

BAB 4

KEPUTUSAN DAN PERBINCANGAN

4.1: PENGENALAN

Projek ini menggunakan sepenuhnya perkakasan rekabentuk terbantu komputer Synopsys dan Xilinx yang dilaksanakan dalam suasana sistem pengoperasian Unix. Proses sintesis dan proses penempatan dan penghalaan menggunakan pustaka daripada Xilinx yang tertera dalam Jadual 4.1. Pustaka ini beroperasi pada 3.3V dan berteknologi $0.25\mu\text{m}$ [35]. Simulasi peringkat akhir menggunakan pustaka *Simprim_Vital* dan data *Standard Delay Format* (SDF) untuk penyelakuan pengesahbetulan fungsi dan pemasaan [36].

Jadual 4.1: Maklumat pustaka yang digunakan

Pustaka	XC4036XL
Pakej	BG432
Gred Kelajuan	2

4.2: PENYELAKUAN

Terdapat dua jenis penyelakuan iaitu penyelakuan untuk pengesahbetulan fungsi dan pemasaan. Pengesahbetulan fungsi adalah hanya membabitkan kebolehfungsian sistem tersebut semata-mata tanpa mempertimbangkan pemasaan (di mana masa-lengah get dan parameter-parameter pemasaan dianggap sifar), manakala pengesahbetulan pemasaan melibatkan pemasaan dan fungsi [27]. Pengesahbetulan fungsi ini dilakukan pada peringkat penyelakuan perihalan dan peringkat penyelakuan

sintesis, manakala pengesahbetulan pemasaan hanya dilakukan pada peringkat penyelakuan penempatan dan penghalaan (rujuk Rajah 3.7).

Penyelakuan Pengesahbetulan Fungsi

Sekiranya reset, jawapan yang diberi ialah **datasifar** yakni “00000000 00000000 00000000 00000001” dan data masukan pada penyelaku ialah pada pinggir-jam-negatif (*negative-edge*) seperti *s* dalam Rajah 4.1a. Ia akan masuk ke perkakasan (yakni PTA) pada pinggir-jam-positif (*positive-edge*) seperti *d* dalam Rajah 4.1(a-d) yang menunjukkan setiap data akan mengambil tiga kitaran untuk mendapatkan jawapan. Tatatanda *ji* merupakan jawapan kepada data masukan *di*, di mana *i* ialah nombor nyata. Jawapan ini boleh disemak menggunakan laman web [37].

Penyelakuan Pengesahbetulan Pemasaan

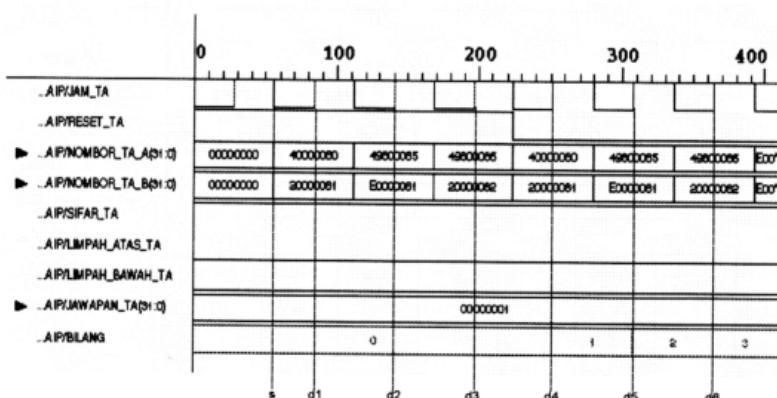
Rajah 4.1(a-d) merupakan gambarajah penyelakuan pengesahanbetulan pemasaan di mana pemasaan-jam yang kenakan mestilah lebih daripada tempoh minimumnya iaitu 54.135ns seperti 55.0ns. Sekiranya tempoh yang dikenakan ialah 40.0ns hanya data *d*19 hingga *d*23 sahaja yang berjaya menghasilkan jawapan yang betul, pada tempoh 42.0ns pula *d*6 dan *d*13 yang berjaya manakala jawapan untuk *d*4, *d*5, *d*7, *d*12 dan *d*15 dihasilkan pada tempoh 44.0ns, ini adalah disebabkan kebergantungan data-data tersebut kepada kewujudan bawaan dalam CPA untuk penambahan bawaan-jumlah. Yakni sekiranya janaan bawaan wujud dalam setiap bit (dalam CPA), maka data tersebut melalui laluan kritikal.

