

## Fóliautak tervezése

ETO 621.3.049.75.001.2:681,3.06 ESZTER

A fóliatervezés feladata az, hogy a rögzítetten kijelölt helyzetű, egy vonalhoz tartozó alkatrészlábak és csatlakozókivezetések között megtervezze az őket elektromosan összekötő fóliavezetékek útvonalatát, a technológiai előírások betartásával. A technológiai előírások közé sorolunk gyakorlatilag minden olyan korlátozást, amelyre a programnak figyelemmel kell lennie, így a tervezéshez felhasználható nyomtatott rétegek számát, annak a területnek az alakját, amelyen belül kell haladniuk az összekötő vezetékeknek, a fóliaszélességeket, a forrasztási szemek átmérőit és a vezetékek közötti minimális szigetelési hézagot. Ide soroljuk még a táp- és földvezetékek kialakítására (pl. az ún. fésűk alakjára) és azoknak az alkatrészlábakhoz történő hozzávezetésére vonatkozó ajánlásokat is.

A tervezéssel szemben támasztott elsőrendű követelmény az, hogy valósítsa meg valamennyi szükséges fóliaösszekötést. Szokásos ugyan a megvalósíthatatlan összekötéseket átkötésekkel pótolni, de a gyártók ettől a megoldástól általában idegenkednek.

Elvárhatjuk a tervezőktől azt is, hogy a megvalósított nyomtatott áramkörben a hamis működést eredményező zavarjelek keletkezésének valószínűsége a lehető legkisebb legyen. Ehhez kívánatos, hogy pl. a vezetékek rövidek legyenek, ill. hogy két vagy több vezeték legfeljebb csak egészen rövid szakaszon fusson egymás mellett. Kézi módszerrel történő tervezéskor a tervező az áramkör felépítésétől és az alkatrészek elhelyezkedésétől függő sorrendben egyenként tervezi meg az egyes vezetékek útvonalatát, közben mindig szem előtt tartva azt, hogy ne vágja el a később sorra kerülő összekötések megvalósításának lehetőségét, azaz ne zárjon körül később bekötendő pontokat.

Ha olyan helyzet adódik, amelyben egy vagy több összekötést nem tud kialakítani, megváltoztatja az akadályt képező vezetékek útvonalatát. Ha ez sem vezet eredményre, az alkatrészek elhelyezésén változtat célszerű módon.

Az ESZR szabvány szerinti, legfeljebb 24 IC tokot tartalmazó kártya ilyen módszerrel történő tervezése általában két-három óráig tart. Bonyolult kapcsolások esetén azonban ez az idő egy-két nap is lehet. Ha ennyi idő sem elég a maradéktalan megvalósításhoz, általában el szokták fogadni a legkevesebb átkötést tartalmazó megoldást, és nem keresik, hogy a feladat elméletileg egyáltalán megoldható-e, illetve, hogy a kápot megoldás a lehető legjobb-e.

Az automatizált tervezéstől azt várhatnánk, hogy kiküszöbölje a kézi tervezés hiányosságait. Ezzel szemben a gyakorlatban is alkalmazott algoritmusok többnyire vagy megfizethetetlenül lassúak, vagy a kézi tervezésnél sokkal kevesebb eredményesek.

Bár elvileg az egész konstrukciós tervezést egyetlen, igen bonyolult optimálási problémaként kellene kezelni, még a lényegesen egyszerűbb nyomtatáster-

vezési feladatra sem létezik kielégítő algoritmus. Olyan algoritmussal sem rendelkezünk, amely legalább a megoldás egzisztenciájának kérdésére biztos választ tudna adni. Ez annál is meglepőbb, mivel a kérdéssel már több mint egy évtizede foglalkoznak, és a probléma megoldása nagy jelentőségű volna.

### 1. Irodalmi áttekintés

A fóliatervezésre kidolgozott, jelenleg ismert legjelentősebb algoritmusokat hét csoportba oszthatjuk. Közös jellemzőjük, hogy a kézi tervezéshez hasonlóan a minimális szigetelési hézag egyszerű betartását raszterhálón való tervezéssel biztosítják. Ez azt jelenti, hogy a fóliautakat általában rasztervonalon vezetik, bár egyes programok ferde (45 fokos) fóliavezetést is lehetővé tesznek [1, 2, 3, 4]. A legtöbb eljárás kétrétegű nyomtatás tervezésére alkalmas. Többnyire felhasználják azt a kézi tervezésből elért tapasztalatot, hogy a vízszintes és függőleges fóliaszakaszokat a kártya különböző oldalán célszerű vezetni.

