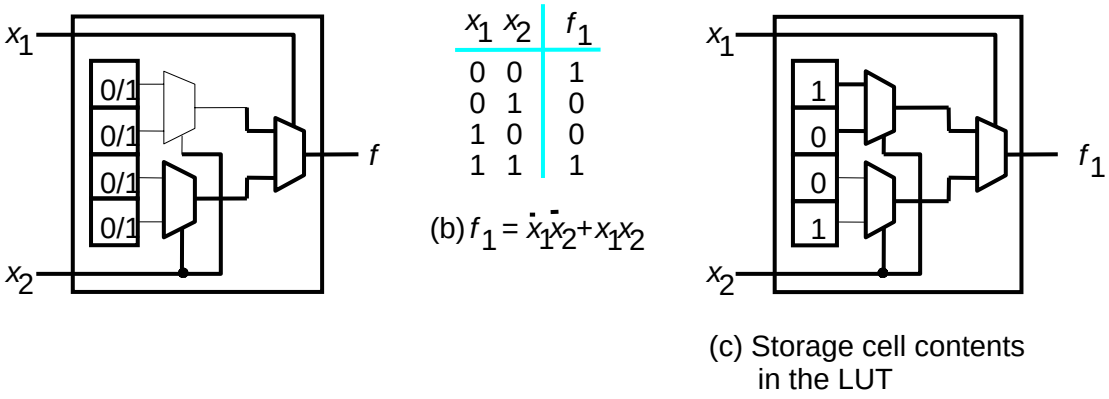
Xilinx solutions : PCI based FPGA boards



Gambar 10 Reconfigurable system dari Xilinx

Komponen Dasar FPGAs

FPGAs mempunyai komponen dasar yang dinamakan LUT (Look-Up Table) untuk menginplementasikan rangkaian kombinasi, register (flip-flop) untuk mengimplementasikan elemen penyimpan/rangkaian sekuensial, carry logic untuk fungsi aritmatika, dan expansion logic untuk mengimplementasikan fungsi yang tergantung pada lebih dari 4 input. Gambar 11 melukiskan LUT dengan dua buah input x1 dan x2 mengimplementasikan fungsi f1 = x1 xnor x2.



Gambar 11 LUT 2-input x1 dan x2 mengimplementasikan fungsi f1 = x1 xnor x2.

Gambar 12 melukiskan 2 buah LUT 4-input pada sebuah CLB(Combinational Logic Block) dari Xilinx Spartan IIE. Pada gambar 12 terlihat bahwa CLB selain memiliki dua buah LUT, juga ada dua buah fast carry logic dan dua buah D flip-flop.

Keberadaan flip-flop membuat CLB juga mampu mengimplementasikan rangkaian sekuensial. D Flip-flop yang bersifat edge-triggered juga cocok untuk implementasi rangkaian sekuensial dengan metoda penyandian one-hot encoding yang walaupun memerlukan lebih banyak flip-flop dibandingkan binary encoding, tetapi next-state logic dan output logic-nya lebih sederhana sehingga delay lebih rendah dan bisa diclock dengan frekuensi yang lebih tinggi.

