

Das virtuelle Gerät - der DigiTrace- Compiler

Der Compiler erweitert den bewährten DigiTrace-Logiksimulator zum Entwicklungssystem für verschiedenste Steuerungsaufgaben. Er compiliert die entwickelte Schaltung zum ausführbaren Runtime-File inklusive bedienbarer Frontplatte, Soundeffekten usw. Das Installieren des DigiTrace-Systems ist auf dem ausführenden Rechner nicht notwendig. Wir stellen Ihnen den Compiler ausführlich vor.

DigiTrace abgekoppelt

Das Logiksimulatorsystem DigiTrace, das wir Ihnen bereits im „ELVjournal“ 2/96 ausführlich vorgestellt haben und das nun inzwischen optisch verbessert und funktionell erweitert als Version 2.0 vorliegt, ermöglicht ja bekanntermaßen nicht nur die Erarbeitung einer Schaltungssimulation am Bildschirm. Vielmehr ist eine einfache Verbindung zu externer Hardware über eine Reihe von Schnittstellen wie LPT-Ports, PC-interne I/O-Karten (z. B. ELV PIO 32-Karte) oder externe Steuer- und Datenerfassungsgeräte wie Schalteinheiten, Logikanalysatoren etc. realisierbar, was dem Programm einen weiten Einsatzbereich zur Erarbeitung selbst von komfortableren Anwendungslösungen eröffnet.

Um das Programm jedoch derart einsetzen zu können, mußte es bisher auf jedem Anwendungs-Rechner installiert werden - nicht nur ein finanzielles Problem.

Denn durch das Vorhandensein des Vollprogramms sind dem „Interessierten“ natürlich Tür und Tor geöffnet, „einmal nachzusehen, was dahinter steckt“. So könnten Manipulationen an der Schaltung etwa zu Problemen im praktischen Einsatz führen. Außerdem möchte wohl auch nicht jeder Schaltungsentwickler sich vom Mitbewerber in die Karten schauen lassen und seine Schaltungslösung irgendwo kopiert wiederfinden - gerade im gewerblichen Bereich ein wichtiger Aspekt.

Also liegt es nahe, die komplette Funktion der Schaltung einschließlich der Frontplatte, über die die Bedienung erfolgt, so vom Entwicklungssystem abzukoppeln, daß Außenstehenden zwar die Nutzung,

aber nicht die Beeinflussung der dahinterstehenden Schaltungslösung, möglich ist.

Die Lösung für dieses Problem heißt DigiTrace-Compiler.

Der DigiTrace-Compiler

Der DigiTrace-Compiler erzeugt aus einer fertig entwickelten, getesteten und funktionsfähigen Schaltung ein ausführbares Exe-File, das sowohl die Funktionen der Schaltung als auch eine Runtime-Version von DigiTrace zur Ausführung auf einem beliebigen anderen (Windows-) Rechner erzeugt. Natürlich werden hier auch die Frontplatten-Bitmaps und eventuell eingesezte Sounddateien abgespeichert.

Wird diese Exe.Datei später gestartet, sieht der Nutzer nur die Frontplatte des entwickelten Gerätes vor sich, mittels derer Bedienelemente und Anzeigen kann er das virtuelle Gerät bedienen.

Somit braucht DigiTrace selbst nur noch auf dem Rechner des Entwicklers zu laufen.

Selbst umfangreichere Schaltungen mit Sounds und großen Frontplatten passen auf eine Diskette, und die Runtimes sind auch bequem transportabel (z. B. auch per Datennetze schnell verteilbar). So ist z. B. eine Fernwartung des virtuellen Gerätes durch Einspielen eines neuen Runtimes ebenso möglich wie die zeitnahe Online-Entwicklung nach spezifischen Kundenwünschen.

Der Compiler erfordert die Version 2.0 von DigiTrace, die von ELV bereits ab Ende April 1998 ausgeliefert wird.

Die Bedienung des Compilers

Die Bedienung des Compilers ist denkbar einfach.

