

Autorouter v editoru schématu

V prvním dílu našeho minicyklu věnovaného možnostem schematického editoru Formica [1] jsme demonstrovali, jak je možno pomocí tzv. zamčení netlistu zabránit nezamýšleným změnám zapojení. Na podstatně elementárnější úrovni obdobná snaha zachovat zapojení vedla autorovy snad všech schematických editorů již dávno k vytvoření funkce (obvykle pojmenované *Drag*), která dovoluje odtahovat součástky (nebo též bloky), aniž by zpřetrhala vodiče. Často je však přítom umístí pod obecným úhlem a/nebo naklade přes sebe, což stejně vyžaduje další rozsáhlé ruční editace. Tento přístup navíc neodpovídá koncepci editorů Formica: „co vypadá jako vodivé spojení, je vodivé spojení“. Formica verze 4.40 se proto ve funkci *Drag* snaží o více – zachovat schéma nakreslené elektricky správně, navíc pokud možno co nejúhledněji nebo alespoň „nejčitelněji“. To byl hlavní motiv postupné implementace autorouteru i do schematického editoru, kde může být zároveň používán také samostatně při vkládání nových spojů.

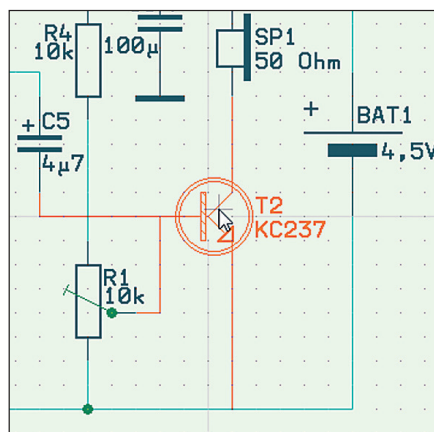
Při editaci schématu se (z estetických důvodů) zpravidla součástky i celé bloky posouvají v rastru, který je poměrně hrubý (často se rovná právě nejmenší vzdálenosti ručně zakreslených vodičů), a zároveň se spoje složené z vodorovných a svislých čar pochopitelně upřednostňují oproti spojům vedeným pod obecným úhlem. Kombinace uvedených dvou požadavků měla jasné důsledky:

- 1) Při tvarování rovnoběžných spojů způsobil nežádoucí propojení i ten nejmenší možný posun ve směru kolmém ke spojům; současně tvarování na sebe kolmých skupin rovnoběžných spojů nebylo vůbec myslitelné.
- 2) Vodiče často překřížily objekty jiných typů, nejčastěji popisy součástek nebo návěští.

Zavedením autorouteru se situace podstatně zjednodušila. Uživatel si jistě může do schématu vnést nezamýšlená propojení, pokud přenášenou součástku nebo blok položí přes jinou část schématu. Tehdy ale je na nově vznikající spoje upozor-

nen a může jejich vytvoření i zakázat. Pokud však umístí součástku nebo celý blok do volného prostoru na výkresu, „inteligentní rozmístování spojů“ zabrání tomu, aby se vytvořila nepatřičná spojení mezi přípojnými vodiči bloku navzájem nebo mezi nimi a prvky okolního schématu.

Autorouter je možné použít samostatně pro pokládání vodičů spojujících dva body a i pro propojení dvou bodů s využitím již existujících spojů. Ve druhém případě



Obr. 1 Automatické rozložení vodičů v okolí jednoduché součástky

(režim „Connect“) tedy může (ovšem jako extrém) být výsledkem operace i pouhé doplnění propojovací značky do patřičného křížení vodičů, zatímco v prvním případě (režim „Route“) se vždy zakreslí celý nový vodič propojující zadané body.

Požadavky na autorouter ve schematicém editoru jsou pravděpodobně méně přísné než při návrhu DPS:

- 1) Vedle uchování konektivity je důležité, aby se vodiče určitým typům objektů zcela vyhýbaly, jiné mohou být vodičem kříženy pouze v některém směru.
- 2) Spoje jsou vedeny vodorovně a svisle v určené minimální vzájemné vzdálenosti.
- 3) Lepší přehlednosti schémat se dosahuje minimalizací počtu ohybů vodičů a teprve s nižší prioritou volbou kratší varianty.
- 4) I když je vhodné, aby se spoje, které ve výchozím stavu vedly souběžně, vytvarovaly opět do souběžného uspo-

RNDr. Ivo Křivka, CSc.