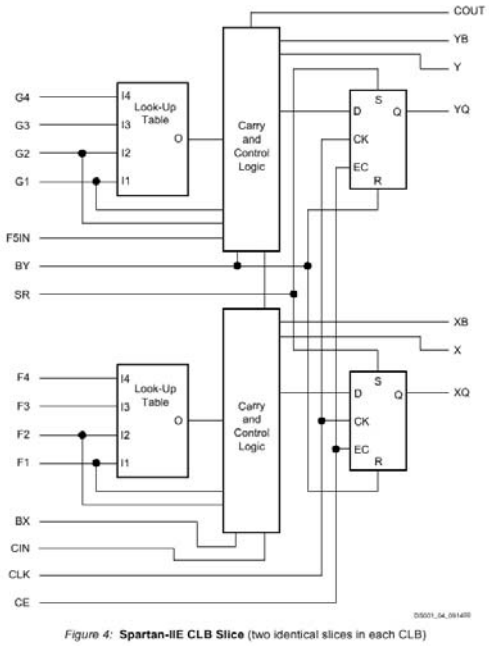


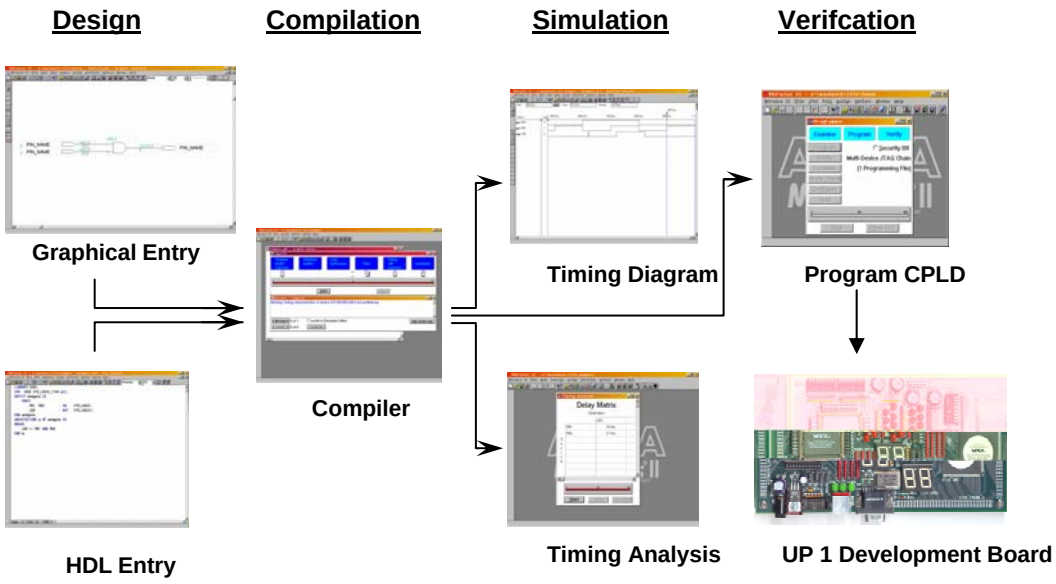
Figure 4: **Spartan-IIE CLB Slice** (two identical slices in each CLB)

Gambar 12 Internal sebuah CLB pada Xilinx Spartan IIE

Teknologi Rekonfigurasi FPGAs

Langkah-langkah merekonfigurasi FPGAs menggunakan contoh Altera digambarkan pada Gambar 13. Design engineer dapat memilih apakah menggunakan schematic capture (graphical entry) ataukah menggunakan HDL(hardware description language) entry. HDL entry mempunyai banyak keunggulan dibandingkan graphical entry, di antaranya :

1. pada HDL entry, modifikasi ukuran/banyaknya bit suatu register/bus jauh lebih mudah dibandingkan dengan graphical entry, karena pada graphical entry akan mengharuskan rancangan digambar ulang
2. design engineer pada HDL entry dapat memilih deskripsi behavioral yang memungkinan struktur hardware yang dirancang ditemukan sepenuhnya oleh software. Engineer cukup menuliskan deskripsi wataknya bukan bentuknya.
3. HDL entry lebih bersifat portabel karena bentuknya text, sehingga dapat dikompile dengan tool yang berlainan. Sementara itu untuk graphical entry, banyak tool berbeda format.
4. HDL entry lebih cocok untuk design team yang tersusun dari banyak engineer, karena lebih modular, dan lebih mudah mengkomunikasikan design antar engineers.

