

**PENGAWAL LAMPU TRAFIK
DENGAN PENGESAN MENGGUNAKAN FPGA**

Oleh

MOHD SHAFIAN BIN SHAFIEE

Disertasi ini dikemukakan kepada

UNIVERSITI SAINS MALAYSIA

Sebagai memenuhi sebahagian daripada syarat keperluan
untuk ijazah dengan kepujian

SARJANA MUDA KEJURUTERAAN (KEJURUTERAAN ELEKTRIK)

Pusat Pengajian Kejuruteraan

Elektrik dan Elektronik

Universiti Sains Malaysia

MEI 2006

ABSTRAK

Projek ini adalah untuk merekabentuk suatu sistem pengawal lampu trafik dengan penderia yang menggunakan perisian Xilinx atau berasaskan konsep FPGA. Rekabentuk pengawal lampu trafik ini melibatkan kawalan penderia pada setiap jalan pada simpang jalan tersebut. Penderia pada setiap simpang akan mengesan kehadiran kenderaan yang hadir pada setiap jalan dan mengawal perjalanan lalulintas berdasarkan kehadiran kenderaan pada simpang tersebut. Pengawal lampu trafik ini boleh digunakan untuk tempoh 24 jam dan dikelaskan kepada beberapa keadaan tertentu. Kenderaan yang dikesan akan diberikan melalui persimpangan ini berdasarkan bilangan kenderaan yang melalui simpang tersebut. Projek ini menggunakan empat simpang sebagai contoh yang paling asas untuk merealisasikan projek tahun akhir ini. Pada permulaannya, idea diperolehi dan kemudian idea tersebut dibuat gambarajah blok sistem. Daripada gambarajah blok sistem ini, projek ini akan direkabentuk menggunakan perisian Xilinx. Seterusnya program yang direkabentuk menggunakan perisian Xilinx akan dipindahkan ke dalam litar bersepadu EPROM. Setelah dipindahkan ke dalam litar bersepadu EPROM, objektif bagi menghasilkan suatu pengawal lampu trafik dengan penderia dapat dihasilkan dengan melihat keluaran daripada litar Xilinx. Sistem ini membolehkan pengawal lampu trafik berfungsi secara lebih efektif dan berkesan serta boleh digunakan pada sebarang keadaan jalanraya yang berlaku.

ABSTRACT

This project is to design a traffic light with sensor using FPGA (Field Programmable Gate Array). This design of traffic light is using sensor for the entire road in the junction. The sensor will sense the vehicle in that junction and will control the system depending on the vehicle that will be sense by the sensor. This project can be use for 24 hour and can be classified with other condition. The idea for this project was started from the beginning and the idea will presented to block diagram. From the block diagram this project will be designing using Xilinx. The design of this project will transfer to IC EPROM and the objective for this project will be seen on the board Xilinx. This system can function very effective and can be used for many other conditions.

ISI KANDUNGAN

	MUKA SURAT
ABSTRAK	ii
ISI KANDUNGAN	iv
SENARAI GAMBARAJAH	vii
SENARAI JADUAL	ix
PENGHARGAAN	x
BAB 1 PENGENALAN	
1.1 Pendahuluan	1
1.2 Tujuan Projek	3
1.3 Operasi Projek	4
1.4 Perjalanan Projek	5
BAB 2 PENGENALAN KEPADA FPGA DAN XILINX	
2.1 Pengenalan	7
2.2 Struktur FPGA	8
2.3 Senibina Xilinx XC 4000	8
2.4 Kelebihan Menggunakan FPGA	9
2.5 Pengenalan Kepada Xilinx	10
2.5.1 Pengarah Projek	11
2.5.2 Penyunting Skematik	12
2.5.3 Penyelaku Logik	15
2.5.4 Pengarah Rekabentuk	16

	2.5.5 Pengarah Pustaka	17
	2.5.6 Penyunting Simbol	17
BAB 3	REKABENTUK LITAR	
	3.1 Pengenalan	18
	3.2 Litar Pembilang Menaik	19
	3.3 Litar Pembilang Menurun	23
	3.4 Litar Utama (Litar Trafik)	28
	3.5 Skematik Litar Keseluruhan Projek	32
BAB 4	PENYELAKUAN DAN PERLAKSANAAN	
	4.1 Pengenalan	33
	4.2 Penyelakuan Projek	34
	4.3 Perlaksanaan	38
BAB 5	PERKAKASAN	
	5.1 Pengenalan	40
	5.2 Proses Perlaksanaan	42
BAB 6	PENUTUP	
	6.1 Perbincangan	43
	6.2 Kebaikan	44
	6.3 Masalah Projek	45
	6.4 Cadangan	46
	6.5 Kesimpulan	47

RUJUKAN

**LAMPIRAN 1: LITAR SKEMATIK MAKRO MASUKAN PEMBILANG
MENAİK 0 KE 23**

**LAMPIRAN 2: LITAR SKEMATIK MAKRO MASUKAN PEMBILANG
MENURUN 40 KE 0**

LAMPIRAN 3: LITAR SKEMATIK PEMBILANG MENURUN 40 KE 0

LAMPIRAN 4: LITAR SKEMATIK PEMBILANG MENAİK 0 KE 23

LAMPIRAN 5: LITAR SKEMATIK LAMPU TRAFİK

SENARAI GAMBARAJAH

TAJUK	MUKA SURAT
Rajah 1.1: Perletakan penderia pada persimpangan	5
Rajah 2.1: Tetingkap awal perisian untuk memulakan projek	10
Rajah 2.2: Tetingkap untuk penamaan projek baru	11
Rajah 2.3: Tetingkap pengarah projek	11
Rajah 2.4: Tetingkap untuk memilih penyunting skematik	12
Rajah 2.5: Tetingkap Penyunting Skematik	13
Rajah 2.6: Tetingkap penghasilan makro	13
Rajah 2.7: Tetingkap penamaan masukan dan keluaran bagi makro	14
Rajah 2.8: Makro yang terhasil	14
Rajah 2.9: Tetingkap aturcara HDL	15
Rajah 2.10: Tetingkap penyelakuan	15
Rajah 2.11: Tetingkap untuk keputusan bagi masukan dan keluaran	16