Jadual 4.2 menunjukkan ringkasan maklumat untuk Rajah 4.1 (a dan b) dan Jadual 4.3 untuk Rajah 4.1 (c dan d). Data (*d*1 ~ *d*4) dan (*d*8 ~ *d*11) adalah untuk ujian reset

manakala (d4 ~ d7) dan (d12 ~ d17) merupakan penganalisaan terhadap empat kes operasi pendaraban iaitu ($+ \times +$, $+ \times -$, $- \times +$ dan $- \times -$). Ujian untuk limpahan atas dan bawah ditunjukkan oleh data (d18 ~ d23).

Jadual 4.2: Keterangan gambarajah penyelakuan I dan II

Masukan	Keluaran	Keterangan	Tempoh Minimum (ns)
D1 - d3	sifar	reset aktif	< 40.0
D4	j4	$+ \times + = + (3.0 \times 5.0 = 15.0)$	44.0
D5	j5	$+ \times - = - (100.75 \times -5.0 = -503.75)$	44.0
D6	j6	$+ \times + = + (201.5 \times 10.0 = 2015.0)$	42.0
D7	j7	$- \times - = + (-5.0 \times 5.0 = -25.0)$	44.0
D8 - d11	sifar	Data diset ke sifar kerana reset aktif.	< 40.0



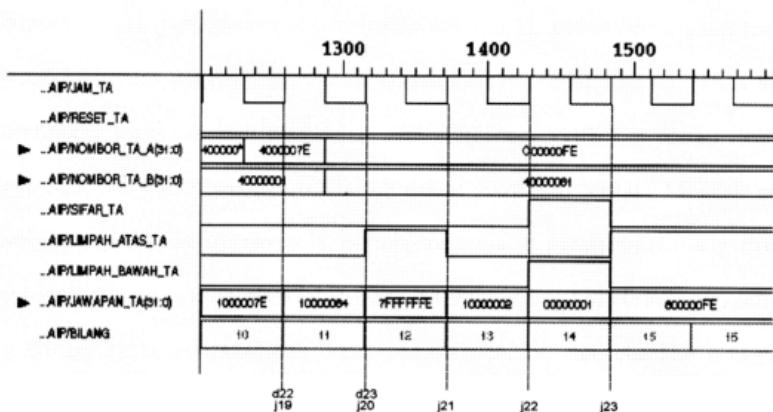
Rajah 4.1a: Gambarajah penyelakuan I

	500		600		700		800	
_AIPJAM_TA								
_AIPRESET_TA								
► _AIPNOMBOR_TA_A[31:0]	E0000061	20000061	CD200067	00000067	32E00067	C0000067	00000067	32E00
► _AIPNOMBOR_TA_B[31:0]		E0000061		00000067	00000067	FRE00067	00000067	00000067 FFE00
_AIPSIFAR_TA			-					
_AIPLIMPAH_ATAS_TA								
_AIPLIMPAH_BAWAH_TA								
► _AIPJAWAPAN_TA[31:0]	000F	70000092	84000087	78E00089	4600007		00000001	
_AIPBILANG	3	4	5	6	7	8	9	10
	d7 j4	d8 j5	d9 j6	d10 j7	d11	d12	d13	

Rajah 4.1b: Gambarajah penyelakuan II

	900		1000		1100		1200	
_AIPJAM_TA								
_AIPRESET_TA								
► _AIPNOMBOR_TA_A[31:0]	32E00067	3EB0027E	C119CEYE	4000007E	40000001	400000FF	4000000F	
► _AIPNOMBOR_TA_B[31:0]	FFB00087	00000087	12800E7E	4000007E	40000001	40000081	40000001	
_AIPSIFAR_TA								
_AIPLIMPAH_ATAS_TA								
_AIPLIMPAH_BAWAH_TA								
► _AIPJAWAPAN_TA[31:0]	00000001	00P3468F	0000006F	00P3468F	3300B88F	5A6A997D	A575577D	1*
_AIPBILANG	2	3	4	5	6	7	8	9
	d14	d15 j12	d16 j13	d17 j14	d18 j15	d19 j16	d20 j17	d21 j18

Rajah 4.1c: Gambarajah penyelakuan III



Rajah 4.1d: Gambarajah penyelakuan IV

Jadual 4.3: Keterangan gambarajah penyelakuan III dan IV.

