



## Nyomatottáramkör-tervezés Linuxszal

Linuxot használva már sikerült megírunk egy mikrovezérlő programot. Ezek után most az áramköri kártyát is magunk tervezzük meg.

**A** Linux a legnagyobb hátrányban a tervezőprogramok terén van – ez az általános vélekedés. Így joggal tarthatunk attól, hogy a nyomtatott áramkör tervezéséhez szükséges CAD programok esetén is nehézségeink lesznek. Szerencsére ebben is megfigyelhető némi javuló tendencia, két irányból is: egyrészt bizonyos programozók már elkezdtek írni az igényeiket kielégítő tervezőprogramokat (belefáradva abba, hogy nincsenek ilyenek); másrészt néhány cég alapvető céljával tűzte ki, hogy a programja a lehető legtöbb rendszeren, köztük Linuxon is fusson.

Először röviden egy, ebbe az utóbbi kategóriába tartozó programot említenék: az EAGLE nyomtatottáramkör-tervezőt. Talán nem véletlen, de ezt a programot egy német cég fejlesztte. Németországban ugyanis ma már mind a kormányzat, mind más intézmények gépein is Linux fut (például a banki rendszereken). Hátha a hazai Linux (UHU) megjelenése is ebbe az irányba mozdítja a magyar felhasználást.

A program ismertetőjét a <http://www.cadsoft.de> oldalon találhatjuk (angol és német nyelven). A *freeware* hivatkozásra kattintva eljuthatunk az ingyenesen használható, ám ennek ellenére nem bemutatónálkülső tartalmú oldalához. Nem bemutatópéldány, mert mindent tud, amit a teljes változat, de az alábbi korlátozásokkal:

- A kártya mérete legfeljebb 100×80 mm lehet.
- Legfeljebb kétrétegű kártyákat lehet vele tervezni.
- A kapcsolásrajz-szerkesztő csak egyoldalas rajzokat kezel.
- Csak az online fórumon lehet hozzá (esetleg) támogatást kapni, telefonon és elektronikus levélben nem.
- Csak nonprofit felhasználást engedélyeznek regisztráció nélkül (kedvtelésből és oktatási célra megfelelő).

Letöltés után egy Linuxon futtatható programot találunk, amit végrehajtva megtörténik a telepítés. Nem rendszergazdaként is lehetséges telepítenünk az EAGLE-t, ekkor a felhasználó saját könyvtárba települ.

Amiért különösen érdemes kipróbálni ezt a programot (azon kívül, hogy létezik teljes értékű Linux-változata), az az, hogy a teljes értékű változatok ára sem elérhetetlenül magas (mint néhány CAD program esetében azt megszokhattuk). Ha például az ingyenes verzióval megegyező Light változatot szeretnénk regisztrálni, az 49 dollárba, a Standard változat 199 dollárba, a profi pedig 399 dollárba fog kerülni.

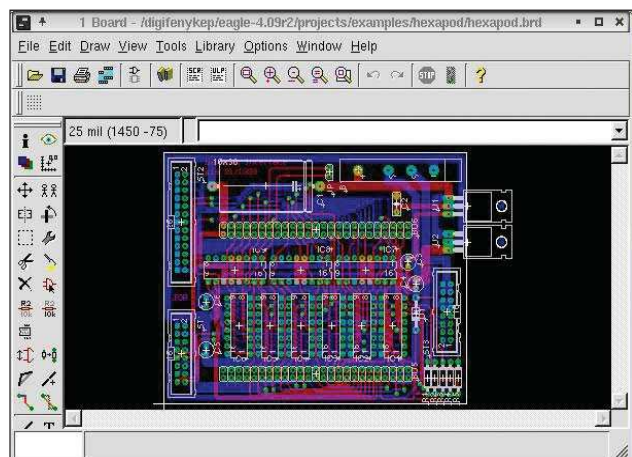
Milyen előnyökkel jár egy ilyen „gyári” CAD program használata a később ismertetendő, ingyenes, nyílt forrású programmal szemben?

- Beépített kapcsolásrajz-szerkesztőt tartalmaz, amiből közvetlenül előállítja az összekapcsolt pontok listáját (netlist).
- Az autorouter lehetőség is megtalálható benne, azaz el tudja készíteni (esetleg több-kevesebb hiánnyal) a nyomtatott áramkör tervét.
- Kimeneti formátuma olyan, hogy a legtöbb nyomtatott-áramkör-gyártó fel tudja használni a gyártáshoz, így ha

nem magunk akarjuk elkészíteni a kártyát, akkor is minden készen áll a megrendeléshez.

