

Optimasi Area Rangkaian Terintegrasi pada Router Jaringan On-Chip dengan Konsep Arsitektur Modular

Faizal Arya Samman

Universitas Hasanuddin, Fakultas Teknik, Jurusan Teknik Elektro

Kampus Tamalanrea, Jl. Perintis Kemerdekaan Km. 10

Makassar 90245

Email: faizalas@unhas.ac.id

Abstrak—Paper ini menunjukkan langkah-langkah optimasi struktur interkoneksi input-output (IO) pada router-router yang digunakan pada jaringan on-chip dalam aplikasi sistem prosesor multicore. Hasil optimasi ini akan memperkecil area rangkaian terintegrasi sehingga berpotensi menurunkan konsumsi daya statik dari rangkaian terintegrasi router jaringan on-chip. Dengan menggunakan kaidah aturan belokan aliran data (turn model) yang berlaku pada sebuah algoritma routing deterministik maupun algoritma routing adaptif, maka beberapa jalur data dan jalur-jalur sinyal kendali pada struktur interkoneksi IO dalam router dapat dihapuskan. Penghapusan jalur-jalur data dan sinyal kendali ini tidak mempengaruhi fungsionalitas dari router. Hasil optimasi menunjukkan bahwa router dengan algoritma routing deterministik dapat dioptimasi hingga mencapai efisiensi sebesar 21%, dimana proses sintesis rangkaian logika dari router tersebut menggunakan teknologi sel-sel logika standar (standard-cell technology) dengan fitur transistor CMOS (Complementary Metal Oxide Silicon) 130 nm.

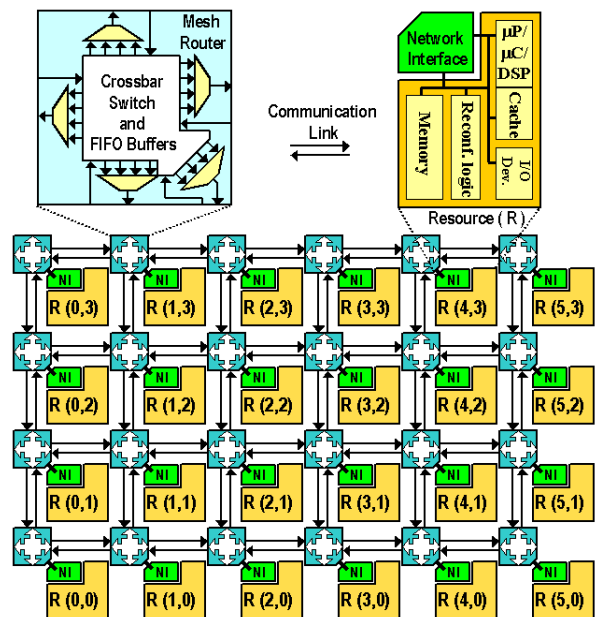
Kata-kata Kunci—Jaringan On-Chip, Testbench, Bahasa Deskripsi Perangkat Keras (VHDL), Algoritma Routing, Rangkaian Terintegrasi.

I. PENDAHULUAN

Jaringan On-Chip atau dalam istilah asing dikenal sebagai *Network-on-Chip* telah menjadi topik menarik dalam dunia teknologi informasi dewasa ini. Perkembangan topik ini didorong oleh perkembangan teknologi komputer mikro yang terus berkembang dari waktu ke waktu.

Perkembangan perangkat-perangkat lunak aplikasi telah memicu kebutuhan perangkat-perangkat keras komputer dengan kinerja yang lebih tinggi. Kemajuan teknologi semikonduktor dewasa ini telah memungkinkan para ilmuwan dalam merancang transistor-transistor dengan ukuran fitur yang lebih kecil. Dengan ukuran yang lebih kecil tersebut, maka komputer mikro dapat dirancang sedemikian rupa hingga mampu dijalankan dengan detak frekuensi yang lebih cepat, sehingga melahirkan komputer mikro dengan kinerja yang lebih tinggi.

Namun demikian, peningkatan detak frekuensi dapat meningkatkan disipasi daya yang lebih besar sehingga chip-komputer menjadi lebih cepat panas. Akibatnya efek negatif seperti konsumsi energi listrik yang lebih besar akan timbul. Siklus usia operasi komputer mikro juga bisa menjadi lebih pendek, kecuali jika komputer mikro dilengkapi dengan



Gambar 1. NoC in a 2-D 6x4 mesh topology.

sistem pendingin atau penyerap panas (*heat sink*) yang lebih memadai.