Az eljárások első négy csoportját az jellemzi, hogy az összekötéseket előre meghatározott sorrendben, egymás után valósítják meg. Egy összekötés megtervezése a még hiányzó összekötések figyelembevétele nélkül történik. Ebből származik e módszerek legfőbb hátránya, nevezetesen az, hogy a megtervezett vezetékek egy-egy területen annyira összesűrűsödhetnek, hogy könnyen lehetetlenné teszik az ilyen területeken levő lábak még hiányzó bekötéseinek megvalósítását.

Ezen algoritmusok közös problémája a vonalak szétदारabolása olyan összekötésekre, amelyek mindegyike egy pontot köt hozzá egy másikhoz vagy a már előzőleg megtervezett fóliautakhoz. (Egy vonalat tehát pontjai számánál eggyel kevesebb összekötéssel lehet megvalósítani.)

További probléma ezen összekötések sorrendbe állítása [5]. A többféle kínáló alternatívából a növekvő hossz szerinti sorbarendezést alkalmazzák a legelterjedtebben.

Az első csoportba tartozó algoritmusok [2, 3, 6, 7, 8, 9, 10, 11] egy összekötést a lehető legrövidebb — vagy általános esetben a lehető legkisebb költségvektorú — vezetékekkel igyekeznek megvalósítani. A kártyát útvesztőnek tekintik, amelyben egy kiindulási ponttól (bekötendő pont) elindulva a lehető legkedvezőbb úton, az akadályok kikerülésével kell elérni egy kijáratit pontot vagy több ilyen pont közül egyet. Az útvesztőproblémát úgy oldják meg, hogy a kiindulási pontból hullámterjedés-szerűen lépnek minden lehetséges irányban újabb és újabb pontokra, míg el nem érik valamelyik célpontot.

Ezen eljárások előnye, hogy — főleg a tervezés elején — rövid vezetékeket eredményeznek, de ez az esetleges sűrűsödések miatt hátrány is lehet. Előny még, hogy egy adott helyzetben maradéktalan vá-

laszt adnak az egzisztencia kérdésére. Hátrányuk a lassú működés.

A második csoport algoritmusai a kártya egyik oldalán vízszintes, a másikon függőleges szabad — azaz fóliavezetésre felhasználható — szakaszokat keresnek a raszterhálón, amelyek átgalvanizálásokkal csatlakoztathatók egymáshoz, és amelyek megvalósítják a szükséges összekötést [11, 12, 13, 14]. Ismeretes olyan megoldás, amely egy összekötésre vonatkozóan az egzisztencia-kérdésre is biztos választ tud adni [13]. Ezek az eljárások nem garantálják egy adott helyzetben a lehető legrövidebb vezetékalkálítást. Kevésbé sikeresek, mint az előző csoporthoz tartozók, de egy nagyságrenddel gyorsabbak. Általában ezek az eljárások eredményezik a legkevesebb átgalvanizálást.

További csoportot alkotnak azok az algoritmusok, amelyek egy bekötést úgy valósítanak meg, hogy a kiindulási (bekötendő) pontból szisztematikusan igyekeznek haladni az előre kitűzött célpont felé, megkerülve a közbeeső akadályokat [15, 16]. Az ilyen eljárások mind sebesség, mind eredményesség tekintetében az előző két algoritmustípus között helyezkednek el.

A negyedik csoportba olyan algoritmusok tartoznak, amelyek az összekötések kialakításának módszerében ugyan esetleg nem jelentenek újat az előzőekhez képest, de alkalmasak arra, hogy utólag módosítsák a már megtervezett fóliautakat.

Az egyik ilyen eljárás [17] a már megvalósíthatatlannak talált összekötéseket úgy tervezi meg, hogy feloldja a különböző vonalakhöz tartozó fóliautak keresztesítésének tilalmát, majd a legtöbbször keresztesített vezetékkel kezdve újratervezi azokat a fóliaösszekötéseket, melyeken mások áthaladtak. Az ilyen elven működő eljárások igen lassúak, csak egészen kis kártyák tervezésére alkalmasak.