- Regisztráció esetén támogatást kaphatunk a fejlesztőtől (esetleg a könyvtárak esetében is kiegészítéshez juthatunk: manapság naponta jelennek meg új áramkörök. Ez utóbbit – mivel még nem regisztráltam – nem próbáltam ki.).

Hátránya nyilvánvalóan az, hogy gyorsan ki lehet nőni a „kedvtelés” kategóriát, és akkor fizetni kell érte, ha nem is sokat. Mint régi szabadprogram-rajongót, engem is – akárcsak valószínűleg az olvasók többségét – az érdekelt, hogy fellelhető-e nyílt forrású megoldás az adott feladatra. Nos, én a legérettebbnek, a gyakorlatban leginkább használhatónak a *Thomas Nau* által fejlesztett, nemes egyszerűséggel csak PCB-nek



nevezett programot találtam. Ő ugyan néhány évvel ezelőtt abbahagyta a fejlesztését, azóta *Harry Eaton* kezelte a programot – bár az ő honlapján se lehet túl sűrűn változásokat tapasztalni. Ezzel együtt a program jól használható, és mivel a forráskód nyílt, bárki nekifoghat a továbbfejlesztésnek. Ha esetleg nincs rajta Linux-terjesztésünk CD-in (számoson szerepel), akkor a <http://bach.ece.jhu.edu/~haceaton/pcb> oldalról elindulva letölthetjük. Jelenleg az 1.6.3-as változat tűnt az utolsónak. Telepítése a szokásos módon zajlik (`./configure ; make ; make install`).

Tekintsük át, melyek a **főbb előnyei**:

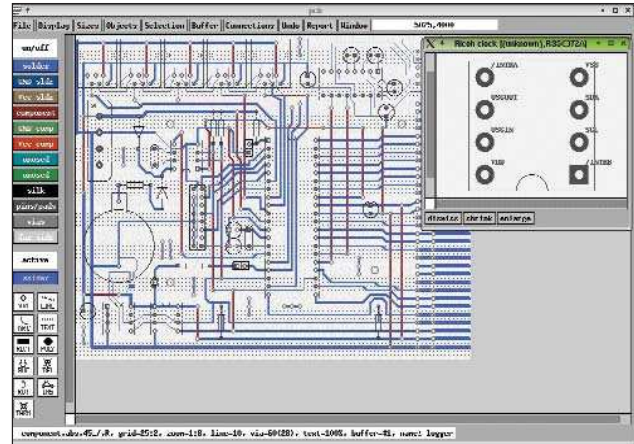
- Nyílt forrású, ingyenes.
- Nyolc réteggel működik.
- Jelentős méretű és bővíthető elemkönyvtárral bír.
- A Linux (Unix) alatt megszokott PostScript formátumú kimenete van, így könnyen nyomtatható, illetve átalakítható más formátumba (például pdf).
- Gerber-kimenet is van rajta, amit egyes kártyagyártók fel tudnak dolgozni.
- A leírása HTML formátumú.

**Hátrányai:**

- Gyakorlatilag nincs hozzá támogatás.
- A kártyagyártók nem biztos, hogy örülnek annak, ha csak Gerber- vagy PostScript-állományt tudunk adni.
- Nincs autorouter, azaz minden vonalat nekünk kell elhelyeznünk.
- Nincs beépített kapcsolásirajz-szerkesztő, tehát a netlist előállítás a többletmunkát igényel; vagy ha nem is készítünk kapcsolási rajzot, később dokumentációs gondjaink lehetnek (elfelejtjük, hogy melyik kapcsolat miért van).

Azért, hogy ezzel a programmal megkönnyítsem a munka megkezdését, néhány (általam alkalmazott) módszert, műveletet ismertetnék. A program a `pcb` paranccsal indítható, ekkor egy üres tervvel indul el. Lássuk a kidolgozás néhány lépését!