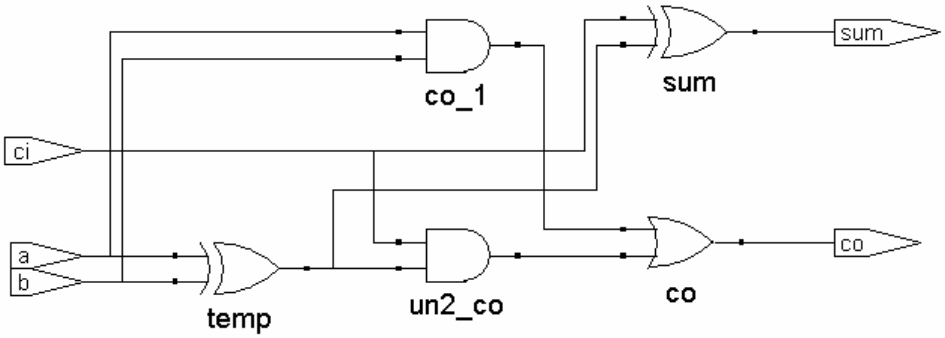


Gambar 13 Tahap-Tahap Perancangan FPGAs

Selanjutnya rancangan dikompilasikan. Sepertihalnya pada kompilasi software, disini dilakukan DRC(Design Rule Check), dan bila ada error akan dilaporkan. Design engineer kemudian dapat membetulkan error yang terjadi. Bila DRC tidak lagi memberikan pesan kesalahan, design engineer dapat mencek watak circuit yang akan dihasilkannya menggunakan timing diagram.

Design engineer juga dapat menguji clocking scheme dengan melakukan timing analysis untuk mengetahui delay-delay pada circuit dan berapa maksimum frekuensi clock. Setelah itu rancangan dapat diubah menjadi configuration bit-stream dan diload ke dalam device sesungguhnya.

Salah satu bahasa untuk HDL entry adalah VHDL (VHSIC Hardware desription Language ; VHSIC = Very High Speed Integrated Circuit). Gambar 14 melukiskan full adder dalam schematic entry. Gambar 15 melukiskan contoh design full adder berbasis VHDL.



Gambar 14 Full Adder dalam schematic entry

```
1   entity FullAdder is
2   port(
3       a  : in  bit;
4       b  : in  bit;
5       ci : in  bit;
6       sum: out bit;
7       co : out bit);
8   end entity;
9
10  architecture arch1 of FullAdder is
11      signal temp: bit;
12  begin
13      temp <= a xor b;
14      sum  <= temp xor ci;
15      co   <= (a and b) or (ci and temp);
16  end arch1;
```

Gambar 15 Full Adder ungkapkan dalam VHDL

State-of-the Art FPGAs

Saat ini teknologi fabrikasi FPGAs menggunakan teknologi proses 90 nm pada wafer 300 mm sehingga diperoleh biaya yang lebih rendah per fungsi (LUT dan register) serta transistor yang lebih kecil dan lebih cepat kinerjanya. FPGAs dapat mencapai system speed hingga 500 MHz. Kinerja ini diperoleh terutama karena diimplementasikannya smart interconnect, clock management, dedicated circuits, dan flexible I/O. Integrated transceiver dapat mencapai kinerja 10 Gigabits/sec. Selain itu FPGAs telah memiliki logic yang lebih banyak dan fitur yang lebih baik yakni di atas 100000 LUT dan flip-flop, serta di atas 200 buah embedded RAM, dan 18x18 multiplier. Teknologi saat ini memungkinkan FPGAs dipaket dalam chip dengan 1156 pin yang memiliki di atas 800 general-pupose I/O pin (50 I/O standard termasuk LVDS dengan terminasi internal) dan memiliki jalur global clock sebanyak 16 buah yang bersifat low-skew. FPGAs juga dapat memiliki on-chip microprocessor dan transceiver beberapa Gbps.

Keterbatasan FPGAs

FPGAs tidak dapat menggantikan mikroprosesor sepenuhnya, sepertihalnya Wireless LAN (IEEE 802.11) tidak akan menggantikan Wired LAN pada dunia computer network. FPGAs bersifat komplementer dengan mikroprosesor.

Otomasi, kendali, dan robotika mungkin menggunakan algoritma tertentu, misalnya jaringan syaraf buatan (ANN = artificial neural network). Pada ANN diperlukan adanya representasi floating-point untuk bobot koneksi ke suatu neuron. Misalnya IEEE Floating-point format 32-bit tersusun atas 1-bit tanda(S), 8-bit exponen (E) dalam sandi excess-127 dan 23-bit mantissa (M) dalam bentuk ternormalisir. Kemudian bobot neuron mesti dikenakan suatu operasi perkalian. Kalau diimplementasikan multiplier dalam hardware tentu diperlukan sumberdaya yang besar (perkalian 2 buah bilangan 32-bit memberikan maksimum 64-bit hasil perkalian). Kalau perkalian diimplementasikan dengan algoritma Booth, sumberdaya yang diperlukan berkurang tetapi kinerja turun.

