

Ranko Jakšić, Goran Zelić
Elektrotehnički Fakultet, Unska 3, 41000 Zagreb

IGEM: Interaktivni grafički editor za maske integriranih sklopova

IGEM: An Interactive Layout Editor for Integrated Circuits

Sadržaj - Maske integriranih sklopova sastoje se od velikog broja geometrijskih podataka, koji se generiraju pomoću interaktivnih grafičkih editora. Da bi se omogućio brz interaktivni rad, potrebna je baza podataka koja će omogućiti brzo pretraživanje i manipuliranje geometrijskim oblicima. Prikazana je organizacija baze podataka metodom kutnih kazaljki, te njena implementacija u interaktivnom grafičkom editoru za maske IGEM.

Abstract - IC masks consist of a considerable number of geometric data, generated by means of interactive graphic editors. In order to provide fast interactive work, a database is necessary, which will allow for fast retrieval and geometric shape handling. A database organization based on the corner stitching method is described in this paper, together with its implementation within an interactive graphic editor.

1. Uvod

Za topološko, odnosno fizičko projektiranje integriranih sklopova koriste se interaktivni grafički editora kojima se unose podaci o maskama pojedinih tehnoloških slojeva. Stvarna reprezentacija geometrije maski sastoji se od oblika, transformacija i grafičkih atributa. Oblik sadržava informacije o stvarnom prikazu objekta i može se opisati na više načina: pravokutnik, poligon, linija, krug, luk i tekst [1]. Transformacija predstavlja modifikaciju oblika: rotacijom, zrcaljenjem, translacijom i skaliranjem. Grafički atribut služi za razlikovanje pojedinih slojeva maski koji se preklapaju, a to se postiže bojom i teksturom. Interaktivni editora postavljaju velike zahtjeve na interne strukture podataka. Baza podataka mora biti sposobna prihvatiti vrlo velik broj informacija, a istodobno omogućiti efikasne operacije pretraživanja.

2. Metoda kutnih kazaljki

Pretraživanje je operacija koja se najčešće primjenjuje u svakoj bazi podataka i predstavlja dobru mjeru za ocjenu efikasnosti organizacije. U interaktivnim sistemima ova operacija postaje još kritičnija. Kada korisnik pri radu grafičkim editorom želi modificirati

neki objekt prikazan na zaslonu on ga izabere ("pokaže" ga strelicom na zaslonu). Program u takvim slučajevima pretražuje bazu podataka (da otkrije što je izabrano). Zato je vrlo bitno koliko je organizacija podataka prilagođena prostornim pretraživanjima. Dvije najčešće i najvažnije operacije pretraživanja su traženje susjeda nekog objekta i potraga za svim objektima na određenom mjestu ili području. Češće korištene tehnike strukturiranja geometrijskih podataka u programima za izradu maski su: povezane liste, dijeljenje sklopa na kvadrante, maske s kazaljka koje pokazuju na susjede, kutne kazaljke.

Povezane liste su najjednostavniji način da se reprezentiraju maske integriranih sklopova, po jedna lista za svaki tehnološki nivo (polisilicij, difuzija itd). Nove maske jednostavno se dodaju na početak liste, ali npr. traženje susjeda može dovesti do pretraživanja cijele liste. Da bi se ublažio ovaj nedostatak, maske se mogu sortirati po horizontalnim i vertikalnim rubovima. U tom slučaju bitno se komplicira struktura, budući da su potrebne dvije ili čak četiri liste (ako se želi pamtitu redosljed u oba smjera).