Untuk mengatasi hal tersebut, salah satu solusi yang cukup menarik digunakan dalam meningkatkan kinerja sebuah komputer mikro adalah dengan meningkatkan jumlah prosesor pekerja dalam sebuah chip komputer mikro. Dengan jumlah prosesor yang lebih besar, maka aplikasi-aplikasi yang membutuhkan kinerja tinggi dapat dieksekusi dengan baik tanpa harus meningkatkan detak frekuensi kerja komputer mikro. Sebuah aplikasi akan diurai ke dalam beberapa bagian tugas (*tasks*), dimana bagian-bagian tugas yang diurai tersebut didelagasikan kepada prosesor-prosesor untuk selanjutnya bersama-sama menyelesaikan seluruh tugas dari aplikasi tadi. Karena dikerjakan bersama-sama, maka durasi waktu eksekusi aplikasi akan menjadi lebih singkat tanpa harus meningkatkan detak frekuensi kerja komputer mikro tadi.

Bila jumlah core prosesor meningkat cukup signifikan, maka sistem komputer mikro membutuhkan sebuah infrastruktur

komunikasi data yang memadai. Sistem komunikasi bus, hanya memberikan kinerja yang baik bila prosesor berjumlah kurang dari 16 core. Lebih dari itu, sistem bus akan mengalami problem kemacetan kinerja (*bottleneck performance problem*). Kinerja buruk ini disebabkan oleh jalur komunikasi yang tidak seimbang dengan jumlah core prosesor. Oleh karena itu, dibutuhkan sebuah infrastruktur komunikasi yang bisa memberikan kapasitas pita komunikasi yang lebih lebar (*wider bandwidth capacity*). Salah satu solusi yang dapat ditawarkan adalah dengan cara membangun jaringan router yang terinterkoneksi dalam chip tunggal bersama-sama dengan core-core prosesor.

Gambar 1 memperlihatkan sebuah jaringan on-chip dengan topologi mesh dengan ukuran 6×4 . Setiap router (*mesh router*) terhubung dengan sebuah prosesor core (*Resource Processor*) melalui sebuah antarmuka jaringan (*Network Interface*). Setiap prosesor memiliki alamat tujuan (x, y) sebagaimana terlihat pada gambar. Router memiliki struktur saklar crossbar (*crossbar switch*) yang memiliki empat buah port-port masukan-keluaran yang terhubung dengan router-router terdekat (*neighbor routers*), dan sebuah port lokal yang terhubung ke sebuah core prosesor melalui sebuah antarmuka jaringan. Topologi jaringan mesh sangat disukai disebabkan oleh kapasitas *bandwidth*-nya yang bersifat *scalable*, yaitu kapasitas *bandwidth* komunikasi akan meningkat bila ukuran jaringan ditingkatkan seiring peningkatan jumlah core-core prosesor perkerja. Struktur jaringan mesh juga terlihat sederhana sehingga implementasi algoritma routing menjadi lebih mudah.

Beberapa industri dan lembaga penelitian di dunia telah memulai upaya dalam merancang chip komputer mikro yang terdiri dari beberapa core prosesor. Intel Teraflops [2] misalnya telah merilis sebuah chip multiprosesor yang terdiri dari 80 prosesor MAC (Multiply-Accumulator) yang terhubung satu sama lain melalui jaringan on-chip dengan topologi Mesh berukuran 8×10 . Jaringan on-chip dengan sistem prosesor multicore yang lain juga telah diperkenalkan, di antaranya TRIPS chip [1], SCC NoC [3] dan STNoC [4] buatan STMicroelectronics.

Paper ini secara khusus akan memaparkan salah satu isu mengenai ukuran area rangkaian terintegrasi sebuah router on-chip. Isu ini sangat penting, karena akan memberikan dampak pada penurunan disipasi daya statik dari router tersebut. Sebagai contoh kasus, optimasi area rangkaian terintegrasi akan dilakukan pada sebuah prototipe router on-chip. Bagian II akan memaparkan secara singkat arsitektur modular dari prototipe router tersebut. Selanjutnya Bagian III akan menjelaskan kontribusi baru dari paper ini.

II. ARSITEKTUR MODULAR

Pada bagian ini akan diperkenalkan sebuah router jaringan on-chip yang didesain berdasarkan arsitektur modular. Router ini dinamakan **XHiNoC** yang memiliki singkatan dari **Extendable Hierarchical Network-on-Chip**. Salah satu karakteristik menarik dari XHiNoC adalah dukungan layanan komunikasi multicast yang bebas dari *deadlock* [5], [7], [10].