Rajah 3.1: Gambarajah Blok Sistem Pengawal Lampu Trafik dengan Penderia Menggunakan FPGA	18
Rajah 3.2: Keadaan sekarang dan selepas untuk pembilang menaik 0 ke 23	21
Rajah 3.3: Makro pembilang menaik yang direkabentuk untuk membilang naik 0 ke 23	21
Rajah 3.4: Makro masukan untuk pembilang menaik yang direkabentuk untuk membilang naik 0 ke 23	21
Rajah 3.5: Tetingkap penyelakuan bagi pembilang menaik 0 ke 23	22
Rajah 3.6: Keseluruhan litar makro pembilang menaik dengan kiraan 0 ke 23	23
Rajah 3.7: Keadaan sekarang dan selepas untuk pembilang turun 40 ke 0	26
Rajah 3.8: Makro pembilang menurun yang direkabentuk untuk membilang 40 ke 0	27
Rajah 3.9: Makro masukan untuk pembilang menurun yang direkabentuk untuk membilang 40 ke 0	27
Rajah 3.10: Tetingkap penyelakuan bagi pembilang menurun 40 ke 0.	27
Rajah 3.11: Keseluruhan litar skematik pembilang menurun dengan kiraan 40 ke 0	28
Rajah 3.12: Makro bagi litar lampu trafik	31
Rajah 3.13: Litar Rekabentuk Logik Projek	32
Rajah 4.1: Tetingkap pemilihan signal untuk masukan dan keluaran	33
Rajah 4.2: Keadaan ketika tiada kereta di persimpangan	34
Rajah 4.3: Keadaan ketika hanya penderia pertama mengesan kenderaan.	35
Rajah 4.4: Keadaan ketika penderia pertama dan ketiga mengesan kenderaan	36
Rajah 4.5: Keadaan ketika penderia pertama, kedua dan	

ketiga mengesan kenderaan	37
Rajah 4.6: Keadaan ketika penderia pertama, kedua, ketiga dan keempat mengesan kenderaan.	38
Rajah 4.7: Tetingkap Perlaksanaan untuk Aliran Enjin	39
Rajah 5.1: Papan Litar Xilinx	40
Rajah 5.2: Pin pada papan litar Xilinx	41
Rajah 5.3: LED, suis DIP, perintang, suis tekan dan paparan tujuh ruas	41
Rajah 5.4: Litar Bersepadu pada papan litar Xilinx	42

SENARAI JADUAL

TAJUK	MUKA SURAT
Jadual 3.1: Jadual keadaan sekarang dan selepas bagi pembilang naik 0 ke 23	19
Jadual 3.2: Jadual keadaan sekarang dan selepas pembilang menurun 40 ke 0	24
Jadual 3.3: Keadaan keempat-empat penderia dan jalan pada simpang	29

PENGHARGAAN

Alhamdulillah, dengan izinnya yang maha kuasa saya dapat menyiapkan projek tahun akhir ini dan seterusnya menyediakan laporan ini dengan jayanya.

Saya ingin merakamkan setinggi-tinggi penghargaan kepada penyelia, Dr. Che Mat Hadzer Bin Mahmud diatas segala idea-idea yang diberikan dalam menjayakan projek ini.

Tidak dilupakan juga setinggi-tinggi penghargaan dan jutaan terima kasih kepada kedua ibubapa dan keluarga yang tersayang yang tidak jemu-jemu memberi sokongan dan doa sepanjang pengajian saya selama empat tahun disini. Tidak lupa juga kepada rakan-rakan terutama Mohd Fauzi dan Khairul Anuar yang memberi kerjasama yang amat baik sepanjang perjalanan dalam menjayakan projek ini. Tanpa bantuan rakan-rakan saya, mungkin projek ini tidak dapat dijalankan dengan sempurna.

Semoga setiap apa yang dilakukan dalam menjayakan projek ini akan mendapat keberkatan dan keredhaan daripadanya yang Maha Esa, Insyaallah.

Sekian, terima kasih.

Sekalung budi.

Mohd Shafian Bin Shafiee.

BAB 1

Pengenalan

1.1 Pendahuluan

Sistem kawalan lampu trafik merupakan kaedah pengawalan lalulintas yang teratur pada sesuatu simpang. Penggunaannya akan dapat memperbaiki sistem perjalanan di persimpangan tersebut seterusnya mengurangkan risiko kemalangan yang bakal meragut nyawa manusia. Kebiasaannya lampu trafik akan dipasang pada setiap persimpangan. Kaedah pengawalan lalulintas menggunakan lampu trafik di negara ini kebiasaannya menetapkan masa tertentu bagi kenderaan melalui persimpangan tersebut. Pengawalan lalulintas tersebut tidak mempertimbangkan jumlah kenderaan pada persimpangan tersebut tetapi menggunakan masa-masa tertentu dalam rekabentuknya. Sebagai contoh, andai persimpangan tersebut mempunyai empat jalan dan penetapan masa bagi setiap jalan adalah selama 30 saat, maka 120 saat diperlukan bagi membolehkan persimpangan tersebut melengkapkan satu operasi.

Suatu sistem lampu trafik yang lebih mantap diperlukan untuk menjadikan sistem yang lama lebih teratur dan sistematik. Sistem tersebut perlu sesuai dengan setiap keadaan lalulintas yang berlaku pada persimpangan tersebut. Kesesakan lalulintas pada satu persimpangan biasanya berpunca daripada sistem lampu trafik yang tidak cekap dan bolehubah. Apabila keadaan ini berlaku, pelbagai masalah akan timbul. Sebenarnya, kesesakan lalulintas boleh berpunca daripada masalah-masalah yang lain, antaranya

ketidaksiempunaan sistem lampu trafik simpang tersebut dan juga struktur jalan yang tidak terurus.

Pengawal lampu trafik yang baik seharusnya tidak akan menyebabkan kesesakan lalulintas dan juga tidak menyebabkan berlakunya kemalangan pada suatu persimpangan. Konsep pengawalan lampu trafik di persimpangan pada masa sekarang adalah berasaskan masa untuk melicinkan perjalanan lalu lintas pada suatu persimpangan adalah tidak lagi boleh digunakan. Konsep ini tidak dapat dilaksanakan dengan sempurna kerana tidak semua kenderaan yang akan memenuhi setiap jalan di persimpangan tersebut dengan jumlah yang sama dan juga kesemua jalan tersebut tidak semestinya mempunyai kenderaan setiap masa. Penggunaan sistem lampu trafik yang lebih berkesan diperlukan untuk membolehkan lampu trafik yang mengawal suatu persimpangan digunakan dengan efektif dan berkesan.

Pengawal lampu trafik yang perlu direkabentuk seharusnya memiliki pelbagai ciri yang membolehkan penggunaannya digunakan dengan maksima. Keupayaannya untuk mengesan kenderaan pada setiap simpang dan juga mengatur perjalanan kenderaan pada setiap persimpangan perlu supaya perjalanannya pada persimpang tersebut tidak mengalami sebarang masalah. Oleh itu, pengawal lampu trafik yang menggunakan penderia akan direkabentuk dalam menjadikan ia suatu sistem pengawal yang lebih berkesan.

1.2 TUJUAN PROJEK

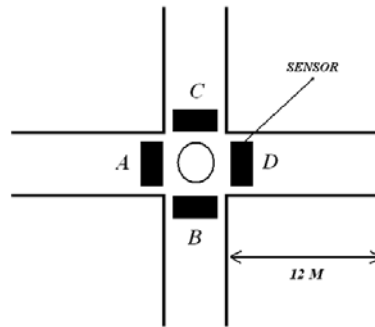
Tujuan projek ini adalah untuk menghasilkan suatu sistem pengawal lampu trafik yang menggunakan penderia. Penderia ini akan diletakkan pada setiap jalan yang berkenaan pada suatu persimpangan. Rekabentuk pengawal lampu trafik yang menggunakan penderia ini akan direkabentuk menggunakan perisian Xilinx. Penderia pada setiap simpang akan mengesan kehadiran kenderaan seterusnya memberikan isyarat pada jalan yang mempunyai kenderaan yang paling banyak atau yang paling awal sampai ke persimpangan tersebut.