Masukan	Keluaran	Keterangan	Tempoh Minimum (ns)
d12	j12	- × + = - ($357.875 \times 256.125 = -91673.4375$)	44.0
d13	j13	+ × + = + ($256.0 \times 256.0 = 65536.0$)	42.0
d14	j14	+ × - = - ($357.875 \times -256.125 = -91673.4375$)	> 44.0
d15	j15	+ × + = + ($357.875 \times 357.875 = 91673.4375$)	44.0
d16	j16	+ × + = + ($0.7457 \times 0.5724 = 0.42683868$)	42.0
d17	j17	- × + = - ($-0.7457 \times 0.5724 = -0.42683868$)	> 44.0
d18	j18	+ × + = + (eks ₁₂₇ dan eks ₁₂₇ untuk data masukan)	> 44.0
d19	j19	+ × + = + (eks _{min} dan eks _{min} untuk data masukan)	40.0
d20	j20	+ × + = + (maksima positif kerana limpahan atas)	40.0
d21	j21	+ × + = + (eks ₁₂₇ dan eks _{min} untuk data masukan)	40.0
d22	j22	sifar kerana limpahan bawah	40.0
d23	j23	+ × - = - (maksima negatif kerana limpahan atas)	40.0

Masalah Dalam Penyelakuan

Pada penyelakuan peringkat perihalan (rujuk Rajah 3.7), pemasalahan yang wujud hanyalah berkenaan dengan penggunaan samada sintaks signal atau variable yang menyebabkan ketaksampaian data dalam pengaturcaraan VHDL, ia diatasi dengan adanya peninjau isyarat dan pembolehubah dalam penyelaku VHDL. Masalah pada penyelakuan peringkat sintesis pula merupakan masalah penggunaan ruang storan yang besar kerana ia menjanakan lebih 6000 fail (bergantung kepada saiz perkakasan yang dibina) ketika penganalisaan untuk penyelakuan. Ini menyebabkan gangguan pada pengguna lain dan pelayan boleh tergantung (*hang*). Masalah ini tidak dapat diatasi tetapi sekadar tidak menyusahkan orang lain, penganalisaan tersebut dibuat selepas waktu bekerja dan ketika penyelakuan pengesahbetulan fungsi, *testbench* yang digunakan akan mengesan kesalahan secara automatik supaya cepat dan fail-fail yang dijanakan akan dihapuskan sebaik sahaja penganalisaan selesai.

Pada penyelakuan peringkat penempatan dan penghalaan, masalah yang dihadapi ialah berkenaan dengan pustaka *Simprim_Vital*, yakni pustaka yang sedia-ada dianalisiskan oleh penganalisa Synopsys versi yang terdahulu. Oleh itu, ia perlu dianalisiskan semula untuk versi yang ada. Pustaka yang dianalisa semula disimpan dalam ruang storan pengguna yang menyebabkan penggunaan ruang storan. Selain daripada itu, proses penyelakuan dan sintesis adalah agak perlahan kerana mesinkerja yang digunakan berfrekuensi 50MHz.

4.3 : SINTESIS

Proses sintesis menggunakan peralatan *Synopsys FPGA Express* [38]. Ia akan menghasilkan fail VHDL untuk penyelakuan pengesahbetulan fungsi dan fail XNF (*Xilinx Netlist File*) untuk perlaksanaan penempatan dan penghalaan. Rekabentuk yang dihasilkan berkelajuan 7.0MHz atau tempohnya ialah 125.5ns di mana frekuensi kekangan yang dikenakan ialah 25.0MHz. Laluan kritikal ini adalah pada tahap kedua yang disebabkan oleh adanya CPA untuk penambahan bawaan-jumlah dari pepohon Wallace pada tahap 1. Laluan kritikal untuk tahap pertama ialah 79.2ns manakala tahap ketiga ialah 120.1ns kerana tahap ini mengandungi penambahan untuk proses pembundaran.