Format paket data yang digunakan dalam jaringan on-chip XHiNoC juga sangat unik. Setiap paket data dibagi ke dalam

sebuah flit. Pada setiap flit, disisipkan dua extra bit kendali, yaitu Label ID (Identitas) dan label tipe flit. Tipe flit terdiri atas flit bertipe header, data payload dan flit ekor (*tail flit*). Dengan menggunakan format paket tersebut, XHiNoC mampu menumpang tindih (*interleaving*) setiap paket pada level flit sehingga paket-paket yang berbeda dapat bercampur. Namun demikian, proses pencampuran dapat dikendalikan dengan baik oleh karena proses pencampuran (*multiplexing*) dilakukan dengan menerapkan aturan bahwa flit-flit dari paket yang sama akan memiliki label ID yang sama pada setiap link. Dengan teknik tersebut maka sebuah metode switching yang baru dapat diperkenalkan yaitu *wormhole cut-through switching* [9].

Ulasan lebih jauh tentang XHiNoC dan karakteristik kinerjanya dapat dilihat lebih jelas dalam referensi [6], [9] dan [8]. Bagian ini hanya mencoba mengulas secara singkat arsitektur XHiNoC secara umum dan fungsi-fungsi modul-modul utamanya secara singkat.

Tipikal arsitektur XHiNoC dengan N buah port masukan-keluaran ditunjukkan pada Gambar 2. Router XHiNoC terdiri dari 4 buah komponen utama, yaitu: FIFO (*First-In First-Out*) buffer, modul REB (*Routing Engine with Input Buffer*), modul Arbiter dan modul MIM (*Multiplexor with ID Label Management Unit*). Tiap-tiap modul saling terhubung dengan N buah jalur data $(1, 2, \dots, N)$, dimana N adalah jumlah port IO dalam router. Selain itu, modul-modul juga saling terhubung dengan jalur-jalur sinyal kendali, yaitu sinyal arbitrase $a(p, q)$ dan sinyal routing $r(p, q)$, dimana $p, q \in 1, 2, \dots, N$. Paper ini akan mengevaluasi jaringan on-chip dengan topologi mesh. Oleh karena itu, jumlah port dalam router adalah $N = 5$.

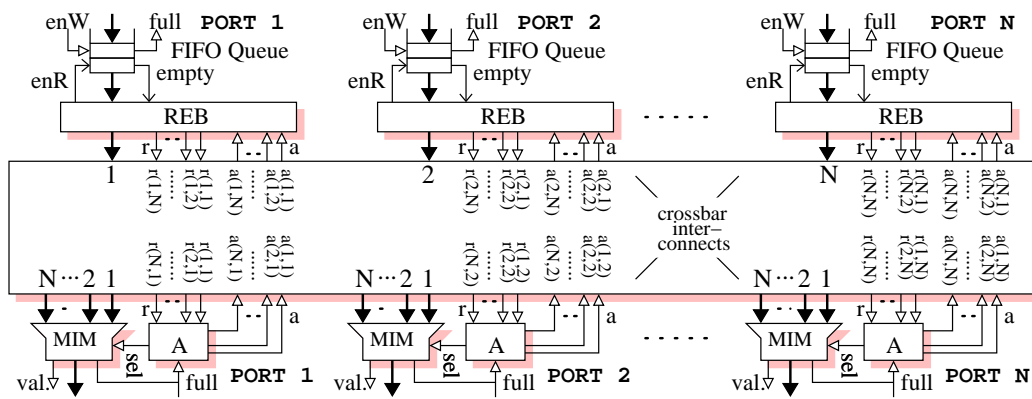
Modul FIFO buffer dan REB terletak pada sisi port masukan, sedangkan modul Arbiter dan MIM terletak pada sisi port keluaran. Keempat modul memiliki fungsi yang dijelaskan secara singkat pada sub-sub berikut ini.

A. Modul FIFO buffer

Modul ini berfungsi menyimpan sementara data yang masuk ke sebuah port masukan sebelum dilakukan proses routing terhadap data tersebut. Mode penyimpanan data menggunakan prinsip *first-in first-out*, atau data yang pertama masuk ke buffer, menjadi data yang pertama keluar dari buffer. Oleh karena itu, buffer ini disebut *First-In First-Out Buffer* atau disingkat FIFO buffer. FIFO buffer memiliki slot-slot data yang berjumlah minimal 2 buah. Semakin besar jumlah slot, semakin besar jumlah data yang dapat disimpan dalam buffer tersebut.

B. Modul REB

Modul ini berfungsi melakukan proses routing sehingga paket dapat diteruskan ke port keluaran yang diinginkan hingga akhirnya paket tersebut mampu mencapai target router. Modul ini juga dilengkapi dengan sebuah buffer yang digunakan untuk menyimpan flit data yang sedang dirouting. Mesin routing dalam XHiNoC terdiri dari sebuah tabel routing yang dapat terprogram secara *runtime* dan sebuah mesin algoritma routing, tempat dimana sebuah routing algoritma diimplementasi.



Gambar 2. Tipikal arsitektur modular dari Router Jaringan On-Chip dengan 5 modul utama, yaitu FIFO buffer, REB, Arbiter dan MIM unit.