Projek ini akan menggunakan simpang empat sebagai asas dalam menggunakan penderia sebagai kaedah dalam mengawal keseluruhan persimpangan tersebut. Sistem lampu trafik yang digunakan pada masa sekarang adalah menggunakan masa, oleh itu penggunaannya secara maksimum tidak dapat dilakukan. Sebagai contoh, suatu simpang yang menggunakan pengawal lampu trafik yang menggunakan masa tidak dapat beroperasi dengan cekap ketika waktu tengah malam atau awal pagi. Pengawal lampu trafik yang menggunakan penderia akan membolehkan penggunaannya akan digunakan sepenuhnya. Jika hanya satu simpang yang mempunyai kenderaan maka simpang tersebut sahaja yang akan diberi isyarat lampu hijau, kuning seterusnya merah. Manakala simpang-simpang lain masih berkeadaan lampu merah. Jika dua jalan pada persimpangan tersebut yang dikesan kehadiran kenderaan, maka pengawal lampu trafik akan memberikan kelebihan pada suatu simpang dan kemudiannya memberikan isyarat pada satu simpang yang lain. Kaedah ini juga akan digunakan andai keempat-empat jalan dipenuhi kenderaan. Kelebihan pada suatu jalan akan diberikan dan kemudian setiap jalan akan diberikan turutan yang sama yakni lampu hijau, kuning dan merah.

Keadaan lampu trafik pada setiap jalan akan berkeadaan lampu merah sekiranya keempat-empat penderia tidak mengesan kehadiran kenderaan pada jalan tersebut. Keadaan akan berubah sekiranya penderia mengesan kenderaan pada suatu simpang tersebut.

1.3 OPERASI PROJEK

Operasi lampu trafik dengan penderia ini bergantung kepada penderia yang dipasang pada setiap jalan pada persimpangan tersebut. Penderia akan mengesan kehadiran kenderaan seterusnya akan memberi arahan kepada proses utama untuk membenarkan kenderaan pada jalan tersebut melalui jalan itu. Sekiranya terdapat kenderaan pada dua jalan pada persimpangan tersebut, kedua-dua penderia akan memberikan arahan kepada proses utama. Proses utama akan memberi keutamaan kepada jalan yang mempunyai kenderaan yang paling banyak atau yang paling awal dikesan oleh penderia supaya melalui persimpangan itu. Selepas itu jalan yang kedua dipenuhi oleh kenderaan pula diberi peluang untuk melalui jalan tersebut. Sekiranya jumlah kenderaan yang sama banyak pada kedua-dua jalan tersebut, masa yang sama akan diberikan untuk membolehkan kenderaan-kenderaan pada jalan tersebut melaluinya. Kelebihan diberikan secara bergilir-gilir kepada kenderaan tersebut.



Rajah 1.1: Perletakan penderia pada persimpangan

Rajah 1.1 menunjukkan perletakan penderia pada setiap jalan pada persimpangan tersebut. Perletakan penderia penting sebenarnya untuk membuatkan sistem pengawal lampu trafik menggunakan penderia berfungsi dengan lebih baik dan sempurna.

1.4 PERJALANAN PROJEK

Terdapat beberapa peringkat yang diperlukan untuk melaksanakan keseluruhan perjalanan projek ini. Peringkat yang pertama adalah untuk mempelajari, memahami bahasa pengaturcaraan dan kaedahnya serta penggunaan perisian Xilinx. Peringkat ini perlu diberikan sedikit penekanan kerana perisian ini belum dipelajari sebelum ini. Terdapat pelbagai kesukaran sewaktu mempelajari perisian ini. Pemahaman terhadap papan litar Xilinx, proses rekabentuk dan sebagainya.

Peringkat kedua adalah untuk merekabentuk litar logik berdasarkan jadual kebenarannya. Peringkat ini memerlukan masa yang paling panjang dalam projek tahun akhir ini. Pelbagai kaedah digunakan dalam merekabentuknya, namun kaedah yang digunakan untuk menyiapkan projek ini adalah dengan menggunakan editor skematik.

Pelbagai kaedah digunakan untuk menyiapkan peringkat yang paling penting ini. Peringkat ketiga adalah untuk melihat samada rekabentuk litar logik yang dibina dengan menggunakan perisian ini adalah benar dengan jadual kebenaran yang diingini. Peringkat ini dikenali sebagai penyelakuan. Peringkat penyelakuan ini akan mengesahkan bahawa litar rekabentuk yang dibina adalah betul.

Peringkat seterusnya adalah proses pelaksanaan. Proses ini adalah untuk menetapkan pin-pin yang akan digunakan pada papan litar Xilinx. Proses ini akan membolehkan pemasangan pada papan litar Xilinx dilakukan. Peringkat seterusnya adalah peringkat pengesahan. Peringkat ini bertujuan untuk menguji rekabentuk logik dan pemasaan dengan menggunakan masukan penyelakuan. Penyelakuan fungsi dan masa akan dilakukan bagi mengenal pasti kesahihan rekabentuk logik yang direkabentuk.

Peringkat terakhir adalah untuk memasukkan rekabentuk litar logik yang dibina ke dalam Litar Bersepadu EPROM. Peringkat ini adalah peringkat terakhir dalam menyiapkan projek pengawal lampu trafik menggunakan penderia.

BAB 2

PENGENALAN KEPADA FPGA DAN XILINX

2.1 PENGENALAN

Perkembangan teknologi yang semakin pesat pada masa kini telah membolehkan pelbagai aplikasi dalam pelbagai bidang dapat dihasilkan dengan lebih mudah dan baik. Pelbagai perisian juga wujud untuk membolehkan penggunaan teknologi digunakan semaksima yang boleh. FPGA atau (*Field Programmable Gate Array*) adalah salah satu daripadanya. FPGA adalah suatu modul yang dapat diprogramkan berdasarkan sistem digital yang mengandungi beribu-ribu get dan mampu melakukan pelbagai aplikasi hanya dalam satu litar terpadu yang tunggal sahaja. Penggunaan FPGA biasanya digunakan dalam pembinaan sistem digit yang rumit dan kompleks. Kelebihan FPGA adalah kebolehan untuk membenarkan sesuatu peranti itu dilaksanakan dalam pelbagai aplikasi.

FPGA terhasil daripada beberapa sistem tatasusunan logik bolehaturcara yang mudah. Antaranya sistem PAL (*Programmable Array Logic*) dan seterusnya berkembang kepada peranti logik bolehaturcara kompleks CPLD (*Complex Programmable Logic Device*). FPGA terbahagi kepada empat teknologi utama iaitu statik RAM, Anti-fuse, Aturcara tatasusunan-EPROM dan Aturcara tatasusunan-EEPROM.