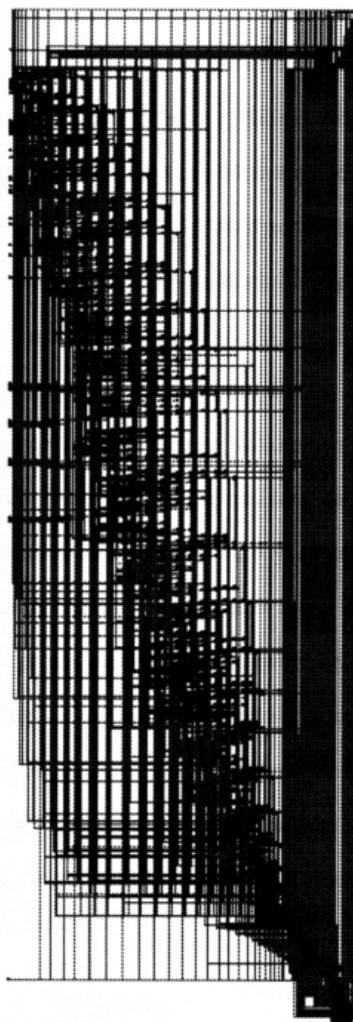
Dalam sintesis, ketidakcukupan memori merupakan masalah utama. Ketika menggunakan mesin-kerja yang mempunyai memori 256MB, senibina PTA hanya ada dua tahap sahaja iaitu tahap pertama dan kedua, tahap ketiga tidak dimasukkan sehinggalah mesin kerja tersebut mempunyai memori 512MB. Masalah ini juga bergantung kepada aturcara iaitu:

- (a) Aturcara tersebut hendaklah dibuka seperti dalam Program 3.7 dan 3.10.
- (b) Elakkan penggunaan sintak `port map`, yakni jadikan subrekabentuk sebagai pakej bukannya komponen seperti dalam Program 3.7 dan 3.10. Walaubagaimanapun, sintak `port map` ini sangat berguna untuk projek berkumpulan.
- (c) Penggunaan data jenis `std_logic_1164` (mengandungi `U, X, 0, 1, Z, W, L, H` dan `-`) juga memerlukan memori yang banyak berbanding dengan data jenis

bit (mengandungi 0 dan 1 sahaja) tetapi data `std_logic_1164` diperlukan ketika penyelakuan pengesahbetulan pemasaan.

- (d) Dalam proses sintesis menggunakan *Synopsys Design Analyzer* [39][40], subrekabentuk boleh disintesiskan satu demi satu dan ketika rekabentuk keseluruhan (*top design*) disintesis, atribut *don't touch* boleh dikenakan pada setiap subrekabentuk supaya proses sintesis tidak menggunakan memori yang banyak. Sungguhpun begitu, ia memberikan prestasi yang tidak memberangsangkan kerana *buffer* dikenakan di antara subrekabentuk.
- (e) Masalah penggunaan memori juga wujud apabila sempadan subrekabentuk dihilangkan selepas sintesis.

Sintesis menggunakan *Synopsys FPGA Express* dibuat berulangkali untuk menentukan pustaka yang boleh menampung saiz cip tersebut, di mana kebolehan penampungan ditentukan ketika proses penempatan dan penghalaan. Secara amnya, dalam proses sintesis, persoalan yang timbul ialah adakah aturcara VHDL boleh disintesiskan atau tidak, di mana ia bergantung kepada cara pengaturcaraan, kemampuan teknologi PLoT dan mesin kerja. Rajah 4.2 menunjukkan gambarajah skematicik PTA 32bit bertalian paip selepas proses sintesis di mana warna biru merupakan penyambungan antara get (warna hijau).



Rajah 4.2 : Gambarajah skematik PTA 32bit bertalian paip

4.4 : Proses Penempatan dan Penghalaan

Jadual 4.4 menunjukkan pemasaan yang didapati setelah melakukan penempatan dan penghalaan dengan menggunakan peralatan Xilinx Alliance [41] dan pustaka xc4036xl-bg432-2, frekuensi kekangan yang dikenakan adalah sama ketika sintesis. Jika diperhatikan tempoh minimumnya,(54.135ns) adalah lebih baik daripada tempoh minimum dalam proses sintesis (125.2ns). Ini disebabkan proses sintesis hanya untuk menghasilkan XNF, di mana XNF (bagi famili XC4000) tidak menampung sel-sel CLB (*Configurable Logic Block*) dan IOB (*Input Output Block*), oleh itu sel-sel ini digantikan dengan set get-get logik gabungan, jujukan dan tiga-keadaan yang setara, maka dengan itu laporan pemasaan yang dihasilkan dalam proses sintesis tidak memberi masa-lengah perlaksanaan CLB dan IOB yang sebenar kerana penggantian tersebut boleh diterjemahkan dalam beberapa aras logik untuk melaksanakan logik dalam CLB [42]. Dalam proses penempatan dan penghalaan ia menghasilkan masa-lengah CLB dan IOB yang sebenar setelah get-get logik setara dipetakan kembali ke dalam bentuk CLB dan IOB di mana masa-lengahnya bergantung kepada frekuensi kekangan dan bagaimana CLB diaturcarakan. Dengan kata lain sintesis menghasilkan pemasaan get manakala penempatan dan penghalaan menghasilkan pemasaan CLB dan IOB.