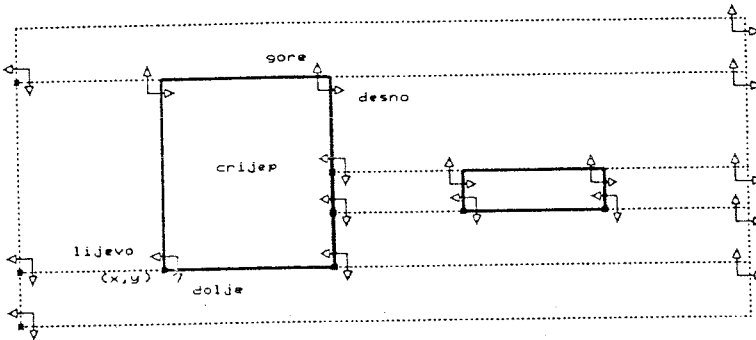
Dijeljenje sklopa na kvadrante (engl. bins). U sistemima u kojima se koristi ova metoda područje sklopa podijeli se na više dijelova (kvadranta), a sve maske unutar jednog kvadranta povežu se u jednostruku listu. Adrese ovih listi čuvaju se onda u dvodimenzionalnom polju. Ovakvom organizacijom pretraživanje se ubrzava jer je potrebno pretražiti samo kvadrante koji sadrže područje od interesa. Problem predstavlja određivanje veličine kvadranta. Ako su kvadranti veliki u odnosu na maske, u svakom kvadrantu biti će mnogo maski i pretraživanje će biti sporo jer će liste biti dugačke. Ako su pak kvadranti mali, pretraživanje će biti brže, ali će se bespotrebno trošiti memorija na velik broj praznih kvadranta. Osim toga ova organizacija ne pomaže mnogo pri traženju susjeda.

Maske s kazaljka koje pokazuju na susjede. Pri operacijama kao što je sabijanje sklopa (engl. compaction) važno je znati najbliže susjede u smjeru koordinatnih osi. Zbog toga se često sreću baze podataka organizirane tako da maska sadrži kazaljke na najbliže susjedne sa svake strane. Iako je pogodna za sabijanje i ovakva organizacija ima mane. Pri pomicanju neke od maski, nemoguće je jednostavno utvrditi koje sve kazaljke treba promijeniti, već treba pregledati cijelu bazu. Osim toga ovakva organizacija nije pogodna za povezivanje jer ne daje podatke o veličini praznog prostora između maski.

Metoda kutnih kazaljki [1, 2] je metoda za strukturiranje geometrijskih podataka naročito primjenljiva na maske pravokutnog oblika čiji su rubovi paralelni s koordinatnim osima (Manhattan geometrija). U opisu za ovakve maske koristimo termin "crijep" (engl. tile).

Osnovna razlika između ove i prije navedenih metoda je da omogućava vrlo efikasno pretraživanje i brze modifikacije baze podataka. Dvije osnovne karakteristike metode su:

- Cijela površina sklopa (tj. ne samo maske već i prazan prostor) je eksplicitno predstavljena u bazi podataka. Razlikuju se dva tipa crepova: crepovi koji predstavljaju maske (puni crepovi) i crepovi koji predstavljaju prazan prostor (prazni crepovi).
- Za povezivanje crepova koriste se kutne kazaljke. Svaki crijep sadrži četiri kazaljke. To su kazaljke "gore" i "desno" (gornji desni kut), te "dolje" i "lijevo" (doljni lijevi kut). Od podataka o crijepu pamti se još njegov tip ("pun" ili "prazan") i koordinate doljnog lijevog kuta (x, y) (slika 1).



Slika 1. Prikaz kutnih kazaljki u polju crepova.

Za svaki tehnološki sloj (npr. metal) postoji prostor ili polje (slika 1) koje pokriva cijelu površinu sklopa u kojem puni crepovi (na slici izvučeni punim linijama) predstavljaju maske, dok ostali crepovi predstavljaju prazan prostor.

U početku je polje prazno tj. sadrži samo jedan vrlo velik prazan crijep. Takav crijep nema susjeda, pa su kutne kazaljke zapravo nul-kazaljke. Dodavanjem punih crepova u bazu prazni crijep se dijeli na više dijelova i tako se dobiva struktura sa slike 1.

Kod metoda koje pamte samo pune crepove podatke o susjedima i praznom prostoru između crepova potrebno je izračunati (što može biti vrlo komplicirano). Upotreba kutnih kazaljki omogućuje da se do ovih podataka dođe brzo i jednostavno jer su crepovi zapravo sortirani u dvije dimenzije. Do podataka o okolini nekog crijepa lako je doći jednostavnim kruženjem po susjednim crepovima prateći kutne kazaljke.