C. Modul Arbiter

Modul arbiter berfungsi melakukan proses seleksi data-data flit dari port-port masukan yang menginginkan proses switching ke sebuah port keluaran tertentu. Setiap modul Arbiter pada port keluaran hanya bisa memilih satu dari sekian data flit yang menginginkan proses switching ke sebuah port keluaran. Pada Router XHiNoC, digunakan mekanisme arbitrase berputar flit-per-flit (*flit-by-flit round arbitration* [8]).

D. Modul MIM

Modul ini berfungsi sebagai multiplexor yang menghubungkan port-port masukan dengan port-port keluaran. Pada modul ini juga terdapat sebuah unit yang sangat penting untuk mengimplementasikan metode *wormhole cut-through switching* yang pertama kali diperkenalkan dalam router XHiNoC, yaitu unit manajemen ID-tag (*ID-Tag Management Unit*). Unit manajemen ID inilah yang berfungsi mengatur Label ID setiap flit sedemikian hingga flit-flit yang tergolong ke dalam paket yang sama akan memiliki Label ID yang sama.

III. KONTRIBUSI

Kontribusi baru yang diberikan pada paper ini adalah berupa rangkaian upaya untuk mengoptimalkan area rangkaian terintegrasi dari sebuah jaringan on-chip berdasarkan algoritma routing yang digunakan untuk merouting paket-paket data dalam jaringan on-chip tersebut. Dalam paper ini, kita akan memilih studi kasus untuk jaringan on-chip XHiNoC yang telah diperkenalkan pada bagian sebelumnya.

Pada publikasi sebelumnya [9], router on-chip XHiNoC dirancang dengan menggunakan struktur interkoneksi IO penuh. Artinya, semua port-port masukan terhubung ke semua port-port keluaran. Bila sebuah algoritma routing diimplementasikan ke dalam router, maka akan terdapat beberapa jalur data yang tidak pernah terpakai. Oleh karena itu, optimasi area rangkaian dapat dilakukan dengan cara menghilangkan beberapa jalur yang tidak digunakan dari saklar crossbar (*crossbar switch*). Penghapusan ini pada akhirnya akan menghasilkan protipe router yang baru dengan jumlah sel-sel gerbang logika

menjadi lebih kecil. Oleh karenanya, area rangkaian terintegrasi dapat dibuat lebih kecil tanpa mempengaruhi fungsi-fungsi routing pada router tersebut.

Bagian-bagian berikut ini akan menjelaskan secara rinci langkah-langkah optimasi area rangkaian terintegrasi dari router XHiNoC.

IV. OPTIMASI AREA RANGKAIAN TERINTEGRASI

A. Optimasi Jalur Data dengan Putaran Balik

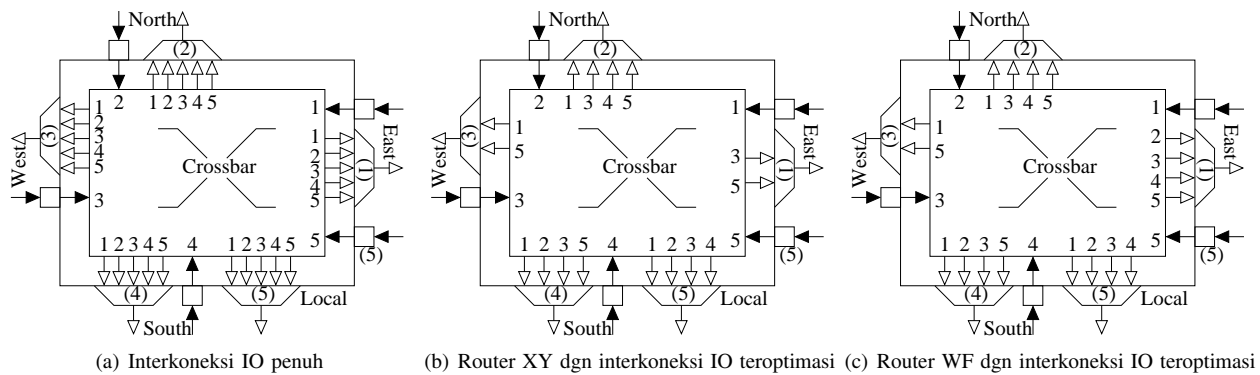
Bila kita mengasumsi bahwa trafik data tidak akan melakukan putaran balik (*backtrace routing*), yaitu trafik data (paket data) tidak akan berbalik arah putaran atau berputar dari port masukan ke port keluaran yang sama, maka beberapa jalur interkoneksi IO untuk data pada saklar crossbar dapat dihilangkan. Bila terdapat N buah pasangan port-port masukan-keluaran (IO), maka akan terdapat port-port masukan $P_j, j \in \{1, 2, \dots, N\}$ dan port-port keluaran $P_k, k \in \{1, 2, \dots, N\}$. Sehingga, jalur-jalur data dengan putaran balik, yaitu dari port masukan P_j ke port keluaran P_k dimana $k = j$, dapat dihapuskan dari struktur interkoneksi saklar crossbar. Untuk N buah pasangan port-port IO, maka kita bisa menghilangkan sejumlah N buah jalur data putaran balik termasuk juga jalur-jalur sinyal kendalinya.