Sementara itu, banyaknya neuron masih ditentukan pula oleh banyaknya hidden layer, demikian pula banyaknya koneksi dan bobot yang memasuki suatu neuron. Sehingga masih belum dimungkinkan mengimplementasikan ANN dalam sebuah FPGAs. Implementasi dalam banyak FPGAs masih merupakan persoalan terutama dalam hal

pemecahan (partisi) desain ke dalam beberapa FPGAs. Kalau partisi manual, mungkin banyak peluang optimasi yang terlewatkan. Kalau partisi otomatik, saat ini masih sulit dilakukan. Andrew, Niehaus, dan Ashenden (2004) mengindikasikan diperlukannya suatu programming model yang sesuai untuk implementasi hybrid CPU/FPGA.

FPGAs saat ini cocok untuk akselerator dari suatu bagian komputasi yang paling menyita sumberdaya komputasi. Bila bagian yang akan di-FPGA-kan kita pilih dengan baik, kinerja secara keseluruhan akan meningkat. Bidang otomasi, kendali, dan robotika akan menarik manfaat dari ketersediaan berbagai alternatif implementasi : mikroprosesor, mikrokontroler, PLC, FPGAs, BASIC-Stamp processor, dsb. Kurang bijaksana untuk sepenuhnya memikulkan seluruh beban kepada satu teknologi saja. Aspek yang terkait sangat banyak dan domain-nya sangat luas, sepertihalnya bidang ICT(Information and Communication Technology) itu sendiri bukan hanya berada di pundak ilmu komputer/informatika, elektronika, teknik elektro, fisika, matematika, dsb. tetapi pada semua bidang ilmu yang mungkin ada.

Penutup

FPGA sebagai suatu platform untuk implementasi sistem digital merupakan alternatif yang menarik dan berpotensi menjadi komponen utama masa depan. Banyak universitas di Indonesia yang sudah mulai mengajarkan (teori dan praktek) FPGA, termasuk prgram studi Elektronika dan Instrumentasi FMIPA UGM. Harapannya adalah sekalipun Indonesia lemah di sisi indutri komponen, paling tidak teknologi perancangan ”silikon ” kita kuasai.

Daftar Pustaka

Andrews, D., D. Niehaus, dan P. Ashenden, 2004: *Programming Models for Hybrid CPU/FPGA Chips*, IEEE Computer pp 118-120. January

Sofyan, dan Jazi Eko Istiyanto, 2005 : *Pembuatan Prototipe Mesin Pelubang PCB Berbasis Komputer dan Optimasi Kinerjanya Menggunakan Algoritma NNH*, SNIKTI2005, Universitas Kristen Satya Wacana, Salatiga (diterima untuk publikasi).

http://www.kressary.de.

Biodata

Jazi Eko Istiyanto memperoleh gelar sarjana ilmu fisika (1986) dari Universitas Gadjah Mada, Yogyakarta, Indonesia. Kemudian ia mengikuti studi lanjut di Inggris dan memperoleh **Postgraduate Diploma in Computer Programming and Microprocessor Applications** (1987), **M.Sc in Computer Science** (1988), dan **Ph.D in Electronic Systems Engineering** (1995), semuanya dari University of Essex. Kini ia adalah Ketua Program Studi Elektronika dan Instrumentasi FMIPA UGM, dan mengajar pada Program Magister Ilmu Komputer (mk. **Sains Manajemen Sistem Elektronik** dan mk. **Network Security**), Program Magister Ilmu Fisika (mk. **Sistem Akuisisi Data**), Program Magister Teknik Elektro (mk. **Asas Perancangan Sistem Elektronik**) dan Program Magister Managemen (mk. **Introduction to Information Systems)** Universitas Gadjah Mada. Juga pernah mengajar Program Magister Sains Ilmu Ekonomi UGM (mk. **Metoda Kuantitatif untuk Managemen**, 2000-2004). Minat risetnya meliputi keamanan informasi, layanan web, dan sistem tertanam (*embedded systems*). Dr. Jazi adalah anggota Himpunan Fisika Indonesia dan Persatuan Insinyur Indonesia.