Jadual 4.5 pula memberikan maklumat tentang bilangan komponen-komponen yang digunakan untuk pendaraban ini, ia terdiri daripada berikut.

- IOB

Setiap IOB mengandungi elemen jujukan seperti DFF (D Flip-Flop)

- *Global Buffer IOB*

Untuk menentukan taburan masa-lengah jam yang bersesuaian

- *CLB*

Ia merupakan blok logik teratucara (atau sel logik asas) dalam famili Xilinx XC4000 yang terdiri daripada selak, flip-flop, penjana fungsi logik F dan G (*LUT-Look Up Table* 4 masukan), penjana fungsi logik H (*LUT* 3 masukan), logik bawaan khusus (*dedicated carry logic*) dan masukan C dan keluaran Y.

- *BUFGLS (Global Low-Skew Buffer)*

Komponen yang boleh memacu semua pin jam flip-flop.

- *STARTUP*

Ia digunakan untuk menyambung set/reset sangat, kawalan tiga-keadaan (*tristate*) sangat dan *user configuration clock*.

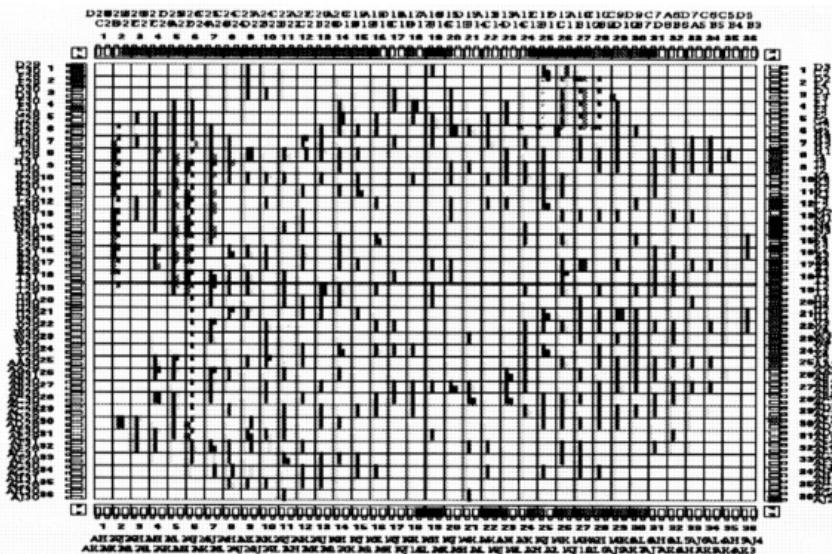
Maklumat lanjut berkenaan komponen dan famili Xilinx XC4000 boleh didapati daripada [27][35]. Rajah 4.3 menunjukkan gambarajah pelan-lantai (*floorplan*) untuk pendaraban titik apungan 32bit bertalian paip. Proses ini mengambil masa 5 hingga 6 jam.

Jadual 4.4 : Pemasaan bagi jam.

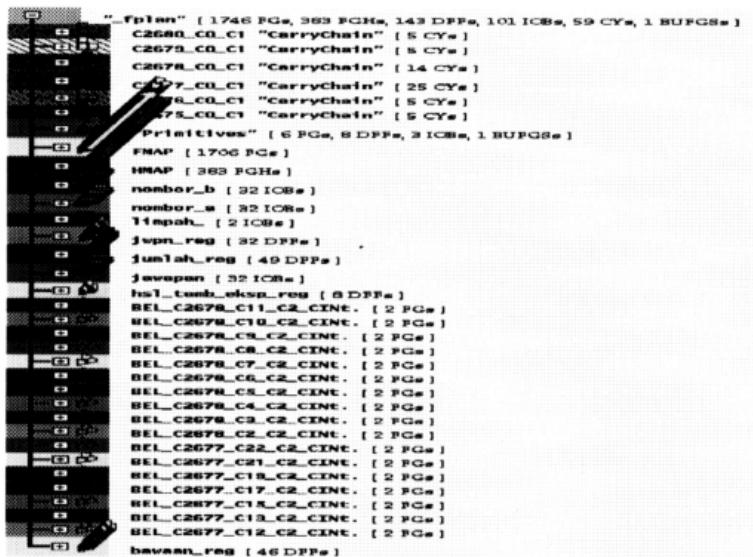
Tempoh minimum	54.135ns
Frekuensi maksima	18.472MHz

