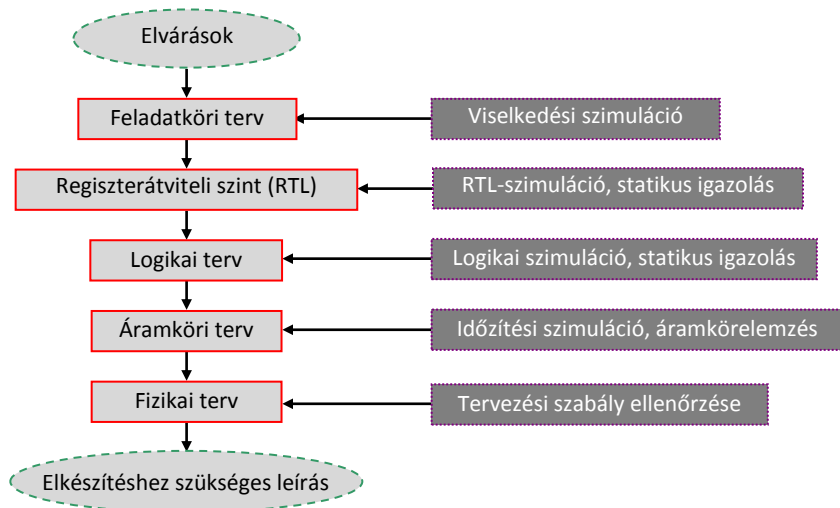


# 1. A VHDL mint rendszertervező eszköz

## 1.1. A gépi tervezés

A gépi leíró nyelvek (HDL) célja az egyes termékek egységesítése, logikai szimulációhoz leíró nyelv biztosítása, a terv hierarchikus felépítésének tükrözése és a nagy tervek áttekinthetővé tétele. Többféle nyelv (VHDL, Verilog, SystemC) létezik, de a VHDL vált a legelterjedtebb világszabvánnyá. A HDL nyelvek lehetővé teszik, hogy a mérnökök a terv felépítését és feladatköri jellemzőit a hagyományos kapuszinthez magasabb elvonatkoztatású megjelenési szinten írják le.

A következő ábra egy szokásos tervezési folyamatot mutat be a kezdeti elvárásoktól a gyártónak szóló leírásig. Ez utóbbi a programozható kapumátrixok esetén azt a programot jelenti, amellyel fel kell programozni az eszközt, hogy az a kívánt áramkörként működjön.



1.1. ábra. Digitális rendszer szokásos tervezési folyamata

## 1.2. A VHDL általános jellemzése

A VHDL a *VHSIC Hardware Description Language* (VHSIC hardverleíró nyelv), a VHSIC a *Very High Speed Integrated Circuits* (nagyon nagy sebességű integrált áramkörök) rövidítése. A nyelv a 80-as évek elejéről származik, jelenleg a legfontosabb szabványos nyelvek egyike az *előírás* (specification), az *igazolás* (verification) és az elektronikai tervezés területén. Ez annak köszönhető, hogy a nyelv alapeírása teljesen eszközfüggetlen, ezért a megfelelő szintéziseszközök széles skálája készülhetett el hozzá.

Az amerikai villamosmérnökök szervezete, az IEEE (IE<sup>3</sup> - Institute of Electrical and Electronics Engineers) 1987-ben szabványosította, a szabvány neve: IEEE 1076-1987, és 1993-ban létrehozták a VHDL-szabvány továbbfejlesztett változatát. Az a tény, hogy 1987 óta nemzetközi szabvány, így *bivatalosan támogatott*, nagymértékben hozzájárult az egyre szélesebb körű felhasználásához. Emiatt a VHDL-re jellemző a *tervek átcserélhetősége*, a VHDL-modellek garantáltan futnak bármelyik rendszeren, ezzel támogatják a csoportmunkát.

A VHDL olyan általános hardvermodellező nyelv, amely a digitális áramkörök különböző elvonatkoztatási szinteken történő, egységes leírására alkalmas. Felhasználható a legkülönbözőbb elektronikai rendszerek tervezésére, különböző céltechnológiákkal, például tervezhetünk vele szilíciumalapú integrált áramkört, hibrid IC-t, nyomtatott áramkört vagy éppen programozható kapumátrixot is. Ilyen programozható kapumátrix-áramkör a Xilinx FPGA vagy az Altera EPLD. Ez utóbbiak kész integrált áramkörök, amelyek felprogramozhatók úgy, hogy különböző felhasználók által tervezett áramkörként működjenek. Így segítségükkel egészen kis darabszám esetén is lehetőség nyílik integrált áramkör alkalmazására a digitális berendezésekben.