A következő három csoport algoritmusainak közös jellemzője, hogy az összekötéseket egyidejűleg alakítják ki, és ezáltal elkerülik egyes fóliautak szerencsétlen vezetését.

Ezek közül az első algoritmustípust topográfikus szimulációnak nevezik [18, 19, 20]. Az eljárás onnan kapta a nevét, hogy az egyes raszterpontokhoz „magassági” mérőszámot rendel, amely a fóliavezetésre legkívánatosabb, tehát a már lefoglalt pontoktól legtávolabb eső raszterpontok esetében a legkisebb. Ezáltal a kártya fóliasűrűségi modelljét kapja, amelynek segítségével az egyidejűleg tervezett fóliautakat mindig a legritkább helyeken igyekeznek vezetni. A topográfikus szimulációt elsősorban egy rétegen való tervezésre dolgozták ki. Előnyei főképpen nem szabályos felépítésű, például a csak diszkrét elemeket tartalmazó kártyákon a legszembetűnőbbek. Működése elég lassú, tágirányú is nagyobb a többinél. A kapott fóliautakban igen sok az irányváltás.

A következő eljárás [21, 22] az elhelyezést és a fóliatervezést egyetlen feladatnak tekinti. Csak egyforma modulokat — pl. IC tokokat — tartalmazó kártyák tervezésére alkalmas. A tokok és oszlopok közti területeket adott kapacitású huzalcsatornákként kezeli. Az elhelyezést és ezzel egyidejűleg az összekötéseknek a csatornába való szétosztását úgy alakítja ki, hogy lehetőleg egyetlen csatorna se legyen túlte-

ltve. A módszer rendkívül gyors és hatékony. Ez utóbbi tulajdonsága abból adódik, hogy nem választja külön a tervezés két, összefüggő fázisát. Az eljárás a gyakorlatban mégsem tudott elterjedni, mert az elméletileg szabályos struktúra semmilyen perturbációját nem tudja figyelembe venni (pl. diszkrét elemek, különböző fóliaszélességek és szemátmérők stb.).

A korábbiak alapján nem merészség azt kijelenteni, hogy a közeljövőben nem várható olyan hatékony, gyors és főleg az egyes technológiákat tekintve univerzális nyomtatástervező algoritmus megszületése, amely versenyre kelhetne az emberi intuícióval. Így a jövő útja feltétlenül az, hogy megfelelő kártyakonstrukció és technológia alkalmazásával könnyítsük meg a számítógép munkáját. Az utolsó csoportba azokat az eljárásokat soroljuk, amelyek ilyen speciális kártyák tervezésére készültek [23]. Ezek a módszerek egyesítik magukban az automatizált tervezés előnyét: másodperces futási idők alatt viszonylag nagy alkatrész-sűrűség esetén is maradéktalan nyomtatástervet készítenek.

Olyan eljárásokat is ismerünk, amelyeket nehéz volna besorolni a fenti hét csoport valamelyikébe. Ezek ismertetése nem célja a jelen cikknek, de a teljesség kedvéért szerepelnek az irodalomjegyzékben [1, 4, 17, 24, 25, 26, 27, 28, 30, 31].

Végül meg kell említenünk, hogy az általános céllal készült programok között azok bizonyultak a leg-sikeresebbeknek, amelyek az első két eljárástípusból ötvöződtek, és fényceruzával ellátott grafikus kijelző segítségével, interaktív módon, az embert is bevonják a tervezésbe.

A nyomtatástervező algoritmusok részletesebb értékelése és összehasonlítása Nakahara nemrég publikált cikkében található [35].

## 2. Az alkalmazott módszerek

Az ESZTER programrendszer fóliatervező programrészének elkészítése előtt a külföldön elért eredmények gondos tanulmányozásával igyekeztünk felmérni a lehetőségeket. Ezek alapján az alábbi célokat tűztük ki:

- a program univerzális legyen az alkalmazott technológiák (fóliaszélesség, szerhátmérő, minimális hézag) tekintetében;
- legyen alkalmas a szabványos, 24 IC tokot tartalmazó ESZR kártyák megtervezésére;
- egy kártya nyomtatástervezésének költsége ne haladja meg a 2000 Ft-ot;
- ha a fóliautak szomszédos toklábak közti átvezetése megengedett, akkor a program lehetőleg maradéktalanul tervezze meg a kártyákat.