2.2 STRUKTUR FPGA

Tiga elemen utama FPGA adalah terdiri daripada

- 1) Konfigurasi Logik Blok (CLBs)
- 2) Blok-blok masukan dan keluaran (IOBs)
- 3) Saling Sambungan Titik (*'Interconnections points'*)

2.3 SENIBINA XILINX XC 4000

Pembinaannya adalah berdasarkan teknologi yang lama iaitu berasaskan kepada teknologi pemrogram SRAM. Struktur logik yang digunakan dapat dilaksanakan ke dalam bentuk tatasusunan blok-blok oleh aturcara yang dinamakan Konfigurasi Logik Blok. Bahagian masukan dan keluaran daripada tatasusunan dikawal oleh blok masukan dan keluaran dikenali sebagai Blok-Blok Masukan dan keluaran. Cip XC 4000 mengandungi susunan 8 x 8 hingga 30 x 30 Konfigurasi Logik Blok dan 64 hingga 2401 Blok-Blok Masukan dan keluaran.

Penggunaan kedua-dua elemen ini disambungkan dengan pelbagai struktur sambungan yang boleh diprogramkan. Setiap segmen unsur boleh disambungkan untuk membentuk laluan dari satu blok ke blok yang lain dengan menggunakan tatasusunan tersebut. Pembentukan laluan ini dipanggil Saling Sambung Titik.

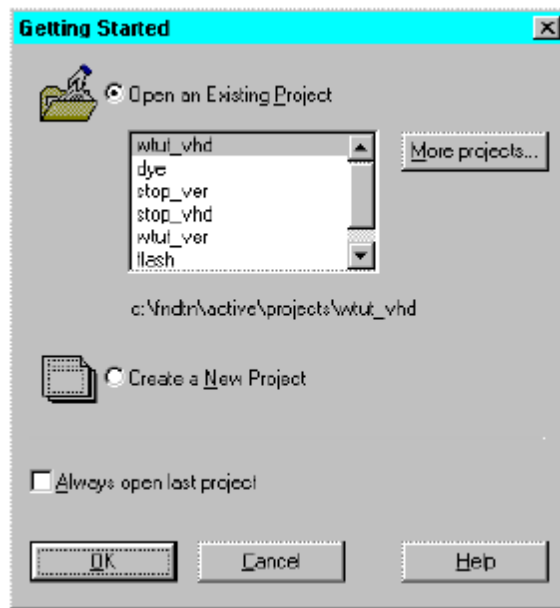
2.4 KELEBIHAN PENGGUNAAN FPGA

Terdapat pelbagai kelebihan FPGA dalam merekabentuk. Antaranya ialah:

- 1) FPGA dapat melakukan perubahan dengan mudah pada rekabentuk sekiranya terdapat sedikit kesilapan ketika dalam proses merekabentuknya.
- 2) Terdapat pelbagai pilihan cip yang disediakan untuk digunakan oleh pereka. Antaranya XC 4000, XC 9000 dan sebagainya.
- 3) Kelebihannya yang utama adalah dapat melaksanakan kebanyakan rekabentuk sistem berdigit dengan hanya memerlukan satu perisian tunggal kerana terdapat beribu-ribu get.
- 4) Penjimatan dapat dilakukan sekiranya terdapat perubahan pada rekabentuk litar kerana ia tidak akan menyebabkan perlunya untuk menukar komponen secara fizikal.
- 5) Oleh kerana terdapat pelbagai get yang dapat digunakan dalam seunit peranti FPGA, maka penggunaan get pada luarannya dapat dijimatkan. Ini dapat menjimatkan ruang bagi sesuatu rekabentuk.
- 6) Proses rekabentuknya juga ringkas dan tidak memerlukan masa yang panjang untuk merekabentuk. Beberapa proses seperti bikin dan selongsong dapat diabaikan.
- 7) Masa juga dapat dijimatkan sekiranya rekabentuk baru dihasilkan menggunakan FPGA kerana prosesnya yang ringkas dan penjimatan dalam kos pembuatannya.

2.5 PENGENALAN KEPADA XILINX

Rekabentuk secara FPGA dengan menggunakan perisian Xilinx boleh dilakukan dengan tiga cara iaitu samaada menggunakan penyunting skematik, penyunting FSM (*'Finite State Machine'*) dan penyunting HDL (*'Hardware Description Language'*). Kesemua kaedah boleh digunakan dengan kaedah seperti yang diterangkan di bawah. Pada permulaan perisian Xilinx, penamaan projek perlu dilakukan sekiranya projek yang baru hendak dimulakan.



Rajah 2.1: Tetingkap awal perisian untuk memulakan projek

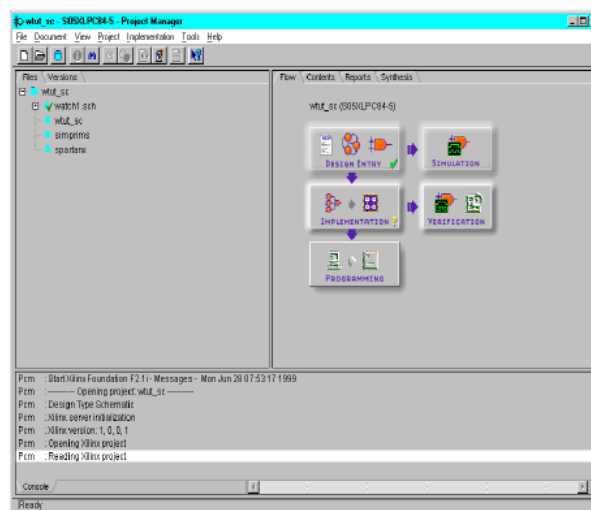
Penamaan projek akan dilakukan pada tettingkap pada Rajah 2.2 dan seterusnya proses ini akan berubah kepada tettingkap pengarah projek.



Rajah 2.2: Tetingkap untuk penamaan projek baru

2.5.1 PENGARAH PROJEK

Fungsinya adalah untuk mengawal keseluruhan peringkat rekabentuk dan juga perjalanan projek. Tetingkap ini juga akan memaparkan segala ralat yang wujud dan kesalahan yang dilakukan pada peringkat rekabentuk samada ketika penyelakuan atau perlaksanaan. Fungsinya yang lain juga adalah untuk memulakan sesuatu projek yang baru. Projek ini akan diperlengkapkan dengan semua sumber rekabentuk.

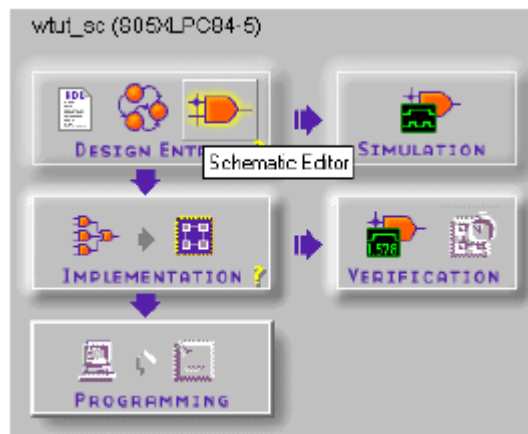


Rajah 2.3: Tetingkap pengarah projek

Pada tetingkap ini terdapat tiga pilihan untuk merekabentuk litar iaitu samaada menggunakan penyunting skematik, penyunting FSM dan penyunting HDL. Pemilihan boleh dibuat pada masukan rekabentuk yang terdapat pada tetingkap tersebut untuk memulakan projek yang baru atau untuk menyambung projek yang sebelum ini. Bagi projek ini, pemilihan untuk menggunakan editor skematik dipilih.