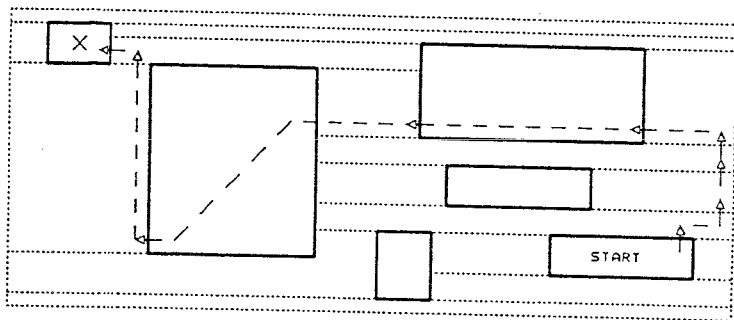
Crepovi se organiziraju u **maksimalno dugačke horizontalne crepove**. To znači da niti jedan crijep nema crijep istog tipa desno ili lijevo od sebe. Ako se usljed neke modifikacije dogodi da se pojave dva crijepa istog tipa jedan pored drugog, treba ih podijeliti da budu jednako visoki i spojiti. Kada se postigne da crepovi budu koliko je god moguće dugački, pokuša ih se spojiti i po vertikali (tj. sa crijepom ispod ili iznad ako je jednako dugačak i istog tipa). Iako bi se prazan prostor mogao pokriti crepovima na više različitih načina, upotrebom ovog pravila dobiva se uvijek jedinstven raspored. Osim toga, ovim spajanjem postižu se uštede u memoriji i mnogo brži algoritmi.

3. Osnovni algoritmi za metodu kutnih kazaljki

Razne vrste pretraživanja, te operacije kreiranja i brisanja postaju vrlo efikasne primjenom metode kutnih kazaljki. Zbog toga su algoritmi opisani u ovom poglavlju vrlo pogodni za interaktivne grafičke editore.

Najvažnija karakteristika svih ovih algoritama je njihov **lokalni karakter**. Svi algoritmi ovise samo o informacijama koje se nalaze u neposrednoj okolini operacije i nije ih potrebno izračunavati.

Nalaženje crijepa koji sadrži točku (x, y) je najčešća operacija pretraživanja. Postupak iterira po x i y osi počevši od bilo kojeg crijepa u promatranom polju (slika 2), kao što je prikazano slijedećim algoritmom:



Slika 2. Pronalaženje crijeva koji sadrži točku X.

- Korak 1: Od prethodno izabranog crijeva idi gore ili dolje slijedeći odgovarajuću kazaljku, dok ne pronađeš crijev koji sadrži y koordinatu tražene točke.
- Korak 2: Idi lijevo ili desno dok ne pronađeš crijev koji sadrži x koordinatu tražene točke.
- Korak 3: Ako prilikom pomicanja u smjeru x osi dođe do pomaka i u smjeru y osi, tada idi na korak 1; inače je crijev, koji sadrži traženu točku pronađen.

Pronalaženje susjeda. Metoda kutnih kazaljki omogućuje vrlo lako nalaženje susjeda nekog crijeva. Svi lijevi susjedi izabranog crijeva C nalaze se primjenom slijedećeg algoritma:

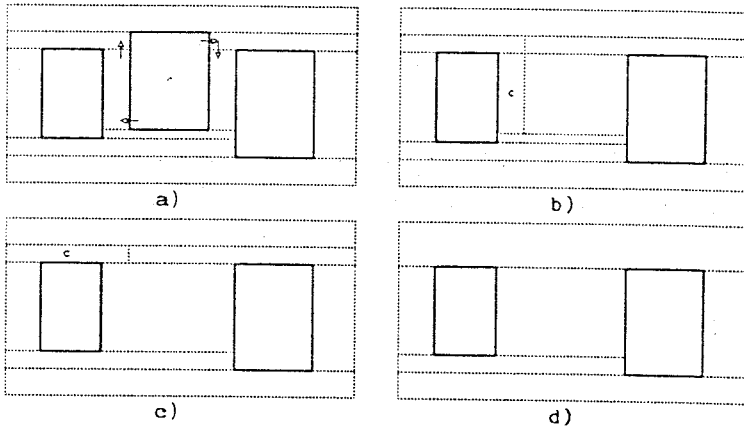
- Korak 1: Kreni od crijeva na koji pokazuje kazaljka lijevo crijeva C. To je najniži lijevi susjed.
- Korak 2: Dok je donji rub susjeda niži od gornjeg ruba crijeva C, prati kazaljku gore.

Na sličan način obilaze se i susjedi s ostalih strana. Potrebno je samo pogodno odabrati prvi crijev, odnosno prvog susjeda od kojeg dalje slijedi pretraživanje.