### 2.1 Az árnyékolási rendszer

A fenti célkitűzések közül az első jelentette a legtöbb problémát. Megoldása az ún. árnyékolási rendszer és az ezt lehetővé tevő tárolási módszer kidolgozásával sikerült. Az alkalmazott tárolási szisztéma két alapvető adattömbre támaszkodik. A számítási idő lerövidítése céljából ezeket a tömböket mindvégig a főtárban tároljuk.

Az első, a FOLIAK nevű adattömb a már megtervezett vezetékeket írja le, felsorolva nevezetes pontjainak (lábpontok, töréspontok, átgálvanizálások) koordinátáit. Minden ponthoz további információkat is tárol az öt követő fóliaszakasz szélességéről, továbbá arról, hogy csatlakozókivezetés-e, és hogy van-e benne átgálvanizálás. Mindezek az adatok pontonként három szótagot (byte) foglalnak el a tárban.

A PANEL tömb a kártya egyes raszterpontjairól tárol olyan adatokat, amelyek az összekötések megtervezéséhez szükséges kereséseket teszik igen gyorsá. A két réteg ugyanazon koordinátájú pontjához egy szótagban őriz különféle információkat. Ilyen információ például az, hogy halad-e át fólia az illető ponton, és ha igen, az az éppen kialakítandó vonalhoz tartozik-e.

Az árnyékolás fogalmának bevezetését az indokolta, hogy különféle szélességű fóliaösszekötések és különféle átmérőjű forrasztási szemek lehetnek a kártyán, amelyek között mindenképpen meg kell tartani a minimális szigetelési hézagot. A fóliautakról a középvonalukba eső raszterpontok megfelelő PANEL szótagjaiban készül feljegyzés. Egy vonal valamely összekötésének kialakításánál azonban nem elég elkerülni a többi vonal ilyen pontjait. A szükséges hézag betartásához ezen pontoknak a fólia-szélességtől vagy szemátmérőtől függő környezetben sem haladhat más vonalhoz tartozó vezeték. Ezeket a fóliavezetéstől letiltott pontokat — amelyekről speciális bejegyzés készül a PANEL tömb megfelelő szótagjaiba — nevezzük árnyékpontoknak.

Minden vonal tervezésekor el kell tüntetni az illető vonal pontjait körülvevő „árnyéket”, de továbbra is figyelembe kell venni a többi vonal árnyékpontjait. Megkülönböztetett bejegyzést kapnak a PANEL tömbben azok a pontok, amelyek egyidejűleg több vonalnak is árnyékpontjai (többszörös árnyékpontok), mert az ilyen árnyéket sohasem szabad eltüntetni. Az előzőekhez hasonló mondható el a fóliavezetés közben szükséges átgálvanizálásokkal kapcsolatban is.

Az idegen vonalak középvonalbeli raszterpontjai körül átgálvanizálási árnyékpontnak kell feltüntetni a PANEL tömbben mindazon pontokat, amelyekben a tervezett vonal fóliaösszekötéseinek kialakítása során nem szabad átgálvanizálni. A fentiek mintájára megkülönböztetjük a többszörös átgálvanizálási árnyékpontokat is.

Az itt leírt rugalmas árnyékolási rendszer nagymértékben leegyszerűsíti és felgyorsítja az összekötések megtervezésénél a fóliautak és az átgálvanizálások számára szabad raszterpontok megkeresését.

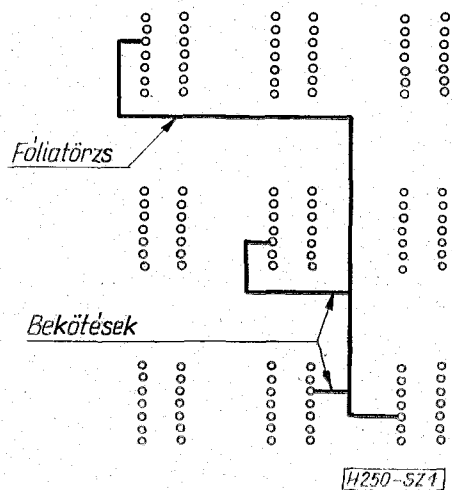
## 2.2 A fóliatervezés menete

A tervezés első fázisa a táp- és földvezetékek kialakítása. Az alapvezetékek — ezeket alakjuk miatt fésűknek nevezik — adatai a kártyakönyvtárban szerepelnek. Ezekhez a fésűkhöz köti be a program a tokok szabványos táp- és földlábait, a kapuk nem használt bemeneteit és a tervező által megjelölt többi lábat.