Jadual 4.5: Bilangan komponen yang digunakan

Bilangan IOB luaran	100 IOB daripada 288	34%
Flip flop	99	
Selak	0	
Bilangan <i>Global Buffer IOB</i>	1 daripada 8	12%
Flip-flop	0	
Selak	0	
Bilangan CLB	1007 CLB daripada 1296	77%
Jumlah selak	0 daripada 2592	0%
Jumlah CLB flip-flop	143 daripada 2592	5%
LUT 4 masukan	1746 daripada 2592	67%
LUT 3 masukan	383 daripada 1267	29%
Bilangan BUFGLS	1 daripada 8	12%
Bilangan STARTUP	1 daripada 1	100%



(a)



(b)

Rajah 4.3: Pelan-lantai (a) beserta petunjuk (b) untuk pendaraban titik apungan 32bit bertalian paip

4.5 : PERBINCANGAN

Struktur

Kebiasaannya, nombor-nombor titik-apungan tak-normal tidak disokong oleh kebanyakan rekabentuk unit aritmetik yang mematuhi piawai IEEE. Ini adalah disebabkan kos yang tinggi untuk perlaksanaannya kerana ia berbeza daripada nombor-nombor titik-apungan yang normal dan ia memerlukan rekabentuk yang lebih kompleks dan semestinya masa perlaksanaan purata yang panjang [13].

PTA ini lebih laju jika dibandingkan dengan dengan projek di dalam [43] yang menggunakan teknologi 4 Actel A1280 FPGA. Di mana ia memerlukan 3 tahap talian paip yang memerlukan masa kitaran 245ns di mana pendarabnya memerlukan enam

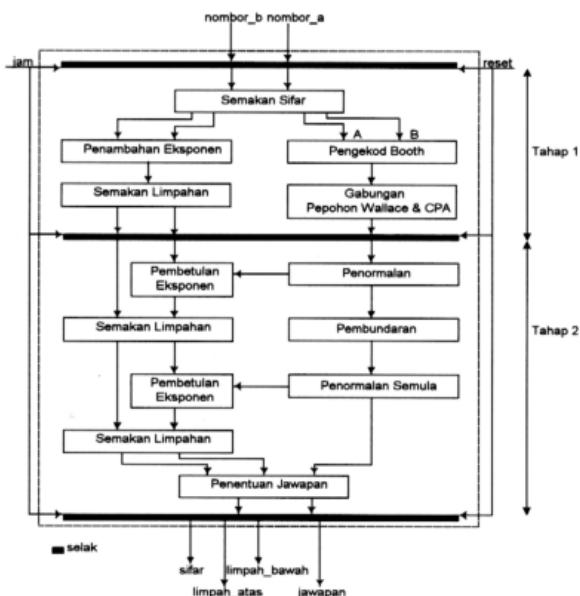
pendaman iaitu satu untuk eksponen, empat untuk pendaraban dan satu lagi untuk penormalan.

Struktur senibina PTA ini, bersamaan dengan struktur di dalam Rajah 2.9 (struktur untuk pendaraban titik apungan 64bit), di mana tahap pertama dalam struktur Rajah 2.9 mengambil masa 9.55ns, tahap kedua pula 8.7ns manakala tahap ketiga ialah 7.5ns. Jika diperhatikan, bahagian kritis ialah pada tahap pertama bukannya pada tahap kedua seperti dalam projek ini. Ini menunjukkan CPA yang berkelajuan tinggi diperlukan dalam pendarab pepohon Wallace supaya penggunaan pepohon Wallace akan memberi makna atau dengan kata lain tiada gunanya menggunakan pepohon Wallace tanpa CPA yang laju kerana pepohon Wallace telah terbukti kepentasannya [44]. CPA yang digunakan dalam projek ini mempunyai masa-lengah yang tinggi kerana ia merupakan rekabentuk yang sedia-ada dalam SDW, bukannya dibina atas kemahuan matlamat perekra. Dalam SDW hanya terdapat tiga model penambah iaitu RCA, CLA dan CLA berkelajuan tinggi. Malangnya, CLA berkelajuan tinggi tidak dapat dicapai atau digunakan kerana tiada lesen SynLib (*Synthesis Library*), manakala RCA dan CLA tidak memerlukan lesen [33]. Untuk membina CPA seperti dalam Rajah 2.11 memerlukan penambahan memori mesinkerja (> 256 MB) untuk proses sintesis.