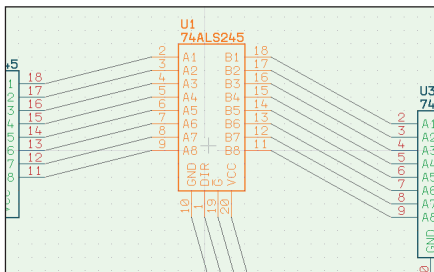
řádání, není to nezbytné; autorouter nemusí sledovat ani přeslechly ani izolační vzdálenosti.

- 5) Na druhé straně má ale ve schématu autorouter k dispozici jen jedinou vrstvu.

A jak ve schematicém editoru Formica autorouter funguje? Leeho algoritmus se pro schémata příliš nehodí, má totiž tendenci vytvářet zubaté spoje, a ani penalizace ohybu tuto tendenci úplně nenapraví. Jako vhodnější se ukázal postup založený na generování polopřímkových sond. Základní princip je odvozen od algoritmu Mikamiho a Tabučiho [2]. Použitý algoritmus upřednostňuje spoje s menším počtem ohybů. Nejvyšší počet ohybů je možno omezit; ukazuje se, že v běžných schématech bohatě postačuje šest ohybů. Z různých nalezených alternativ pro stejný počet ohybů se vybírá spoj nejkratší a případně ten s nejmenší penalizací za křížení. Propojovací algoritmus je modifikován tak, že koncové objekty mohou být buď body (přípojná místa vývodů součástek) nebo celé úsečky (vodiče).

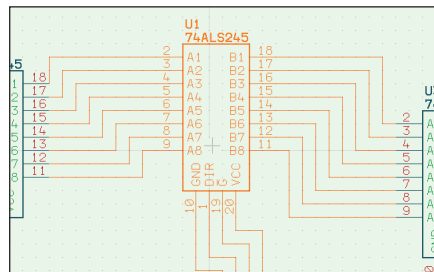
Propojovaných koncových objektů může být více než dva; propojování začíná od těch, které dělí nejkratší (manhattan) vzdálenost. Další důležitý koncept souvisí s generováním uzlových bodů (bodů, kde se navrhované vodiče mohou lámat) tzv. rektilineárního gridu. Potenciálními uzlovými body v něm jsou všechny průsečíky vodorovných a svislých přímk procházejících význačnými body schématu (konce segmentů vodičů, konce vývodů, obrysy pouzder). Speciální algoritmus pak z nich vyloučí ty, které jsou odstíněny neprůchodnými překážkami (urychluje a zároveň zlepšuje vizuální výsledek). Před routováním se spoje seřadí – zejména v případě souběžných spojů má pořadí umístování významný vliv na uspořádanost výsledku, což zlepšuje nejen estetický dojem (viz též obr. 4).

Co se děje při operaci Drag? Jakmile se objekt (blok) na kurzoru při tažení zastaví, začne se na pozadí automaticky generovat rozložení přípojných vodičů pro okamžitou polohu. Pokud je blok znovu posunut před ukončením běhu routeru, částečný výsledek se zahodí. Trvá-li zastavení dostatečně dlouho, zobrazí se namísto přímých spojení konců tvarovaných vodičů (v šedé barvě) rozložení vodičů navrhované routerem (v barvě kurzoru a objektů přesouvaného bloku). V průběhu dalšího posunu bloku se opět nejprve zobrazí šedé propojovací „pružné“ spoje a po zastavení se zase rozběhne router atd. K dispozici jsou jak režim „Route“, tak i „Connect“



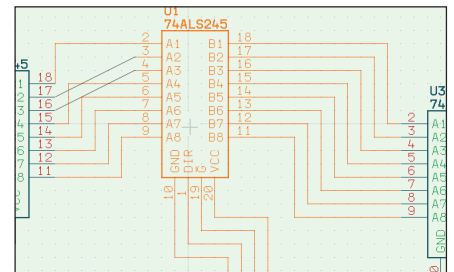
Obr. 2 Zobrazení „pružných“ spojů během posuvu nebo činnosti autorouteru

vodní či jiné skupiny objektů) – při něm se „gumičky“ přetvarují zase do běžných vodičů, kdykoliv je to možné. Druhá využitá skutečnost, že schematický editor obsahuje autorouter také jako samostatnou operaci, jejímž operandem může být libovolná množina vodičů (označené vodiče jsou odstraněny a všechny jejich přílehlé přípojné body jsou znovu automaticky propojeny). Stačí tedy označit „gumičky“ jakožto operand a spustit router (k tomu je připraveno makro <Ctrl-G>). V případě potřeby mohou být operandem i všechny vodiče ve schématu. Výsledek routování lze ovlivnit nastavením parametrů, zejména je třeba vhodně volit