A második tervezési fázisban egy szakaszonként kereső algoritmus kísérli megtervezni az összekötéseket, különböző oldalon vezetve a vízszintes és függő-

leges szakaszokat. Az eljárás rendkívül gyors. A megvalósított program átlagban fél másodpercenként képes megtervezni egy összekötést. A tapasztalat szerint ezen tervezési fázis végén a bekötések 10–20%-a marad megvalósíthatatlannal.

Az eljárás két lépésből áll. Az első lépésben a vonalak  $y$  irányban legtávolabb eső két-két pontját köti össze a távolságok növekvő sorrendjében. Ezáltal fóliatörzseket képez, amelyekhez a második lépésben igyekszik hozzákötni a vonalak további pontjait. Egy ilyen módszerrel tervezett vonal látható az 1. ábrán.



1. ábra. Négy pont összekötési vonala

Két pont közti összekötést legfeljebb öt szakaszból — három vízszintesből és két függőlegesből — alakítunk ki oly módon, hogy az összekötő vezeték ne hagyja el a két ponton át húzott vízszintes által közrezárt területet. Ennek az alakzatnak a kiválasztását az indokolja, hogy egy IC tokokat sor—oszlopos elrendezésben tartalmazó kártyán — rendkívüli akadályokkal nem számolva — minden esetben össze lehet kötni két toklábát ilyen útvonalon (lásd az 1. ábrát).

Az eljárás először vízszintes szakaszokat igyekszik fektetni a pontokon keresztül, majd ezen szakaszok minden pontjából függőleges szakaszokat indít. Végül keres egy vízszintes összekötő szakaszt a függőlegesek között.

A második lépésben egyenként végighalad a program a vonalakon és megpróbálja megvalósítani a vonalak hiányzó bekötéseit. Azok a vonalak kerülnek előbb sorra, amelyek bekötetlen pontjai a legközelebb helyezkednek el a fóliatörzshöz. A már megtervezett fóliatörzs és a vonal egy további pontja között legfeljebb három szakaszból — két vízszintesből és egy függőlegesből — álló összekötő utat keresünk, mivel várható, hogy az esetek többségében ilyen létezik. Elsőnek vízszintes szakaszt indítunk a bekötendő pontból, majd ennek minden pontján át egy-egy függőlegest. Végül vízszintes összekötést keresünk a fóliatörzs és valamelyik függőleges szakasz között.

A tervezés harmadik fázisa a Lee-algoritmust [6, 7] alkalmazza a még meg nem valósított összekötések kialakítására. Ezt vonalanként végzi, szükség esetén két lépésben.

A két lépés között csak az a különbség, hogy míg az első a vízszintes és függőleges szakaszokat a kártya különböző rétegén vezeti, a második már figyelmen kívül hagyja ezt a megkötést.

A Lee-algoritmus lényegében lassabb, mint az előző eljárás — a programban megvalósított változat általában harminc másodpercenként tervez meg egy összekötést —, viszont képes minden lehetséges főlíut megkeresésére.

Az eljárás ún. hullámfront terjesztéssel kezdődik. Ennek során újabb és újabb hullámfrontokat képez, és az egyes hullámfrontok pontjainak ciklikusan az 1, 2, ill. 3 távolságkódokat adja.

Kiinduláskor a vonal egyik, még bekötetlen pontja alkotja a hullámfrontot. Az új hullámfrontok mindig az előző hullámfront pontjainak távolságkóddal még nem rendelkező szabad szomszédaiából alakulnak ki. Ha egy pont nem átgálvanizálási árnyékpont, a másik rétegbeli megfelelője is megkapja ugyanazt a távolságkódot, és bekerül ugyanabba a hullámfrontba.

A terjesztési eljárás addig tart, amíg meg nem jelenik a hullámfrontban egy olyan célpont, amely az illető vonalhoz tartozik, de nem pontja a kiindulási hullámfrontnak. Ekkor a célponttól kiindulva és a távolságkódokat fordított sorrendben követve kijelölhető az új összekötés. Ezt úgy végzi el a program, hogy a fóliaút lehetőleg minél kevesebb átgálvanizálást tartalmazzon.