Bila kita meninjau kembali arsitektur router jaringan on-chip XHiNoC sebagaimana terlihat dalam Gambar 2, maka jalur-jalur data d_i dihilangkan dari port masukan dari modul MIM yang terletak pada port keluaran $Port i$. Sebagai konsekuensinya, jalur-jalur sinyal kendali $r_{i,i}$ dan $a_{i,i}$ juga dihilangkan dari port-port masukan-keluaran modul-modul REB dan Arbiter.

B. Optimasi Jalur Data dari Belokan Terlarang

Proses optimisasi pada bagian ini dibuat berdasarkan jenis fungsi routing yang akan digunakan untuk mengimplementasi router jaringan on-chip. Optimasi dilakukan dengan memperhatikan arah jalur belokan-belokan aliran data yang dilarang pada fungsi routing yang digunakan tadi.

Sebelum sampai ke pembahasan lebih jauh kita akan membahas model belokan data dari dua buah algoritma routing yang umum digunakan untuk merouting data dalam sebuah



Gambar 3. Struktur dari Crossbar switch untuk router dengan interkoneksi IO penuh dan teroptimasi.

jaringan komunikasi. Kedua fungsi routing tersebut adalah algoritma routing deterministik atau statik dan algoritma routing adaptif *West-First* (Routing WF). Salah satu algoritma routing statik/deterministik yang sangat terkenal adalah 'Algoritma Routing XY'. Pada algoritma XY paket data selalu dirouting pertama kali ke arah horizontal (X) kemudian ke arah vertikal (Y). Oleh karena itu, algoritma ini disebut Algoritma Routing XY. Dalam fungsi routing XY, arah-arah belokan data *North-East*, *North-West*, *South-East* dan *South-West* dilarang diterapkan karena melanggar kaidah aturan algoritma. Dalam paper ini, kita menetapkan nomor-nomor dari port-port *East*, *North*, *West*, *South* dan *Local* masing-masing dengan urutan berikut, yaitu *Port 1*, *Port 2*, *Port 3*, *Port 4* dan *Port 5*. Penetapan nomor port ini juga telah ditunjukkan pada arsitektur router jaringan on-chip XHiNoC seperti terlihat pada Gambar 2.

Berdasarkan gambar arsitektur dari router XHiNoC dan aturan-aturan atau model belokan jalur data pada port-port *Port 1* dan *Port 3*, jalur data masukan d_2 dan d_4 yang terhubung ke modul *MIM* pada kedua port dapat dihilangkan dari struktur interkoneksi saklar crossbar. Sebagai konsekuensinya, jalur-jalur sinyal kendali $r_{2,1}$, $r_{2,3}$, $r_{4,1}$ dan $r_{4,3}$ begitu pula dengan jalur-jalur sinyal kendali $a_{2,1}$, $a_{2,3}$, $a_{4,1}$ dan $a_{4,3}$ dihilangkan dari port-port modul *REB* yang terletak pada *Port 2* dan *Port 4*. Jalur-jalur sinyal kendali port-port modul *Arbiter* yang terletak pada *Port 1* dan *Port 3* juga dihilangkan.

Secara umum dapat disimpulkan bahwa, jika belokan aliran data dari *Port n* ke *Port m*, dimana $n, m \in \{1, 2, \dots, N\}$, dilarang pada fungsi routing tertentu, maka jalur data masukan d_n dari sebuah mesin routing, atau *routing engine*, (*RE module*) yang terhubung ke modul *MIM* pada port keluaran *Port m*, termasuk juga jalur-jalur sinyal kendalinya, yaitu $r_{n,m}$ dan $a_{n,m}$, dapat dihilangkan dari struktur interkoneksi saklar (*switch*) crossbar-nya.

Gambar 3 menunjukkan tiga bentuk susunan saklar crossbar dari tiga prototipe router dengan tipe interkoneksi IO yang berbeda. Gambar 3(a) memperlihatkan structure saklar crossbar dari router dengan interkoneksi IO penuh (*fully IO interconnects*). Hasil optimasi jumlah interkoneksi IO dari struktur router untuk Router XY dengan algoritma routing deterministik ditunjukkan pada Gambar 3(b)). Sedangkan hasil optimasi

interkoneksi IO untuk Router WF dengan algoritma routing WF adaptif minimal diperlihatkan pada Gambar 3(c)).