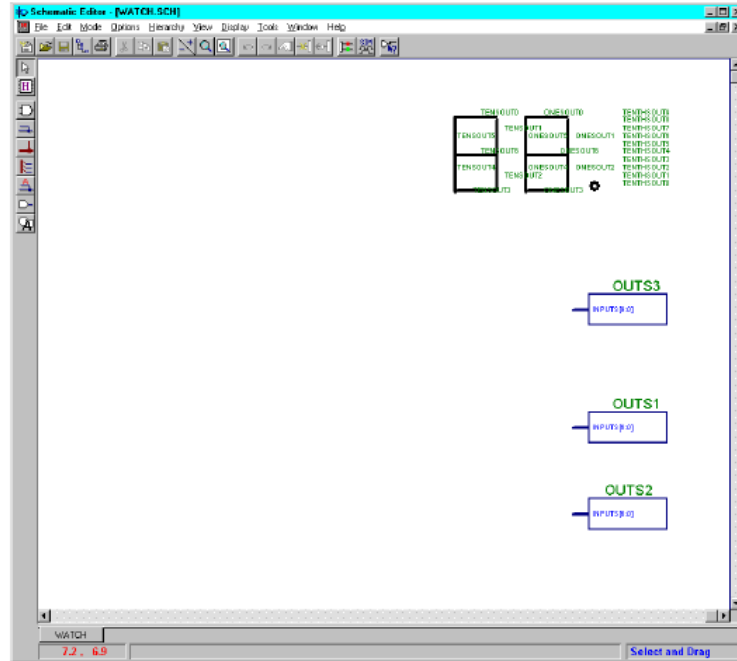
2.5.2 PENYUNTING SKEMATIK

Penyunting ini adalah merupakan suatu bahagian rekabentuk yang utama dalam melaksanakan projek ini. Proses rekabentuk litar dan juga penyelakuan dapat dijalankan pada tetingkap ini. Proses ini menyediakan sokongan rekabentuk yang lebih teratur. Kesemua penyambungan akan disambung secara automatik untuk membolehkan proses penyelakuan dilakukan dengan lebih sempurna. Pada proses ini terdapat penyelaku yang membolehkan pengujian akan dilakukan keatas rekabentuk projek.



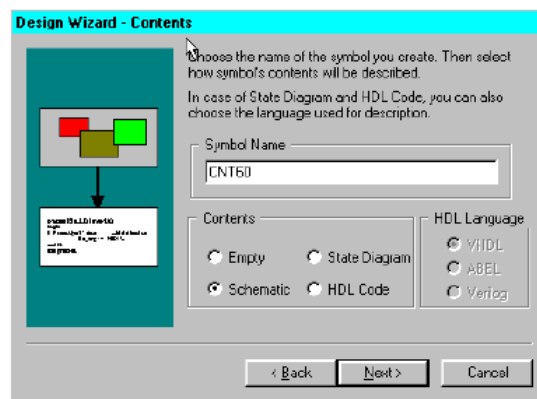
Rajah 2.4: Tetingkap untuk memilih penyunting skematik

Setelah pemilihan dilakukan dengan memilih penggunaan penyunting skematik, maka tetingkap penyunting skematik akan diperolehi.



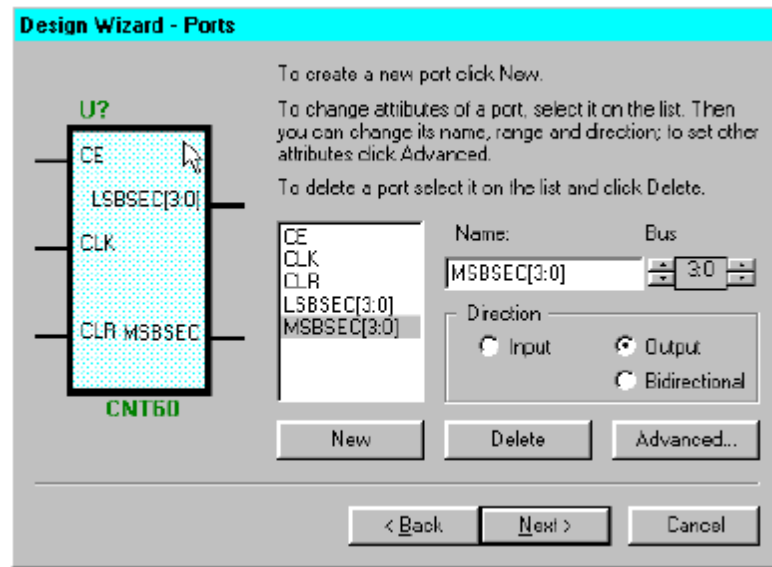
Rajah 2.5: Tetingkap penyunting skematik

Makro boleh dibentuk dengan menggunakan 'symbol wizard' yang terdapat pada tetingkap penyunting skematik.



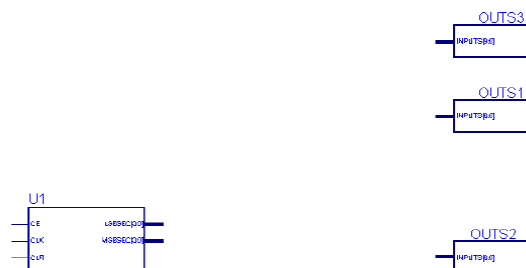
Rajah 2.6: Tetingkap penghasilan makro

Oleh kerana pemilihan adalah berdasarkan skematik, maka skematik dipilih pada tettingkap penghasilan makro. Tettingkap seterusnya membolehkan pemilihan masukan dan keluaran yang diperlukan untuk menghasilkan makro yang diperlukan. Penamaan masukan dan keluaran akan dihasilkan pada tettingkap ini.