Ha még hiányzik összeköttetés a vonalból, előlről kezdjük a hullámfront terjesztését. Ha a legutóbbi célpont az előző fázis által megtervezett vonalrészlethez tartozott, az új kiindulási hullámfront egy újabb bekötetlen pont lesz. Ha nem, a most megtervezett vonalrészlet pontjaiból indul majd a következő terjesztés.

A két lépés közötti, már említett különbség lényegében egy pont szomszédainak meghatározásakor jelentkezik. Az első lépés során a forrasztási oldali pontoknak csak az azonos abszcisszájú, az ültetési oldal pontjainak pedig csak az azonos ordinátájú szomszédai vesznek részt az új hullámfront képzésében. A második lépésben ilyen megkötés nincs. Szükséges még megjegyezni, hogy a fóliatervező program felülbíráhatja az elhelyező program csatlakozókiosztását (a tervezőét természetesen nem), ha a tervezés során egy szabadon maradt csatlakozókivezetést kedvezőbb helyzetűnek talál, mint a vonalhoz kiosztottat.

Az összekötéstervezés végén sornymotatón kiírjuk a sikertelen összekötések listáját, valamint a nyomtatási kép betűkből és jelekből kirajzolt vázlatát. A tervezés eredménye mágnesszalagra kerül, így lehetőség van a huzalozás utólagos, manuális módosítására.

### 3. Összefoglalás

A fentiekben az ESZTER programrendszer nyomtatástervező programrészét ismertettük. A program alkalmas arra, hogy az ESZR szabvány szerinti kártyák fóliaösszekötéseit megtervezzék, mivel a megfelelő tárolási rendszer kialakításával 16 130 raszterpontot tud kezelni. A tárolási módszer azt is lehetővé

teszi, hogy a tervező tetszés szerint megadható — legfeljebb háromféle — fóliaszélességet, ill. forrasztási szemátmérőt, továbbá a gyártási technológia által megkívánt minimális szigetelési hézagot alkalmazzon a kártyán. Ez utóbbi tulajdonsága révén a program változtatás nélkül felhasználható a különböző technológiákhoz.

A programot több gyakorlati példán kipróbáltuk. A mintapéldák ESZR szabványú, általában 20 IC tokot, 15 diszkrét elemet és 200 összekötést tartalmazó kártyák voltak. A kártyákon előzőleg az elhelyező program alakította ki az alkatrészrendezést.

A tapasztalat azt mutatta, hogy a program maradóképtelenül megtervezte mindegyik kártyát, ha megengedtük, hogy két szomszédos tokláb között is vezethessen fóliát. Egy kártya tervezési ideje általában 10 perc, ennek megfelelően a nyomtatástervezés gépidőkölttsége kb. 1400 Ft volt.

A szerző köszönetét fejezi ki Lambert Teréziának az algoritmusok kidolgozásában és a program elkészítésében nyújtott értékes segítségéért.

### I R O D A L O M

- [1] *Bitz, D.*: Algorithmen zum Festlegen von Verdrahtungswegen auf zweiseitig geätzten Platten. Informations Fernsprech — Vermittlungstechnik, 1969. Heft 4, pp. 213—217
- [2] *Campagna, R.*: Computer-aided design of high density printed circuit cards. GTE Report, 1971
- [3] *Heiss, S.*: A path connection algorithm for nmlti-layer boards. Proc. of the 5th Design Automation Workshop, 1968. pp. 6—1...6—14
- [4] *Rozenberg, D. P.*—*Rupp, J. S.*: The automatic routing of multiple plane wiring. IBM Report No. 64—825—1159, 1964
- [5] *Abel, L. C.*: On the ordering of connections for automatic wire routing. IEEE Trans. on Computers, November, 1972. pp. 1227—1233
- [6] *Lee, C. Y.*: An algorithm for path connections and its application. IRE Trans. on EC, September, 1961. pp. 346—365
- [7] *Ackers, S. B.*: A modification of Lee's path connection algorithm. IEEE Trans. on EC., February, 1967. pp. 97—98
- [8] *Kadis, R. T. W.*—*Thompson, K. L.*: Building block programs for the layout of printed circuit boards utilizing integrated circuit packs. Proc. of the 5th Design Automation Workshop, 1968. pp. 5—1...5—16
- [9] *Majorani, E.*: Simplification of Lee's algorithm for special problems. Calcolo, 1964. pp. 247—256
- [10] *Saedler, J.*—*Schäller, K.*: Zu einigen problemen der rechnerunterstützten Konstruktion von Leiterplatten. Nachrichtentechnik, Heft 4, 1970. pp. 153—160
- [11] *Glem, R. R.*—*Lathrop, J. W.*: Two approaches to the computer routing of interconnections. Proc. of 20th Electronic Components Conference, 1970. pp. 390—411
- [12] *Aramaki, I.*—*Kawabata, T.*—*Arimoto, K.*: Automation of Etching-Pattern Layout. Communication of the ACM, November, 1971. pp. 720—730
- [13] *Mikami, K.*—*Tabaehi, K.*: A computer program for optimal routing of printed circuit conductors. IFIP Congress, 1968. pp. H47—H50
- [14] *Shostack, K.*: Computerized methods for the routing of printed circuit boards. Computer Design, July, 1970. pp. 80—87
- [15] TOPEX: Computer software for printed circuit layout generation.
- [16] *Galy, P.*—*Ghendrih, P.*—*Guillaume, G.*—*Ombredane, E.*—*Wolff, A.*: Tracé automatique des circuits imprimés à l'aide d'un ordinateur. L'Onde Électrique, Jan. 1969. pp. 113—119