Sebagaimana terlihat pada Gambar 3(b), jalur-jalur data dari port masukan *North* (2) ke port keluaran *East* (1) dan ke port keluaran *West* (3) dihilangkan dari struktur saklar crossbar. Jalur-jalur data lain yang dihilangkan adalah jalur-jalur data dari port masukan *South* (4) ke port keluaran *East* (1) dan port keluaran *West* (3). Penghapusan jalur-jalur data tersebut dari interkoneksi saklar crossbar dapat dilakukan, karena dengan menggunakan algoritma routing statik atau deterministik, jalur-jalur belokan aliran data *North-East*, *North-West*, *South-East* dan *South-West* dilarang diterapkan. Oleh karena itu, jalur-jalur tersebut tidak akan pernah terpakai sehingga dapat dihapuskan dari saklar crossbar.

V. HASIL SINTESIS DAN LAYOUT RANGKAIAN TERINTEGRASI

A. Hasil Sintesis

Router dari Jaringan on-chip telah dimodelkan ke dalam model Bahasa Deskripsi Perangkat Keras (VHDL - *Very High-Speed IC Hardware Description Language*). Model VHDL tersebut kemudian disintesis secara otomatis menggunakan software sintesis dengan menggunakan teknologi sel-standard CMOS (*CMOS standard-cell technology*) buatan *UMC* sehingga diperoleh rangkaian digital dengan komponen-komponen sel-sel gerbang logika dari teknologi tersebut. Proses sintesis dilakukan menggunakan software sintesis rangkaian logika buatan *Synopsys*. Hasil sintesis dari router-router jaringan on-chip diperlihatkan pada Tabel I.

Pada Tabel I terlihat hasil sintesis dari modul-modul utama dari router-router jaringan on-chip yang ditunjukkan pada tiap-tiap baris table dari kolom pertama dari tabel tersebut. Empat komponen utama dari router jaringan on-chip adalah FIFO buffer, *REB*, *Arbiter* dan *MIM* modul. Tiga kolom berikutnya menunjukkan data-data hasil sintesis dari masing-masing 3 prototipe router, yaitu Router WF yang menggunakan algoritma routing adaptif *West-First* dengan interkoneksi IO penuh, Router XY yang menggunakan algoritma routing deterministik dengan interkoneksi IO penuh, serta Router XY dengan interkoneksi IO teroptimasi. Kolom terakhir dari tabel tersebut menunjukkan data efisiensi hasil

TABEL I
HASIL SINTESIS DARI ROUTER-ROUTER MENGGUNAKAN TEKNOLOGI SEL-STANDARD CMOS (CMOS STANDARD-CELL TECHNOLOGY) 130-NM
DENGAN TARGET FREKUENSI DETAK 1.1 GHz ATAU DENGAN PERIODA DETAK 0.9 ns.

Komponen Router	Router WF dgn interkoneksi IO penuh (mm^2)	Router XY dgn interkoneksi IO penuh (mm^2)	Router XY dgn interkoneksi IO teroptimasi (mm^2)	Efisiensi: Opt./Penuh dgn XY routing
FIFO buffer (E)	0.003419	0.003390	0.003274	3.4 %
FIFO buffer (L)	0.003343	0.003376	0.003276	2.9 %
FIFO buffer (N)	0.003375	0.003382	0.003254	3.9 %
FIFO buffer (S)	0.003357	0.003387	0.003250	4.0 %
FIFO buffer (W)	0.003343	0.003374	0.003263	3.3 %
Total 5 FIFO buffers % dari Total area sel	0.016837 14.87 %	0.016909 16.0 %	0.016317 19.3 %	3.5 %
Arbiter (E)	0.001571	0.001517	0.000195	87.2 %
Arbiter (L)	0.001588	0.001518	0.000549	63.8 %
Arbiter (N)	0.001611	0.001518	0.000580	59.9 %
Arbiter (S)	0.001659	0.001445	0.000587	61.9 %
Arbiter (W)	0.001642	0.001469	0.000201	86.3 %
Total 5 Arbiters % dari Total area sel	0.008071 7.13 %	0.007489 7.0 %	0.002112 2.4 %	71.8 %
MIM (E)	0.011644	0.010969	0.008345	23.9 %
MIM (L)	0.010895	0.011592	0.009996	13.8 %
MIM (N)	0.010951	0.011153	0.010139	9.1 %
MIM (S)	0.011518	0.011300	0.009873	11.1 %
MIM (W)	0.011464	0.011230	0.008433	24.9 %
Total 5 MIMs % dari Total area sel	0.056472 49.88 %	0.056224 53.0 %	0.046786 55.7 %	16.8 %
REB (E)	0.006404	0.005162	0.004140	19.8 %
REB (L)	0.006325	0.005175	0.004391	15.1 %
REB (N)	0.006378	0.005157	0.002953	42.7 %
REB (S)	0.006316	0.005155	0.002953	42.7 %
REB (W)	0.006411	0.005161	0.004626	10.4 %
Total 5 REBs % dari Total area sel	0.031834 28.12 %	0.025810 24.0 %	0.019063 22.6 %	26.1 %
Total area sel	0.113214	0.105877	0.083703	20.9 %

implementasi antara Router XY dengan interkoneksi penuh dengan Router XY yang lain dengan interkoneksi teroptimasi.