- Először be szoktam állítani a kártya méretét. Ez azért célszerű, mert így mindig látjuk a „határainkat”. Később, ha nagyon kell, lehet módosítani (általában növelni kell). Ezt a programablak felső részén látható menügombok közül a *sizes*-ra kattintva tehetjük meg. Ennek *PCB width*, illetve *PCB height* mezőjében állítható be a méret mil-ben, azaz egyezred collban (ez elég gyakran alkalmazott mértékegység az elektronikai alkatrészeknél). Ugyanitt lehet beállítani az aktív vonal, valamint az átforrasztás méreteit is, illetve a szöveg nagyságát.
- Maradjunk még egy kicsit a *sizes* menüpontnál! Itt négyféle huzalozási stílus közül lehet választani, illetve a hozzájuk tartozó vonal-, forrasz-szem és furatméreteket beállítani. Kényelmes, ha ezeket (signal, power, fat, skinny) az általunk kedvelt értékekre előre beállítjuk, és akkor később, munka közben csak ki kell választani a megfelelőt. Egyébként az előző pontban említett helyen megváltoztathatjuk az aktív méreteket, anélkül, hogy ezeket a beállításokat módosítanunk kellene. Figyelem, a kártyára már felhelyezett elemek nem módosulnak attól, hogy itt változtattunk! A tervben mindig az éppen beállított méretek jelennek meg, és nem a stílusok.
- A *display* menüpontban lehet megadni a kívánt rácsosztást, és hogy mutassa-e a rácsot (mint a pontokat) a terven. Ha lehetséges, a legkisebbalkatrész-lábosztásnál kisebbre szoktuk választani, hogy elég finoman lehessen vezetni a vonalakat. Felületforrasztott alkatrészek esetén esetleg megegyezhet a legkisebb lábosztással (ott úgysem fér el vonal a lábak között). Ugyanitt választható ki a nagyítás mértéke. Figyelem, miután kiválasztottuk, még kattintanunk kell a terven ahhoz, hogy alkalmazásra kerüljön! Azon a ponton kattintsunk, amelyiket szeretnénk, hogy a helyén maradjon. Ehhez képest lesz nagyobb, illetve kisebb a kártya többi része.
- A *file* pontban lehet menteni (vagy ha már létezik, betölteni) a terveket, itt tudunk nyomtatni is. Tudnivaló, hogy itt nem történik a megszokott, „valódi” nyomtatás, hanem csak fájlba írás. Ki lehet választani, hogy PostScript vagy Gerber formátumban írja-e ki a terveket, és hogy mi legyen rajta (körvonal, igazító jelek, fúrású segédjelek), valamint hogy milyen legyen a kimenet: színes, tükrözött, forogtatt, invertált stb. Ezután egy másik programmal (PostScript esetén például a `gv`) megnézhetjük az eredményt. Ha magunk akarjuk elkészíteni a kártyát, akkor a `gv`-ből is kinyomtathatjuk a megfelelő rétege(ke)t. (Fotóeljárással készítenő áramkörnél célszerű a végleges változatot feketeben pauszpapírra, lézernyomtatóval nyomtatni, ez használható maszknak a fotózásnál).



- A bal szélén lévő színes gombokkal választhatjuk ki, hogy melyik réteget akarjuk látni, az alatta lévő felbukkanó menüvel (az *active* feliratú címke alatti gombra kattintva jelenik meg) pedig azt, hogy melyikre akarunk elemeket elhelyezni. Leggyakrabban a *solder* és a *component* réteget alkalmazzuk.
- Bal oldalt lent helyezkednek el az eszköz kiválasztó gombok (átforrasztás, vonal, ív, szöveg, négyszög stb.), általában a bal egérgombbal tehetjük le őket a kártyára. Vonal esetén, ha nem az előzőt akarjuk folytatni, hanem újat szeretnénk kezdeni, a CTRL gomb lenyomása közben az új vonal első pontján kattintsunk.
- A jobb gombbal kattintva jelölhetünk ki egy elemet. A jobb gombbal négyszöget húzva kijelölhetjük a benne foglalt elemeket. Ha üres helyen csináljuk, ezzel törölhetjük az addigi kijelöléseket.
- A középső gombbal (kétgombos esetben mindkettővel egyszerre) mozgathatjuk az elemeket.
- Ha egész kijelölést akarunk mozgatni, akkor a *buffer* menüből válasszuk ki a *cut selection to buffer* pontot, majd egy kattintással a referenciapontot, az előbbi ezután az utóbinál fogva tetszőleges helyre mozgathatjuk, és a bal gombbal lerakhatjuk. Ha a *copy selection to buffer* pontot választjuk, ugyanígy egész részleteket is másolhatunk.
- Az adott eszközt vagy kijelölést, esetleg elemet az Esc billentyű lenyomásával dobhatjuk el.
- A beépített alkatrészkönyvtár a *window* menü *library* pontjával érhető el.
- Az *objects* menüben szerkeszthetjük a rétegeket, valamint itt lehet megváltoztatni az elemek nevét, és lekérni a lábkiosztást (ez egy külön ablakban jelenik meg).
- A *selection* menüből indulva módosíthatjuk a kiválasztott elemek méreteit (például utóbb jövünk rá, hogy mégis jobb lenne vastagabb vonal), és néhány egyéb tulajdonságát.
- A CTRL-M gombbal jelet tehetünk le. A jobb felső sarokban az ettől a jeltől való távolságot is mutatni fogja, nem csak a kártya bal felső sarkától mértet.