A VHDL nyelv kiemelkedő tulajdonságai, hogy nyelvi szerkezete átgondolt és technológiafüggetlen. Az utóbbiból adódik, hogy nem kötődik egyetlen szimulátorhoz vagy adatbázishoz sem, és nem erőltet rá a tervezőre egyetlen tervezési módszertant sem. A VHDL lehetővé teszi továbbá az új technológiák beépítését a már létező tervekbe. A *technológiafüggetlenségéből* adódik, azaz abból, hogy független a berendezési és IC-gyártási eljárástól, az, hogy el lehet választani egy szintet, ahol még technológiafüggetlen a terv, de az egyes logikai kapukhoz ezután a különböző technológiáktól függően különböző könyvtárakat lehet rendelni (CMOS, NMOS,

GaAs, FPGA, diszkrét elemek stb.). Emiatt technológiaváltás után is lehet használni a terveket.

Használatának előnyei közé tartozik az elektronikai terv fejlesztési idejének a le rövidítése és a terv későbbi módosításának az egyszerűsítése. A VHDL-t eredetileg előírási és modellezőnyelvnek szánták. Az első szimulátorokat a 80-as évek végén fejlesztették ki. Mivel a VHDL nyelv szabványos, a különböző VHDL-alapú tervezési eszközök bemeneti és kimeneti felületei megfeleltethetők egymásnak. Sajnos a VHDL-t nem szabványosították áramkörösztézésre, ugyanakkor a jelenlegi számítógépi teljesítmény lehetővé teszi az áramkörösztézés is.

A VHDL elképzelése az, hogy úgynevezett *egységeket* (entity) határoz meg, amelyeket külső és belső oldalról közelít meg. Ezek az egységek (összetevők, áramkörök, rendszerek) megoszlanak egy külső, látható (egységnev és kapcsolódások) és egy belső vagy rejtett rész (egységalgorithmus és megvalósítás) között. Az egységek egymást ezen a külső határfelületen keresztül használhatják.

A VHDL tervezési és modellezési nyelv. Segítségével a számítógép és az ember számára is olvasható alakban le lehet írni a digitális gépi rendszerek, áramköri kártyák és az összetevők, alkatrészek *szerkezeti felépítését* és *feladatkörét*. Egy tervezési tárgy VHDL nyelvű leírását *VHDL-modellnek* hívjuk.

A VHDL azért jobb a többi gépi leíró nyelvnél, mert az ezekkel szembeni különböző elvárások *mindegyikét* kielégíti, például támogatja a különböző tervezési módszereket és tervezési technikákat. Ennek keretében eltérő tervezési módszereket is támogat, például a fentről-lefelé, illetve az alulról-felfelé (könyvtárra támaszkodó) módszert. Hasonlóan különböző tervezési technikák is alkalmazhatók, mint a szinkron és az aszinkron, illetve a PLA- és a random logika.

A VHDL *egyidejű* (concurrent) programozási nyelv, ezért kifejezetten alkalmas igen nagy bonyolultságú digitális rendszerek viselkedésének a leírására.

### 1.3. Tervezési eljárás VHDL-lel

Egy berendezést különböző szinteken lehet leírni, attól függően, hogy az egész rendszer működését akarjuk ábrázolni vagy annak csak egy részegységét. Ezen kívül az egyes leírási módok különböznek attól függően, hogy az egység működésmódját írjuk le, vagy pedig a szerkezeti felépítésére, röviden a *szerkezetére* (structure) vagyunk

kíváncsiak. A VHDL mindkét fajta leírást lehetővé teszi. Segítségével egy tetszőleges digitális áramkört különböző elvonatkoztatási szinteken modellezhetünk. Ezt tükrözi, hogy a VHDL-ben megfogalmazott feladatok három különböző fő elvonatkoztatási szinten írhatók le:

- *Viselkedési* (behavioral) leírás: a terv feladatköri vagy algoritmikus megadása sorozatos VHDL-folyamatban van leírva.
- *Adatáramlási* (dataflow) leírás: az adatokat a terv bemenetétől a kimenetéig, mint egy folyamat elemeit írjuk le, az egyes lépések egyszerű adatátalakítások.
- *Szerkezeti* (structural) leírás: a modellben *összetevők* (components) egymáshoz való kapcsolódása van megadva.