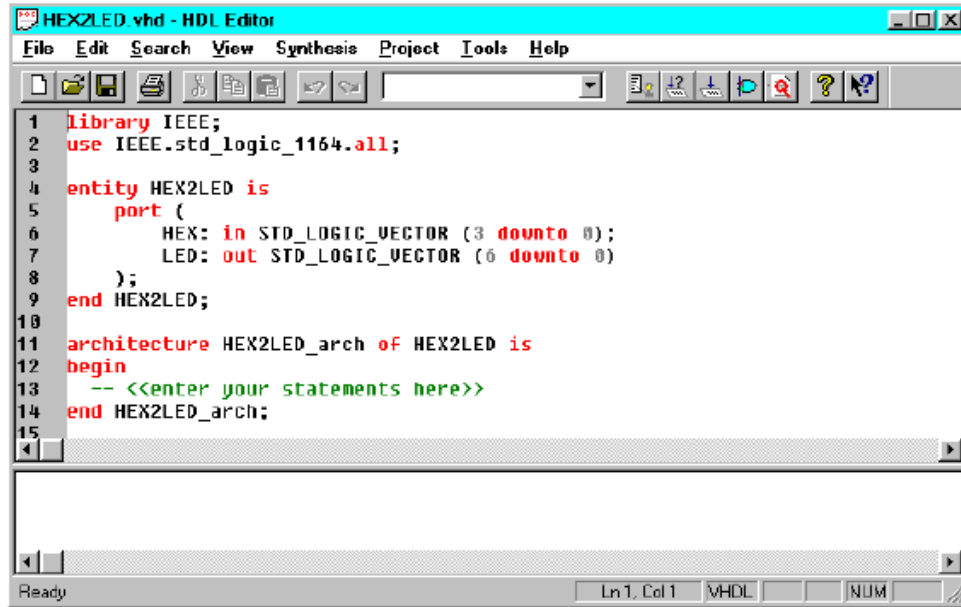
Rajah 2.7: Tettingkap penamaan masukan dan keluaran bagi makro

Setelah tettingkap untuk penamaan masukan dan keluaran dilakukan, maka makro yang dikehendaki akan diperolehi seperti Rajah 2.8.



Rajah 2.8: Makro yang terhasil

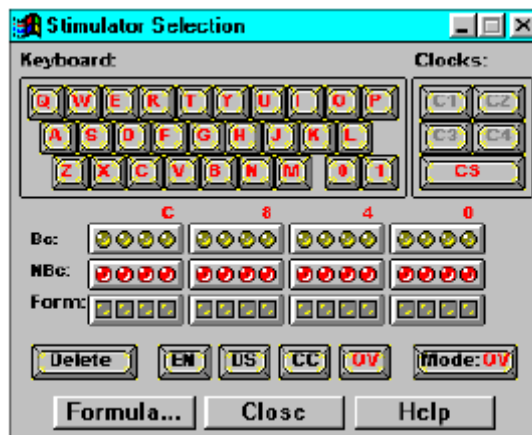
Makro yang terhasil ini akan di masukkan dengan aturcara HDL dengan tettingkap editor skematik pada Rajah 2.4.



Rajah 2.9: Tetingkap aturcara HDL

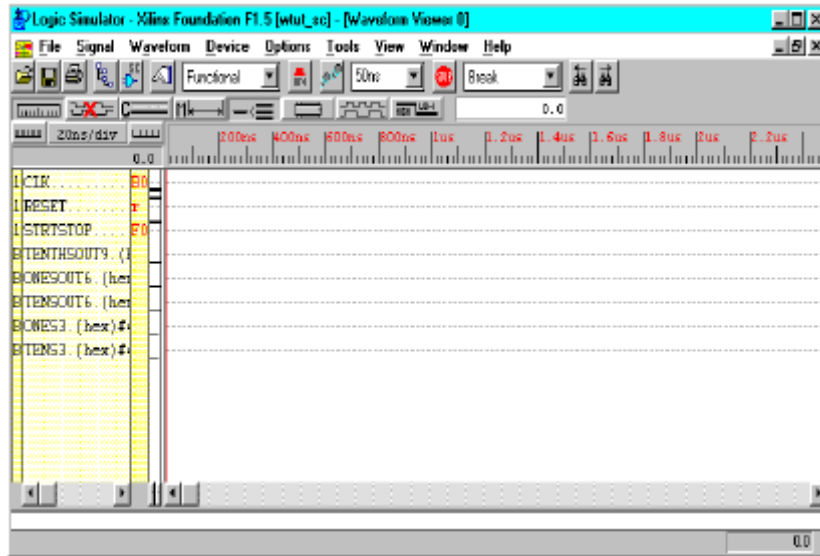
2.5.3 PENYELAKU LOGIK

Proses ini adalah untuk memeriksa samada rekabentuk litar ini dapat direkabentuk dengan keluaran yang dikehendaki berdasarkan masukan yang tertentu.



Rajah 2.10: Tetingkap penyelaku

Tetingkap pada Rajah 2.10 adalah untuk memilih masukan dan keluaran manakala rajah 2.11 adalah untuk melihat keputusan penyelakuan bagi rekabentuk projek.



Rajah 2.11: Tetingkap untuk keputusan penyelakuan

Pada tetingkap ini, keputusan penyelakuan bagi keseluruhan projek dapat dilihat. Ia juga boleh digunakan untuk melihat keluaran pada satu-satu makro yang digunakan atau direkabentuk. Keputusan penyelakuan biasanya dilihat pada satu-satu makro yang direkabentuk bukannya pada keseluruhan litar. Jika keseluruhan penyelakuan litar dilihat, maka kesilapan pada projek ini tidak dapat dikesan dengan cepat.

2.5.4 PENGARAH REKABENTUK

Fungsinya adalah untuk membaca dan menterjemah secara automatik fail-fail rekabentuk dan mengaturncara suatu projek dengan baik.

2.5.5 PENGARAH PUSTAKA

Bahagian ini merupakan bahagian paling penting yang menyimpan segala data yang dihasilkan oleh pengguna dan juga perisian itu sendiri. Jika pustaka sistem ini hilang atau terpadam maka keseluruhan litar yang direkabentuk oleh pengguna tidak dapat dijanakan samada untuk melihat keluaran yang terhasil atau pun untuk melakukan proses pelaksanaan.

2.5.6 PENYUNTING SIMBOL

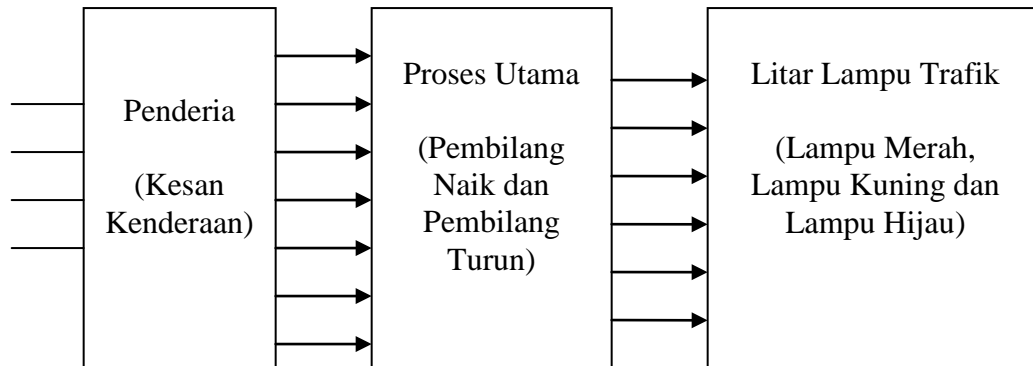
Proses ini adalah untuk membentuk dan menyunting simbol yang akan digunakan pada penyunting skematik. Oleh itu, ia memudahkan pengguna untuk menyunting kembali makro yang terhasil andai makro tersebut perlu ditambah masukan atau keluaran. Ia membolehkan nombor pin, bentuk pin dan juga penamaan makro boleh dilakukan. Proses ini akan membolehkan makro yang mengalami kesilapan sedikit dapat digunakan sekali lagi untuk proses yang diingini oleh pengguna tanpa perlu membuat kembali makro tersebut.