Penggunaan talian paip bertujuan untuk meningkatkan masa truput, malangnya ia secara tidak langsung meningkat keperluan perkakasan dan masa pendaman dengan wujud flip-flop. Jika diperhatikan pada Rajah 3.2, peralihan di antara tahap pertama ke tahap kedua menggunakan 50 flip-flop untuk bawaan dan 50 untuk jumlah, ini menjadikan 110 flip-flop digunakan (termasuk 10 flip-flop untuk peralihan eksponen)

dan ia menyebabkan masalah taburan isyarat jam. Masalah ini boleh dikurangkan dalam dua cara pertama menggunakan struktur pendaraban mantisa seperti dalam Rajah 2.10 dan yang kedua menggunakan *fanout tree* untuk penaburan isyarat jam (mukasurat 341 [30]).

Jika menggunakan struktur pendarab mantisa Rajah 2.10, berkemungkinan PTA boleh dilaksanakan dalam dua pendaman sahaja seperti dalam Rajah 4.4 dan masa keseluruhan pendaraban mantisa menjadi lebih singkat kerana disebabkan oleh tindanan masa perambatan untuk bahagian tatasusun dan bahagian penambahan akhir selagi bahagian tatasusun dan penambahan akhir tidak dipisahkan oleh talian paip. Struktur Rajah 4.4 juga memerlukan penambah yang cepat dalam pembundaran supaya masa kritisikal tahap pertama dan kedua tidak jauh beza.



Rajah 4.4 : PTA 32bit dalam dua pendaman

Teknologi

Dengan menggunakan VHDL, aturcara pendarab ini dengan mudah boleh dikembangkan ke titik apungan 64bit. Sungguhpun begitu, ia memerlukan stesen-kerja yang mempunyai ingatan yang tinggi untuk sintesis bagi mendapatkan fail XNF. Teknologi ini juga membolehkan sesebuah rekabentuk (misalnya PTA) menjadi blok HI yang digunakan sebagai komponen-maya atau sub-komponen dalam SoC (*System on Chip*).

Projek ini menunjukkan bagaimana VHDL boleh menolong atau membantu dalam merekabentuk dan membangunkan PTA. Secara ringkas, ini disebabkan oleh:

- Pendekatan rekabentuk berhierarki modul.
- Rekabentuk berbilang domain
- Tahap pengabstrakan yang boleh ditakrifkan oleh pengguna

Ia juga menunjukkan bagaimana penyelaku VHDL membantu penganalisaan PTA melalui berbagai peralatan penganalisa yakni termasuklah menganalisa gambarajah skematik, gambarajah gelombang pemasaan dan sistem peninjau isyarat dan pembolehubah. VHDL juga mempermudahkan proses sintesis PTA dengan menggunakan peralatan sintesis yang boleh disintesiskan secara-terus.

Walaupun dengan penggunaan bahasa pengaturcaraan biasa dapat meningkat kelajuan penyelakuan, namun ia dapat diimbangi dengan peningkatan kekayaan algoritma yang disediakan melalui VHDL. Sungguhpun begitu, tidak semua data dalam VHDL boleh disintesiskan seperti data titik-apungan (contoh:1.35), fizikal (contoh:masa) dan lain-

lain. Kejayaan sintesis juga bergantung kepada komponen yang ada dalam SDW. Sebagai contoh, penggunaan tanda ‘/’ (tanda untuk operasi pembahagian) dalam aturcara VHDL ataupun Verilog, tidak boleh disintesiskan kerana tiada komponen pembahagi dalam SDW. Oleh itu, operasi pembahagi hendaklah diterjemahkan kedalam bentuk operasi penolakan (komponen yang ada SDW).