Nos, indulásnak talán ennyi elég is. Még egy dolgot emelnék ki, ami jóllehet benne van a leírásban, nekem némi fejtörést okozott – ez pedig a saját elem létrehozása. Nálam főleg csatlakozók esetében fordult elő, mivel annyi fajta csatlakozó létezik, hogy képtelenség mindet felvenni a könyvtárba, viszont célszerű elemként kezelni őket. Egyrészt, hogy lássuk a körvonalikat, és hogy milyen közel lehet tenni hozzájuk más elemeket, másrészt pedig, hogyha több is van belőle, akkor elemként



nagyon gyorsan felpakolhatjuk őket a tervre. Hogyan járunk el? A következő lépések útján hozhatjuk létre az elemet:

- Rakjuk fel a megfelelő átforrasztásokat (a lábak helyét).
- Egy vékony vonalat választva rajzoljuk meg a körvonalat.
- Jelöljük ki az így kialakult alkatrészt (a jobb gombbal húzva).
- Válasszuk a *buffer* menüből a *cut selection to buffer* pontot.
- Kattintsunk a kívánt referenciapontra (IC-knél egyes láb szokott lenni, másutt igény szerint).
- Válasszuk a *buffer* menüből a *convert buffer to element* pontot.
- Ezután a frissen létrejött elemet helyezzzük el a kártyán.

Ha nagyon alaposak akarunk lenni (és hosszabb idő múlva is meg szeretnénk érteni a saját tervünket), akkor – az állományt mentve – a fájlba a lábak neveit is beleírhatjuk. Íme egy példa:

```
Element(0x00000000 "Ricoh clock" ""
↳ "RS5C372A" 1380 1600 1 100 0x00000000)
(
  Pin(1550 1650 60 28 "/INTRB" "1" 0x00000101)
  Pin(1550 1550 60 28 "SCL" "2" 0x00000001)
  Pin(1550 1450 60 28 "SDA" "3" 0x00000001)
  Pin(1550 1350 60 28 "VSS" "4" 0x00000001)
  Pin(1250 1350 60 28 "/INTRA" "5" 0x00000001)
  Pin(1250 1450 60 28 "OSCOUT" "6" 0x00000001)
  Pin(1250 1550 60 28 "OSCIN" "7" 0x00000001)
  Pin(1250 1650 60 28 "VDD" "8" 0x00000001)
  ElementLine (1600 1700 1600 1300 10)
  ElementLine (1600 1300 1200 1300 10)
  ElementLine (1200 1300 1200 1700 10)
```

```
ElementLine (1600 1700 1450 1700 10)
ElementLine (1350 1700 1200 1700 10)
ElementArc (1400 1700 50 50 180 180 10)
Mark (1550 1650)
```

)  
A fájlban az elemet (ha már több van) a helyzete alapján találhatjuk meg, ugyanis az elsősorban a szöveges jellemzők utáni két szám az elem helyzete mil-ben.

Miket írhatunk itt be? Az `element` paraméterlistájában a három idézőjeles szövegbe lehet beírni az alkatrész nevét, csoportját és típusszámát. Lejebb a pin- (láb) leírásoknál a két szöveges részbe írhatjuk be a láb nevét és számát. A számát egyébként – ha ide nem írunk semmit – a bevitel sorrendjéből állapítja meg. Ezek után visszatöltjük a tervet, és az *objects* menüből kiválasztjuk a *display pinout* pontot, majd az elemre kattintunk, és egy külön kis ablakban máris láthatjuk a lábkiosztást. Sajnos annak a kérdésnek a tárgyalása, hogy miként készítsük el a megtervezett kártyát, meghaladja e cikk kereteit. Mindenesetre már így is sokkal közelebb kerülünk ahhoz, hogy működő mikrovezérlős rendszerünk legyen. Ha pedig az EAGLE programot választjuk, akkor akár profi termelőmunkát is végezhetünk kedvenc Linux-rendszerünkben.



Havránek Ferenc

Automatikamérnöként dolgozik. Kedvtelése közé tartozik mindenféle kétkerekű járművön (kerékpár és motor) való közlekedés. Ezenkívül szívesen tölti idejét programozással, nemcsak PC-s, hanem egyéb környezetben is, például mikrovezérlő programokat ír.