Jellemző rá a *leíróképesség széles tartománya*, támogatja a digitális rendszer tisztán viselkedési, azaz feladatköri leírását, de támogatja a tisztán szerkezeti modellezését és a fő logikai egységeivel, valamint az azok közötti kapcsolattal való leírását is. Így például elképzelhető, hogy egy nagyobb rendszert szerkezeti leírással kisebbekre bontanak, amelyeket már viselkedési leírással lehet modellezni.

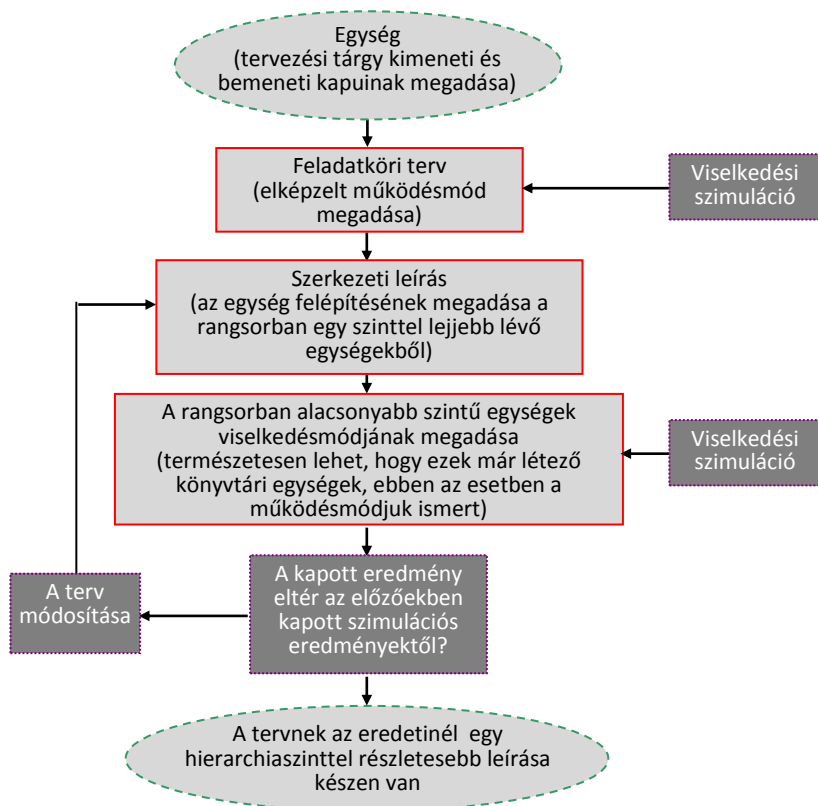
A VHDL-alapú tervezés általában a felülről-lefelé és az alulról-felfelé módszer együttes felhasználásával történik. Ennek során, miután megtörtént az egész áramkör általános működésének a meghatározása, az áramkört funkcionális részegységekre bontjuk, majd ezen egységek működésének viselkedési leírása és határfelületeiken keresztüli összekapcsolódásának a meghatározása következik. Természetesen az egész áramkör ki- és bemeneteit is megadjuk.

Amikor egy áramkörü modellt meghatároztunk, szimulálhatjuk a viselkedési leírásának (vagy az összetevői viselkedési leírásának) a végrehajtásával. Ennek során az adott időtartamot diszkrét lépésekben vizsgáljuk. Egy adott szimulációs időpontban az áramkörü egység bemenetén egy bemeneti kapocs értékének változását látjuk, ha az új érték különbözik az előzőtől.

Ez a tervezési lépés a leírásokon kívül magába foglalja a részegységek, illetve a teljes áramkör szimulációját, működési helyességének az ellenőrzését. A következő lépés (alulról-felfelé) a részegységek szerkezeti szintű leírása szabványelemek felhasználásával. A részegységek szimulációja után azok összeköttetésének megadása következik, majd az egész rendszer szimulációja. A végső ellenőrzés a kétféle leírás szimulációs eredményeinek az összehasonlítása.

A részegységek modellezésénél optimalizálás is végezhető azzal, hogy a problémás részegységekre több eltérő megvalósítást adunk. Ezek a megvalósítások aztán a vizsgálálandó paraméterek alapján összehasonlíthatók.

A *szintézis* (synthesis) mindig két elvonatkoztatási tervezési szint között történik, amelynek során egy magasabb elvonatkoztatási szintről egy alacsonyabbra jutunk úgy, hogy közben többletinformációkat építünk be a modellbe. A tervezés menete fentről lefelé haladva az egyszerűbb tervekől a részletesebbek felé tartva szintézis-lépésekből áll. Egy ilyen lépést mutat be az 1.2. ábra.



1.2. ábra. Egy szintézislépés a VHDL-alapú tervezés folyamatában