- [17] *Vincent-Carrefour, J. J.*: Design optimization of small logical system. Proc. of the 23rd National ACM Conference, 1968. pp. 379—386
- [18] *Fisk, C. J.—Caskey, D. L.—West, L. E.*: ACCEL: Automated Circuit Card Etching Layout, Proc. of the IEEE, November, 1967. pp. 1971—1982
- [19] *Fisk, C. J.—Caskey, D. L.—West, L. E.*: Taking the puzzle out of p-c design, Electronics, September 4, 1967. pp. 72—82
- [20] *West, L. E.—Caskey, D. L.*: Topographic simulation as an aid to printed circuit board design. Sandia Lab. Report SC—RR—66—424, 1966
- [21] *Hashimoto, A.—Stevens, J.*: Wire routing by optimizing channel assignment within large apertures. Proc. of the 8th Design Automation Workshop, 1971. pp. 155—169
- [22] *Alia, G.—Di Giacomo, V.—Frosini, G.—Maestrini, P.*: Tracciamento automatico di circuiti stampati secondo una schema predeterminato. Nota Interna B 72—7, 1972. IEI Pisa
- [23] *Hosking*: The use of a very fast routing algorithm for printed circuit board design. The Marconi Review, 3rd quarter, 1971. pp. 207—226
- [24] *Crockford, L. E.—Maller, V. A. J.—Carnell, P. E.*: Procedures for the placement and interconnection of integrated circuits in digital systems. IEE Conference Publication 32, pp. 1—8
- [25] *Dunne, G. V.*: The design of printed circuit layouts by computer. Proc. 3rd Australian Computer Conference, 1966. pp. 413—423
- [26] *Hitchcock, R. B.*: Cellular wiring and the cellular modeling technique. Proc. of the 6th Design Automation Workshop, 1969. pp. 25—43
- [27] *Kodres, U. R.—Lippmann, H. E.*: SLT board layout. IBM Tech. Rep. TROO. 1010, 1964
- [28] *Lass, S. E.*: Automated printed circuit routing with a stepping aperture. Comm. of the ACM, May, 1969. pp. 262—265
- [29] *Mah, L.—Steinberg, L.*: Topologic class routing for printed circuit boards. Proc. of the 9th Design Automation Workshop, 1972, pp. 80—93
- [30] *Sami, M.*: Un programma per il tracciamento automatico dei circuiti stampati. Alta Freguenza, 1968. No. 4. pp. 293—298
- [31] *Rose, N. A.*: Computer aided design of printed wiring boards. Ph. D. Thesis, University of Edinburgh, 1970.
- [32] *Atiayh, J.—Wall, P. K.*: Practical layout of printed circuit boards using interactive graphics. Computer Graphics '70 Conference, 1970
- [33] *Leevers, D. F. A.*: The use of a graphical display in the automatic design of printed circuit boards. International Conference on CAD, 1969. pp. 11—20
- [34] *Pezé, F. A.*: Program for semi-automatic tracing of printed circuit connections. International Conference on CAD, 1969. pp. 89—96
- [35] *Nakahara, H.*: Computer-aided interconnection routing: General survey of the state-of-the-art. Networks, Vol. 2. 1972. pp. 167—183