Prostorno pretraživanje je vrlo česta pretraga, kojom se ispituje da li se na nekom području već nalazi koji puni crijev. Ova provjera ostvaruje se na slijedeći način:

- Korak 1: Pronadi crijev C koji sadrži gornji lijevi kut promatranog područja (koristi se algoritam za nalaženje crijeva koji sadrži točku (x, y)).
- Korak 2: Ako je crijev C pun, tada je to kraj pretraživanja.
- Korak 3: Ako je desni rub crijeva C unutar promatranog prostora, to je rub jednog punog crijeva (zbog pravila o maksimalno dugačkim crepovima). Kraj pretraživanja.
- Korak 4: Izaberi crijev $C1$ na koji pokazuje kazaljka dolje crijeva C. Postavi $C = C1$.
- Korak 5: Ako crijev $C1$ nije unutar promatranog prostora, izaberi crijev C na koji pokazuje kazaljka desno crijeva $C1$;
- Korak 6: Dok je crijev $C1$ različit od crijeva C, postavi $C1 = C$, te ponavlaj korak 5.
- Korak 7: Dok nije donji rub promatranog područja, ponavlaj korak 3.

Brisanje crijepa bila bi vrlo jednostavna operacija (dovoljno je promjeniti samo tip crijepa iz punog u prazan), da nije potrebno sačuvati organizaciju crepova u bazi, tako da bude zadovoljeno pravilo o maksimalno dugačkim horizontalnim crepovima. Algoritam za brisanje crijepa, odnosno stvaranje maksimalno dugačkih crepova primjenjen na crijep C je sljedeći (slika 3):



Slika 3. Tok brisanja crijepa.

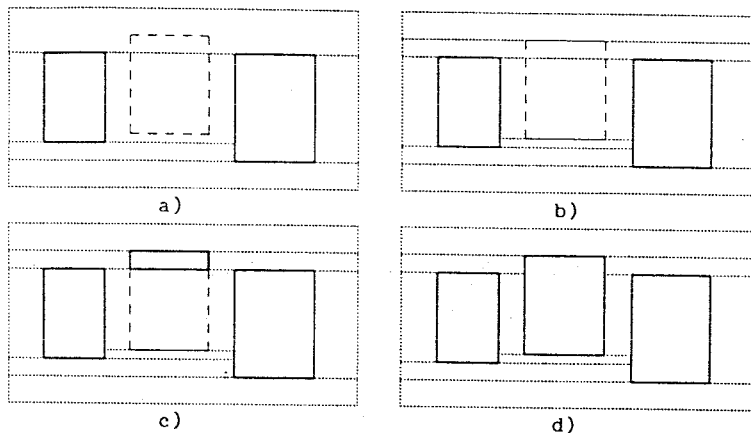
- Korak 1: Promjeni tip crijepa iz punog u prazan crijep.
 Korak 2: Za svaki desni susjed istog tipa, kao što je i crijep C podijeli bilo C, bilo susjeda ili pak oba crijepa tako da budu jednako visoki; nakon toga ih spoji.
 Korak 3: Idi prema gore po svim lijevim susjedima obrisanog crijepa C, i svaki od njih pokušaj spojiti s ostacima starog crijepa C. Tretiraj svaki lijevi susjed kao crijep C u koraku 2. Uz to svaki crijep pokušaj spojiti sa crijepom iznad sebe.

Kreiranje crijepa. Prvi korak u kreiranju novog crijepa je da se provjeri da nema drugih punih crepova na tom prostoru. U slučaju da ima, novi crijep neće biti kreiran.

Novi crijep se ubacuje u bazu podataka tako da se "ureže" u okolne prazne crepove. Dijelovi praznih crepova pokušaju se onda spojiti u maksimalno dugačke horizontalne crepove (slika 4). Algoritam za kreiranje crijepa je:

- Korak 1: Ako prostor za kreiranje novog crijepa nije prazan, javi da crijep ne može biti kreiran.
 Korak 2: Pronadi prazni crijep koji sadrži gornji rub novog crijepa (zbog pravila o maksimalno dugačkim crepovima cijeli gornji rub je sadržan u samo jednom praznom crijepu). Pronađeni crijep podijeli onda po tom rubu na dva dijela.
 Korak 3: Korak 2 ponovi i za donji rub novog crijepa.
 Korak 4: Idi niz lijevi rub novog crijepa i svaki od crepova na koji naiđeš podijeli na tri dijela: jedan na desno, jedan na lijevo i jedan unutar novog crijepa (crijep u sredini je segment novog crijepa i zato tipa pun, a ostala dva crijepa su

prazna). U svakom koraku pokušaj spojiti svaki od ova tri dijela sa crijepom iznad.