Zunächst erstellen Sie Ihre Schaltung mit DigiTrace 2.0 wie gewohnt (Abbildung 1). Dabei ist natürlich darauf zu achten, daß sich alle Bedien- und Anzeigeelemente auch auf der fertigen Frontplatte befinden, die zudem so gestaltet sein sollte, daß sie später vom Nutzer als übersichtlich empfunden und bequem bedient werden kann.

Denken Sie bei der Konzeption auch daran, daß die Schaltung stets über einen Start/Stop-Schalter o. ä. verfügen sollte, da später beim Start des Runtime-Files auch die Schaltungsfunktion automatisch gestartet wird.

Nach dem Speichern des fertigen Schaltungsfiles (Projektdatei) erfolgt nun das Starten des Compilers (Abbildung 2) über das entsprechende Windows-Programmsymbol.

Nach dem Betätigen der Schaltfläche „Quelldatei festlegen“ wird die gewünschte Projektdatei angewählt (Dateiendung .dig), die nach Schließen der Auswahlbox in der

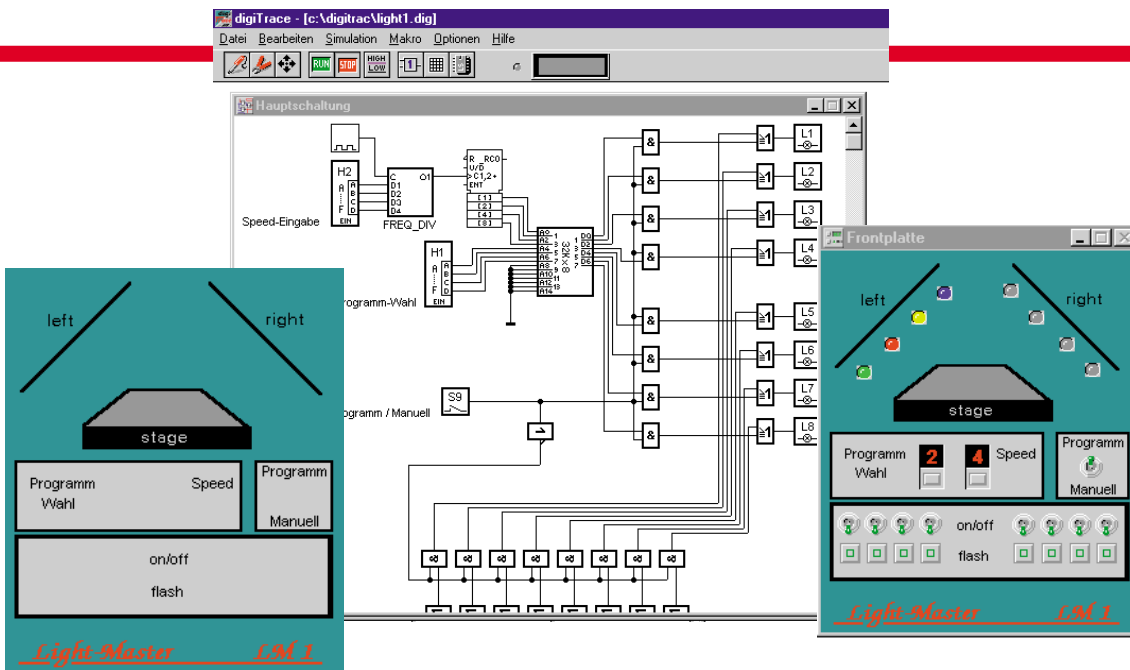


Bild 1: Erste Arbeitsgänge: Frontplatten-Bitmap zeichnen, Schaltung und Frontplatte entwerfen und testen.

Statuszeile für die Quelldatei erscheint.

Daraufhin ist die Zieldatei sowie deren Standort festzulegen. Dies sollte man zweckmäßigerweise in einem bereits vorher angelegten Ordner vornehmen, um alle zusammengehörigen Dateien auch zusammengefaßt ablegen zu können und später bei der Weitergabe nicht etwa eine der Dateien zu vergessen.