BAB 3

REKABENTUK LITAR

3.1 REKABENTUK LITAR

Peringkat ini adalah merupakan peringkat yang paling utama dalam projek tahun akhir ini. Rekabentuk litar adalah berdasarkan litar skematik. Gambarajah blok sistem pengawal lampu trafik menggunakan penderia adalah seperti dibawah. Gambarajah ini akan memberikan sedikit gambaran untuk merekabentuk projek. Gambarajah blok sistem adalah merupakan permulaan kepada rekabentuk bagi projek tahun akhir ini. Perancangan yang teliti akan dilakukan berdasarkan gambarajah blok sistem ini.



Rajah 3.1: Gambarajah Blok Sistem Pengawal Lampu Trafik dengan Penderia Menggunakan FPGA

3.2 LITAR PEMBILANG MENAIK (0 KE 23)

Litar pembilang ini akan membilang daripada 0 ke 23 dan seterusnya akan kembali ke 0. Litar ini diperlukan untuk membilang selama 24 jam dalam satu hari. Oleh kerana rekabentuk projek ini akan digunakan untuk kiraan selama 24 jam, maka pembilang ini akan digunakan untuk membilang. Jadual 3.1 menunjukkan keadaan sekarang dan keadaan selepas pembilang. Keadaan ini penting untuk merekabentuk pembilang. Keadaan sekarang yang pertama pada jadual tersebut adalah keadaan '00000' dan keadaan selepasnya adalah '00001' maka pembilang ini akan membilang secara menaik dengan kiraan satu berdasarkan perduaan. Keadaan sekarang selepas itu adalah '00001' dan keadaan selepasnya adalah '00010'. Keadaan ini akan berterusan sehinggalah kiraan tersebut sampai ke 23. Apabila kiraan tersebut 23, maka ia akan mula membilang semula dengan kiraan 0 atau '00000' dalam perduaan.

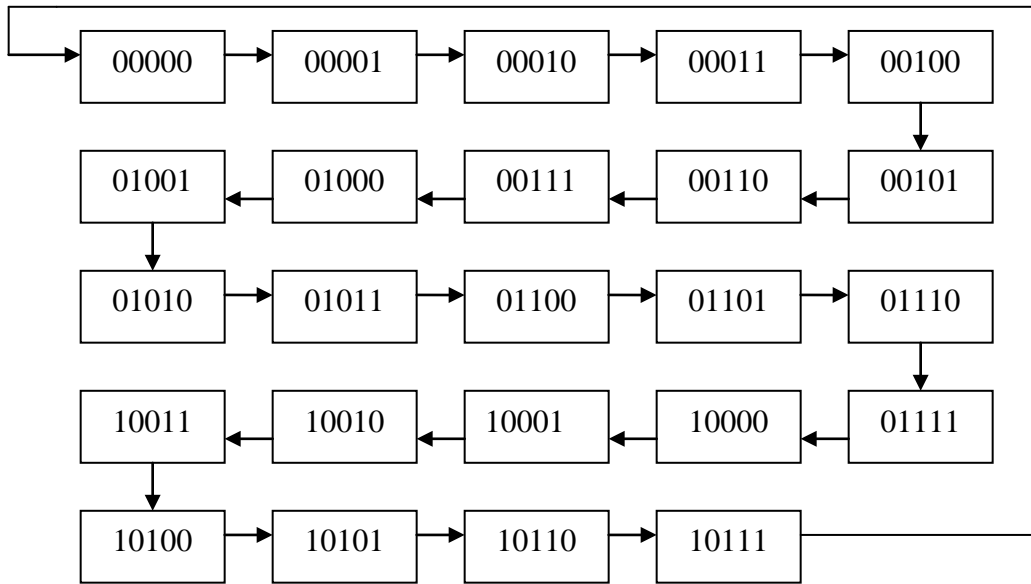
Jadual 3.1: Jadual keadaan sekarang dan selepas bagi pembilang menaik 0 ke 23

<i>KEADAAN SEKARANG</i>					<i>KEADAAN SELEPAS</i>				
E	D	C	B	A	E	D	C	B	A
0	0	0	0	0	0	0	0	0	1
0	0	0	0	1	0	0	0	1	0
0	0	0	1	0	0	0	0	1	1
0	0	0	1	1	0	0	1	0	0
0	0	1	0	0	0	0	1	0	1

PENGAWAL LAMPU TRAFIK DENGAN PENDERIA

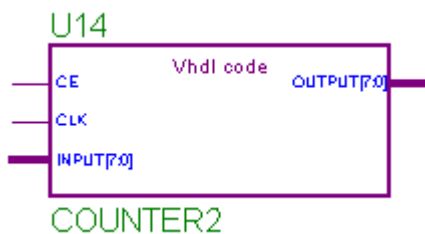
0	0	1	0	1	0	0	1	1	0
0	0	1	1	0	0	0	1	1	1
0	0	1	1	1	0	1	0	0	0
0	1	0	0	0	0	1	0	0	1
0	1	0	0	1	0	1	0	1	0
0	1	0	1	0	0	1	0	1	1
0	1	0	1	1	0	1	1	0	0
0	1	1	0	0	0	1	1	0	1
0	1	1	0	1	0	1	1	1	0
0	1	1	1	0	0	1	1	1	1
0	1	1	1	1	1	0	0	0	0
1	0	0	0	0	1	0	0	0	1
1	0	0	0	1	1	0	0	1	0
1	0	0	1	0	1	0	0	1	1
1	0	0	1	1	1	0	1	0	0
1	0	1	0	0	1	0	1	0	1
1	0	1	0	1	1	0	1	1	0
1	0	1	1	0	1	0	1	1	1
1	0	1	1	1	0	0	0	0	0

Jadual diatas diperlukan untuk melihat keadaan sekarang dan keadaan selepasnya. Jadual 3.1 akan memudahkan pemahaman tentang pembilang yang direkabentuk.



Rajah 3.2: Keadaan sekarang dan selepas untuk pembilang menaik 0 ke 23

Setelah keadaan pada Rajah 3.2 diperolehi, maka rekabentuk akan dilakukan dengan membilang secara menaik. Rajah 3.2 akan memudahkan pemahaman tentang pembilang menaik ini. Oleh kerana pembilang ini akan membilang secara menaik, nilai awal yang telah ditetapkan adalah sifar. Nilai sifar ini adalah nilai masukan untuk kiraan ini bermula. Masukan ini akan ditetapkan dalam rekabentuk ini. Makro pembilang menaik 0 ke 23 adalah pada Rajah 3.3. Rajah 3.4 adalah masukan pembilang menaik yang ditetapkan sifar atau `00000` dalam perduaan. Masukan pembilang ditetapkan sifar kerana ia akan membilang secara menaik.