Dari Tabel I terlihat bahwa efisiensi tertinggi dari segi persentasi diperoleh dari hasil optimasi *Arbiter*, yaitu sekitar 71.8%. Namun demikian dari segi area sel-sel logika, efisiensi tertinggi diperoleh dari hasil optimasi modul *MIM*, yaitu sekitar $0.01 mm^2$. Hasil ini diperoleh dari selisih $0.056224 mm^2 - 0.046786 mm^2$. Modul FIFO buffer tidak memberikan efisiensi yang berarti disebabkan oleh metode optimasi yang telah dilakukan tidak secara langsung menyentuh jalur-jalur sinyal kendali dan data dari modul-modul FIFO buffer. Secara umum hasil optimasi router jaringan on-chip memberikan total efisiensi area sel-sel logika sebesar 21%.

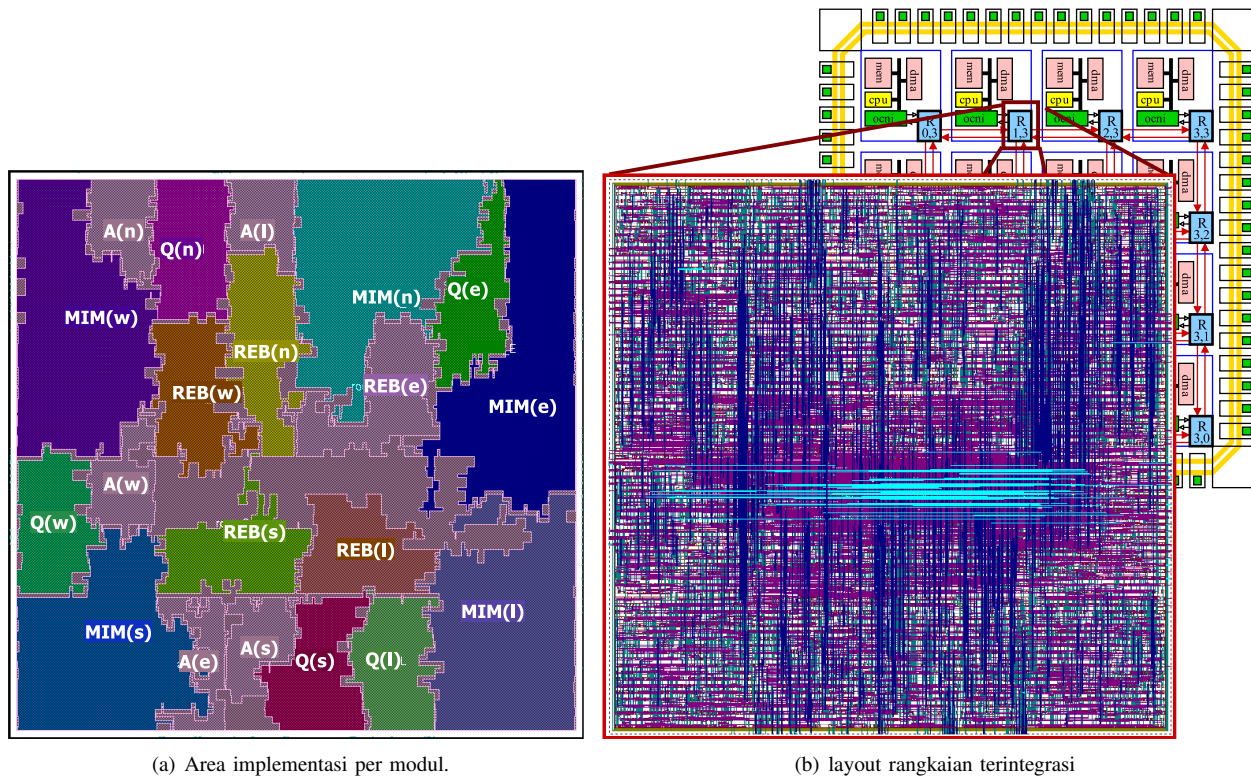
B. Hasil Layout Rangkaian Terintegrasi

Gambar 4 menunjukkan layout dari rangkaian Router On-Chip dengan menggunakan teknologi sel-standard dengan ukuran fitur transistor sebesar 180 nm. Gambar 4(a) memperlihatkan tampilan amoeba (*amoeba view*) dari rangkaian terintegrasi router on-chip. Tampilan tersebut menunjukkan alokasi area setiap modul-modul utama dari jaringan on-chip. Modul-modul untuk setiap port diindikasikan oleh keterangan dalam tanda kurung dari setiap port. Tanda '(e)', '(n)', '(w)', '(s)' dan '(l)' masing-masing menunjukkan modul-