Slika 4. Kreiranje crijepa.

Vidljivi susjedi. Metoda kutnih kazaljki omogućuje da se dosta efikasno ostvare i neke kompleksnije operacije. Kao primjer može poslužiti traženje vidljivih susjeda nekog punog crijepa (susjed je vidljiv ako postoji horizontalan (ili vertikalni) pravac koji ga spaja s početnim crijepom a pri tome ne siječe niti jedan drugi puni crijep). Svi vidljivi susjedi desno od nekog crijepa C traže se ovako:

- Korak 1: Izaberi pomoću kutne kazaljke desno najgornjeg desnog susjeda crijepa C.
- Korak 2: Ispitaj sve desne susjede crijepa C (dali su puni ili prazni).
- Korak 3: Ako je susjed pun, onda je automatski vidljiv i prelazi se na slijedećeg susjeda.
- Korak 4: Ako je susjed prazan, provjeri da li se proteže do ruba sklopa. Ako je to istina, nema vidljivih susjeda. U protivnom, crijep ima susjeda ili susjede koji moraju biti puni i prema tome su vidljivi (ali samo ako im je donji rub niže od gornjeg ruba početnog crijepa C).

Kako se crepovi uvijek slažu u najduže moguće horizontalne crepove, traženje susjeda u vertikalnom smjeru se bitno razlikuje od onog u horizontalnom. Traženje susjeda ispod crijepa C se izvodi algoritmom:

- Korak 1: Izaberi pomoću kutne kazaljke dolje najbližijeg donjnjeg susjeda crijepa C.
- Korak 2: Dok je susjed prazan i proteže se duž cijele donjnjeg stranice početnog crijepa, idi dalje prema donjnjem rubu sklopa, prateći kazaljke dolje. Ako naiđeš na rub, nema vidljivih crijepova.
- Korak 3: Ako je puni crijep ili prazan crijep koji se ne proteže duž cijelog donjnjeg ruba crijepa C kreni udesno prateći kazaljke desno. Svaki tako pronađeni puni

crijep je vidljiv. Za prazan crijep pozovi rekurzivno ovaj algoritam (tj. traži donjnje vidljive susjede za ovaj prazan crijep).

Traženje svih lijevih (ili traženje svih gornjih) vidljivih crijepova obavlja se na sličan način.

4. Implementacija

Interaktivni grafički editor za maske integriranih sklopova (IGEM) napisan je za IBM PC kompatibilno računalo, za monokromatski monitor i grafičke kartice rezolucija od CGA do VGA. Podešen je za rad s 5 tehnoloških slojeva nMOS tehnologije (polisilicij, difuzija, implantacija, metal i kontakti), a novi slojevi mogu se jednostavno dodati. Slojevi se međusobno razlikuju šrafurama. Maske su pravokutnog oblika (oblik crijepa) i svi kutevi su 90 stupnjeva (Manhattan geometrija). Moguće su operacije kreiranja, brisanja i modifikacije (rotacija, kopiranje, pomak, zrcaljenje, kreiranje polja itd.).

Program omogućava hijerarhijsko projektiranje sklopa. Svaka ćelija u projektu sastoji se od proizvoljno mnogo podćelija i maski. Operacije s podćelijama omogućuju kreiranje ćelija (pri projektiranju odozdo prema gore), skiciranje ćelija (kreira se prazna ćelija - za projektiranje odozgo prema dolje), učitavanje ćelija, sve modifikacije kao i za obične maske, prikaz sadržaja ćelije i ubacivanje sadržaja podćelije u ćeliju (operacija "flatten" ili izravnavanje). Ukupan broj slojeva crepova u programu je zapravo 6 tj. 5 tehnoloških i jedan za podćelije. Svaki od slojeva u programu je smješten u poseban prostor u kojemu su crepovi povezani kutnim kazaljicama.

Korisničko sučelje je grafičko (slika 5), a naredbe se biraju preko tastature ili mišem.