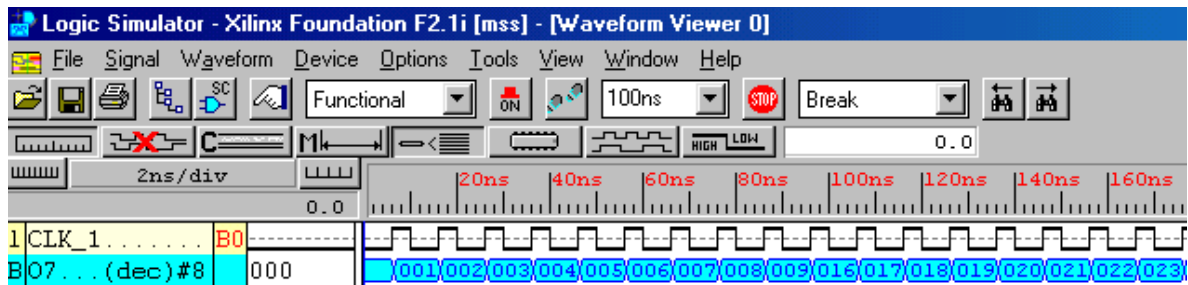


Rajah 3.3: Makro pembilang menaik yang direkabentuk untuk membilang naik 0 ke 23

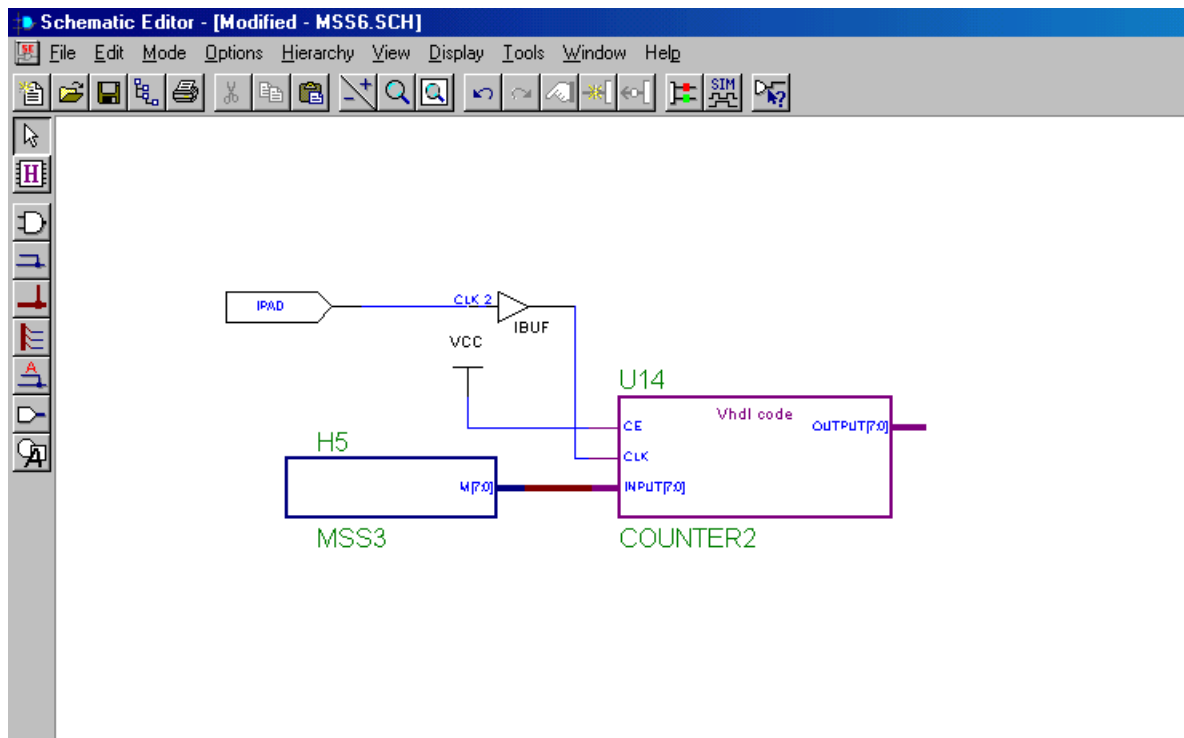


Rajah 3.4: Makro masukan untuk pembilang menaik yang direkabentuk untuk membilang naik 0 ke 23

Rajah 3.5 pula menunjukkan keluaran yang terhasil daripada rekabentuk pembilang menaik tersebut. Keluaran dapat dilihat apabila penyelakuan dilakukan. Setelah keluaran diubah dalam bentuk normal atau desimal, keluaran itu menunjukkan kiraan dikira daripada 0 ke 23. Oleh kerana keluaran yang diperolehi adalah betul dengan keluaran yang ingin direkabentuk, maka pembilang menaik yang direkabentuk telah diperolehi seperti Rajah 3.5. Tetingkap ini akan memudahkan proses rekabentuk ini dilakukan.



Rajah 3.5: Tetingkap penyelakuan bagi pembilang menaik 0 ke 23



Rajah 3.6: Keseluruhan litar makro pembilang menaik dengan kiraan 0 ke 23

Rajah 3.6 menunjukkan keseluruhan litar makro bagi pembilang menaik dengan kiraan 0 ke 23 yang direkabentuk menggunakan kaedah skematik.

3.3 LITAR PEMBILANG MENURUN (40 KE 0)

Litar ini akan membilang secara menurun iaitu daripada 40 ke 0. Pembilang ini akan digunakan untuk membilang masa untuk perubahan warna pada lampu trafik. Jadual 3.2 menunjukkan keadaan sekarang dan keadaan selepas pembilang. Keadaan ini penting untuk merekabentuk pembilang menurun ini. Keadaan sekarang adalah `101000` dalam perduaan atau 40 dalam desimal. Penetapan nilai masukan pembilang yang ditetapkan adalah 40 kerana pembilang ini akan membilang secara menurun.

Jadual 3.2: Jadual keadaan sekarang dan selepas pembilang menurun 40 ke 0

<i>KEADAAN SEKARANG</i>						<i>KEADAAN SELEPAS</i>					
F	E	D	C	B	A	F	E	D	C	B	A
1	0	1	0	0	0	1	0	0	1	1	1
1	0	0	1	1	1	1	0	0	1	1	0
1	0	0	1	1	0	1	0	0	1	0	1
1	0	0	1	0	1	1	0	0	1	0	0
1	0	0	1	0	0	1	0	0	0	1	1
1	0	0	0	1	1	1	0	0	0	1	0
1	0	0	0	1	0	1	0	0	0	0	1
1	0	0	0	0	1	1	0	0	0	0	0
1	0	0	0	0	0	0	1	1	1	1	1
0	1	1	1	1	1	0	1	1	1	1	0
0	1	1	1	1	0	0	1	1	1	0	1
0	1	1	1	0	1	0	1	1	1	0	0
0	1	1	1	0	0	0	1	1	0	1	1
0	1	1	0	1	1	0	1	1	0	1	0
0	1	1	0	1	0	0	1	1	0	0	1
0	1	1	0	0	1	0	1	1	0	0	0
0	1	1	0	0	0	0	0	0	1	1	1
0	1	0	1	1	1	0	0	0	1	1	0
0	1	0	1	1	0	0	0	0	1	0	1