modul untuk port-port *east, north, west, south* dan *local*. Gambar 4(b) menunjukkan hasil penempatan sel-sel logika ke dan routing sambungan kawat metal antar modul. Layout router jaringan on-chip dibuat dengan menggunakan software *Silicon Encounter* dari *Cadence*.

VI. KESIMPULAN

Optimasi jumlah interkoneksi IO (jalur data) pada struktur interkoneksi router jaringan on-chip telah dipaparkan dalam paper ini. Hasil optimasi telah memberikan nilai efisiensi area sel-sel rangkaian logika yang cukup berarti, yaitu sekitar 21% untuk router jaringan on-chip dengan algoritma routing deterministik. Efisiensi tertinggi diperoleh dari hasil optimasi modul *MIM*, yaitu sekitar $0.01 mm^2$. Dengan ukuran area yang lebih kecil, maka konsumsi daya statik dari rangkaian terintegrasi router jaringan on-chip menjadi lebih kecil, dimana daya statik dari sebuah rangkaian terintegrasi sangat tergantung dari jumlah transistor yang dibutuhkan untuk merancang sebuah rangkaian terintegrasi. Data kuantitatif dari penurunan disipasi daya statik hasil dari optimasi area belum ditunjukkan pada paper.



Gambar 4. Layout rangkaian terintegrasi dari Router On-Chip dengan algoritma routing deterministik (e =east, n =north, w =west, s =south, l =local, Q =FIFO queue, A =Arbiter).

REFERENCES

- [1] P. Gratz, C. Kim, K. Sankaralingam, H. Hanson, P. Shivakumar, S. W. Keckler, and D. Burger. "On-Chip Interconnection Networks of the TRIPS Chip". *IEEE Micro*, 27(5):41–50, Sep./Oct. 2007.
- [2] Y. Hoskote, S. Vangal, A. Singh, N. Borkar, and S. Borkar. "A 5-GHz Mesh Interconnects for A Teraflops Processor". *IEEE Micro*, 27(5):51–61, Sep./Oct. 2007.
- [3] D. A. Iltzky, J. D. Hoffman, A. Chun, and B. P. Esparza. "Architecture of the Scalable Communications Core's Network on Chip". *IEEE Micro*, 27(5):62–74, Sep./Oct. 2007.
- [4] G. Palermo, C. Silvano, G. Mariani, R. Locatelli, and M. Coppola. "Application-Specific Topology Design Customization for STNoC". In *Proc. the 9th EUROMICRO Conf. on Digital System Design: Architectures, Methods and Tools (DSD'07)*, pages 547–550, 2007.
- [5] F. A. Samman, T. Hollstein, and M. Glesner. Multicast Parallel Pipeline Router Architecture for Network-on-Chip. In *Proc. Design Automation and Test in Europe (DATE'08)*, pages 1396–1401, 2008.
- [6] F. A. Samman, T. Hollstein, and M. Glesner. Networks-on-Chip based on Dynamic Wormhole Packet Identity Management. *VLSI Design, Journal of Hindawi Pub. Corp.*, vol. 2009(Article ID 941701):1–15, Jan 2009.
- [7] F. A. Samman, T. Hollstein, and M. Glesner. Adaptive and Deadlock-Free Tree-based Multicast Routing for Networks-on-Chip. *IEEE Trans. Very Large Scale Integration (VLSI) Systems*, 18(7):1067–1080, July 2010.
- [8] F. A. Samman, T. Hollstein, and M. Glesner. New Theory for Deadlock-Free Multicast Routing in Wormhole-Switched Virtual-Channelless Networks-on-Chip. *IEEE Trans. Parallel and Distributed Systems*, 22(4):544–557, April 2011.
- [9] F. A. Samman, T. Hollstein, and M. Glesner. Wormhole Cut-Through Switching: Flit-Level Messages Interleaving for Virtual-Channelless Network-on-Chip. *Elsevier Science Journal, Microprocessors and Microsystems-Embedded Hardware Design*, 35(3):343–358, May 2011.
- [10] F. A. Samman, T. Hollstein, and M. Glesner. Planar Adaptive Network-on-Chip Supporting Deadlock-Free and Efficient Tree-Based Multicast Routing Method. *Elsevier Science Journal, Microprocessors and Microsystems-Embedded Hardware Design*, 36(6):449–461, Aug. 2012.