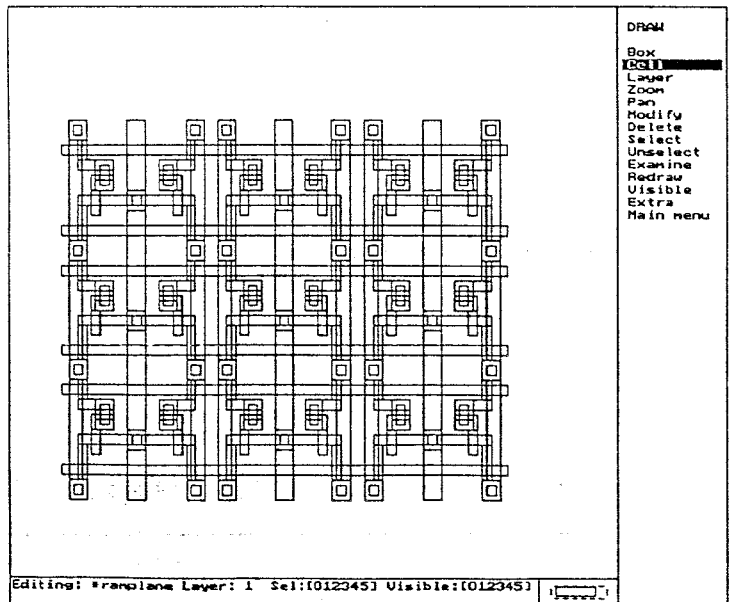
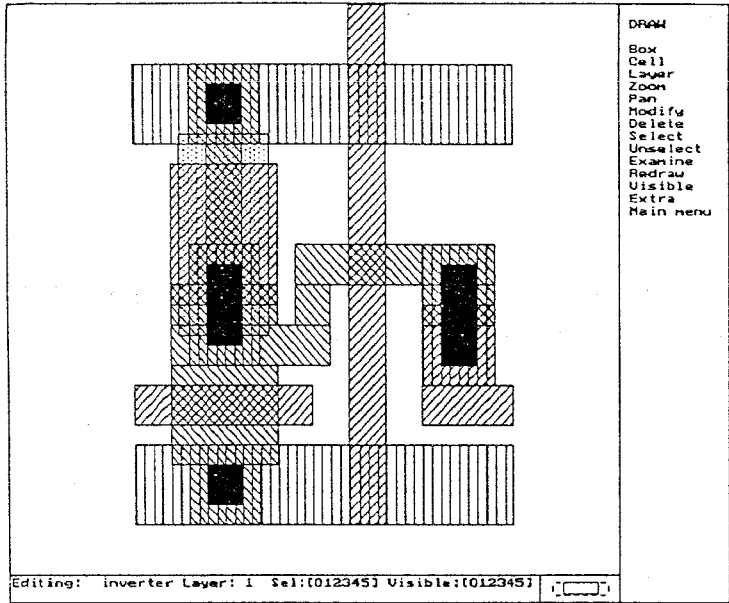
4. Zaključak

Realizirani grafički editor za maske integriranih sklopova (IGEM) pokazao je da se i na PC računalo mogu primjenom metode kutnih kazaljki vrlo efikasno implementirati sve osnovne operacije potrebne za interaktivni rad. Osim toga, moguća je izuzetno efikasna implementacija kompleksnih operacija (kao što su sabijanje ili povezivanje), jer su podaci o praznom prostoru neposredno prisutni u bazi podataka u vidu praznih crepova i kutnih kazaljki.

Osnovna mana metode je da zahtjeva više memorije nego klasični pristupi (prazni crepovi), ali se ovaj problem izbjegava upotrebom hijerarhije u projektiranju, tako da je u memoriji uvijek samo manji dio sklopa. Osim toga primjenom pravila o maksimalno dugačkim crepovima broj praznih crepova ostaje relativno malen (praktički od 1,5 do 2 prazna na svaki puni crijep).

5. Literatura

- [1] S. M. Rubin: "Computer Aids for VLSI Design", Addison-Wesley, Reading Mass., 1987.
- [2] J. K. Osterhout: "Corner Stitching: A Data-Structuring Technique for VLSI Layout Tools", IEEE Transactions in CAD, vol. CAD-3 No.1, pp 87-100, January 1984.



Slika 5. Prikaz zaslona računala pri radu programom IGEM.