Bei der Compilierung werden dann alle

Nachdem man sich nochmals vergewissert hat, daß Quell- und Zieldatei richtig ausgewählt sind, startet man durch Betätigen des Buttons „Compilierung starten“ den Compilerlauf. Der Compiler arbeitet sehr schnell, lediglich bei Abspeicherung direkt auf Diskette kann man das Speichern der einzelnen Dateien in der untersten Statuszeile verfolgen.

Hier meldet der Compiler auch den erfolgreich verlaufenen Compilerlauf.

Nun kann man das Ergebnis der Compilierung bequem testen, indem man das EXE-File im Zielverzeichnis startet.

Auf dem Bildschirm erscheint nun sofort die Frontplatte des virtuellen Gerätes, das sich per Maus bedienen läßt (Abbildung 3).

Erscheinen statt dessen mehrere leere DOS-Fenster mit der Aufschrift „Zu wenig Speicher“, so kann es bei Installation des Compilers bzw. beim Compilerlauf, abhängig von der Hardware- und

Ab zum Nutzer!

Nach dem Probelauf des compilierten Files kann nun das komplette Zielverzeichnis des Projekts an den Nutzer weitergegeben und auf dessen Rechner gestartet werden. Eine Installation ist dort nicht erforderlich, lediglich die notwendige periphere Hardware muß natürlich angeschlossen und entsprechend konfiguriert sein, wie es vom Projektentwickler festgelegt wurde.

Sie sehen, so kann in kürzester Zeit aus einem Schaltungsprojekt ein anwendbares und teilweise nur virtuell vorhandenes Gerät werden - ein schneller und kostengünstiger Weg, der das kleine Schaltungsentwicklungssystem DigiTrace zum mächtigen Werkzeug für den Entwickler macht.

Mehr zu DigiTrace und dem DigiTrace-Compiler können Sie nachlesen in dem Buch „Logiksimulation mit dem PC“, das von den DigiTrace-Autoren verfaßt wurde und im Franzis-Verlag erschienen ist.

Die dem Buch beiliegende CD-ROM enthält eine Demoversion von DigiTrace mit einer Vielzahl von weiteren Beispielen, die gleich von der CD-ROM aufrufbar sind und sofort zur praktischen Übungen anregen. Auch finden Sie weitere Erläuterungen und Beispiele zum DigiTrace-Compiler. **ELV**



Bild 2: Der Compiler übersetzt das Projektfile in eine Runtime-Version.

zugehörigen Dateien wie z. B. Frontplatten-Bitmaps, Klangdateien etc. abgelegt.

Legen Sie keine Projekte mit dem gleichen Namen in einem Ordner ab, da die vorhandenen Dateien während des Compilerlaufs ohne Rückfrage überschrieben werden. Sinnvollerweise legt man also Projekt für Projekt in einem eigenen Verzeichnis bzw. Ordner ab.

Geben Sie dem Projekt in der Auswahl einen Namen. Anderenfalls wird es als „Noname.exe“ abgelegt.

Nach Schließen der Auswahlbox erscheint der Dateiname in der Statuszeile „Zieldatei“.

Betriebssystemkonfiguration des Computers dazu gekommen sein, daß im Compilerverzeichnis (z. B. c:\digicom) oder aber im Zielverzeichnis die Datei Bwcc.dll als Datei mit 0 kB Inhalt erscheint. In diesem Falle sollte entweder der Compiler nochmals installiert werden oder aber diese Datei aus dem DigiTrace-Verzeichnis manuell in das Compilerverzeichnis kopiert (nicht verschoben) werden. Sie muß sich neben den eigentlichen vom Compiler im Zielverzeichnis abgelegten Dateien ebenfalls in diesem befinden.

Eine Hilfedatei ist übrigens ebenfalls aus dem Compiler heraus aufrufbar.

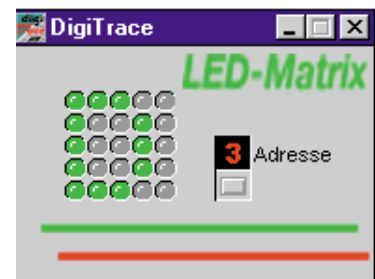


Bild 3: Als Exe-File gestartet, erscheint nur die Frontplatte mit Anzeigen und Bedienelementen auf dem Bildschirm.