Obr. 3 Rozložení několika skupin paralelních vodičů navrhované autorouterem

Závěrem lze konstatovat, že k výsledku činnosti autorouteru mohou z hlediska estetického existovat výhrady, někdy možná plynoucí z principiální nedůvěry k umělé inteligenci, ale častěji jistě oprávněné. To není překvapením, protože i přes poměrnou složitost a výpočetní náročnost se autorouter řídí jen několika jednoduchými pravidly, a těžko může plně nahradit uvažování návrháře (jenž se leckdy také snaží polohou a tvarem jednotlivých vodičů vyjádřit jejich funkci, kterou router ale nezná). Nicméně jako nástroj pro zachování netlistu v průběhu operace Drag plní svou funkci dokonale. Využitelnost ostatních funkcí autorouteru pak podstatně závisí

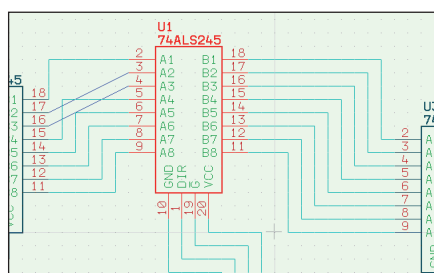


Obr. 4 Není-li k dispozici dostatek prostoru, zobrazí se namísto navržených vodičů pouze „pružné“ spoje

popsané výše. Co se stane, když příkaz k umístění bloku přijde předtím, než router dokončí zpracování, nebo když se mu některý vodič nepodaří v aktuálním rastru umístit? V prvním případě má router povolen ještě krátký čas navíc na dokončení. Ve druhém nezbývá, než namísto vodiče vložit do schématu jenom informaci, které dva body mají být propojeny.

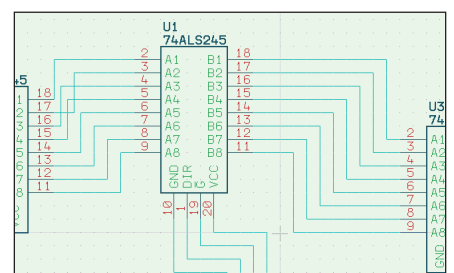
Proto byl zaveden nový typ prvku. Jde o obdobu „airline connections“, které se běžně užívají při návrhu desek. Ve schématu se označují jako tzv. „gumičky“ a pro odlišení od běžných vodičů se zakreslují jasně modrou barvou. Přesněji, „gumička“ je vodič, který nemá běžné pokročilé vlastnosti vodičů – jeho vnitřní body *nevytvoří* při dotyku s přípojným bodem vodivé propojení. „Gumička“ propojuje obecným diagonálním segmentem vždy právě jen koncové body. Neexistuje samostatný příkaz pro vložení „gumičky“. Ta se vkládá jedině automaticky právě v případě potřeby zachování netlistu po nekompletním routování.

„Gumiček“ se lze zbavit dvěma způsoby. První možností je další dragování (pů-



Obr. 5 Spoje, které autorouter nebyl schopen vytvarovat, se do schématu vloží jako „gumičky“

rastr – zpravidla bývá shodný s kurzorovým rastrem, který určuje nejmenší vzdálenost souběžných vodičů při ručním kreslení. Pokud se nedaří vodič umístit a router opakovaně vloží jen „gumičku“, je většinou nejrychlejším řešením zjemnit rastr, protože případné přílišné zahuštění vodičů představuje ve schématu pouze prohřešek proti přehlednosti. Další možností je změna pořadí routování. I když před samotným rozmístováním router vodiče setřídí, někdy může návrhář výsledek zlepšit tak, že routovací operace na vodiče aplikuje po skupinách.



Obr. 6 V jemnějším rastru si autorouter napodruhé s „gumičkami“ poradí

na složitosti a hustotě editované oblasti i na tom, jaký důraz je kladen na splnění (většinou nepsaných) estetických pravidel.

Více informací i samotný program pro vlastní pokusy najdete na www.formica.cz.

Literatura:

- [1] Horský, P. – Křivka, I.: *Formica 4.40 – zamykání netlistu, DPS od A do Z*, č. 2, 2012
- [2] Mikami, K. – Tabuchi, K.: *A computer program for optimal routing of printed circuit connectors, Proceedings of IFIP, H47: 1475–1478, 1968.*