

ProfiLab“ und das Entwicklungssystem für Digitaltechnik „Digital ProfiLab“. Als man schließlich vor etwa 5 Jahren beide Systeme zu einem zusammenfasste und mit einem integrierten Compiler für die Erzeugung von Runtime-Files kombinierte, war die erste Generation von ProfiLab Expert geboren. Mit der jetzt vorliegenden Version 3.0 hat sich das System zum wirklich kompletten Entwicklungssystem für analoge/digitale Schaltungen entwickelt, das äußerst umfangreich an inzwischen hunderte PC-Peripheriegeräte anbindbar und so zu einem komplexen Steuer-, Mess- und Regelsystem ausbaubar ist.

Denn seit unserer ersten Vorstellung des Systems vor gut 5 Jahren hat sich eine Menge getan, vor allem in Sachen Komfort, Peripherie, Ausgabemöglichkeiten und integrierten Laborkomponenten. Wir wollen die neue Version in einem kurzen Report kennen lernen.

Schaltung per Drag & Drop

Nach der Installation von CD-ROM wird man von einer aufgeräumten Arbeitsfläche empfangen. Links und oben befinden sich die so genannten Toolbars mit allen für die unmittelbare Programmbedienung wichtigen Werkzeugen wie Zeichen- und Platzierungswerkzeugen, Buttons für die Bedienung der Simulation und für das Handling der Zeichenfläche. Um es vorwegzunehmen – bereits nach den ersten Schritten fällt es äußerst angenehm auf, dass viele zusätzliche Werkzeuge direkt in den Dialogboxen oder z. B. im Frontplattenentwurf als Taskleiste oder Buttons verfügbar sind, man muss also kaum in die Pull-down-Menüs hinein.

Die neue Version bringt fertige Beispiele in einer großen Anzahl mit – der schnellste Weg, um die vielen Features des Programms kennen zu lernen. Hier kann man direkt anfangen zu experimentieren.

Die Bauteile werden einfach per Drag &

Das virtuelle Labor – Digital ProfiLab Expert 3.0

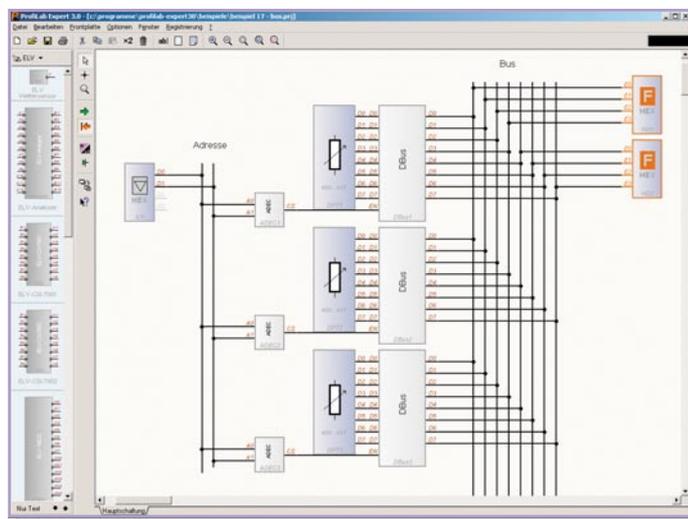
ABACOM hat die Konsequenzen gezogen – und Digital ProfiLab Expert nunmehr in der dritten Version herausgebracht. Sie vereinigt unter einer einheitlichen Bedienoberfläche eine komplette analoge/digitale Entwicklungsumgebung mit Echtzeit-Schaltungssimulation, erweiterter PC-Peripherie-Anbindung und einem integrierten Compiler für die Erstellung praktisch einsetzbarer Runtime-Files.

Schaltungsentwicklung am PC ...

... ist nichts Neues, diese Art der programmzeilenlosen Schaltungsentwicklung gab es spätestens seit Erscheinen des ersten „Volks-Schaltungssimulators“ Digi-Trace vor etwa 9 Jahren schon. Der war damals ein Renner, konnte man doch erstmals zu einem für jedermann erschwinglichen Preis Schaltungen komplex per Drag & Drop am Bildschirm entwerfen, testen und perfektionieren.

In der Konsequenz dieses Erfolges entstand die „Profilab-Reihe“ – das auf der Basis von Digital-Multimetern mit serieller Schnittstelle agierende Entwicklungssystem für Messtechnikprojekte „DMM-

Bild 1: Auch umfangreiche Schaltungen sind schnell erstellt



**Tabelle 1:
Unterstützte ELV-Hardware***

- Schaltinterface CSI 7001/7002
- ELV-USB-I/O-Interface UIO 88
- ELV-Messmodul M 232
- ELV-RS-232-I/O-Board IO 88
- PC-IO 16 I/O Modul
- PIO 32 Digitalkarte
- ADS 12 AD-Wandlermodul
- ADA8/16-2 Wandlerkarte
- ELV-Digital-Analyzer
- ELV-Wettersensor-Empfänger
- DMMs mit serieller bzw. USB-Schnittstelle:
DMM750 (=Wens700), DT4000ZC, DVM 345 DI, GDM 703, GDM 704, GDM 705, M 3610 D, M 3640 D, M 3650 D, M 3830, M 3850, M 3850 M, M 3860 M, M 3890 D USB, M 4650 CR, M 4660, MAS 345, ME-22, ME-32, ME-42, MS 9150, MS 9160, MXD 4660 A, Protek 506, VC 350, VC 608, VC 630, VC 635, VC 650, VC 655, VC 670, VC 675, VC 820, VC 840

* Weitere unterstützte Hardware anderer Hersteller finden Sie im Internet: www.software.elv.de

Drop aus den wieder einmal umfangreicher gewordenen (und selbst erweiterbaren) Bauteilbibliotheken auf der Arbeitsfläche platziert (Abbildung 1). Sofort fällt hier auf, dass insbesondere die Ausgabemöglichkeiten moderner und die Anschlussmöglichkeiten für Peripherie umfangreicher geworden sind. So sind jetzt LCD-ähnliche Textanzeigen, Mehrfarbleds ebenso enthalten wie die Möglichkeiten, etwa Wettersensor-Empfänger oder zahlreiche USB-I/O-Geräte wie z. B. unser UIO 88 anschließen zu können. Tabelle 1 enthält allein die Geräte aus dem ELV-Sortiment, daneben sind ungezählte von Kolter, Meilhaus, Velleman, AK Modulbus, Conrad, HygroTec, BMC, fischer-technik u. v. a. anschließbar. Abbildung 2 zeigt einige Peripheriegeräte aus dem ELV-Programm. Ach ja, die Liste der einbindbaren Digital-Multimeter ist auch wieder



Bild 2: Zahlreiche I/O-Baugruppen und Geräte sind mit ProfiLab Expert einsetzbar.

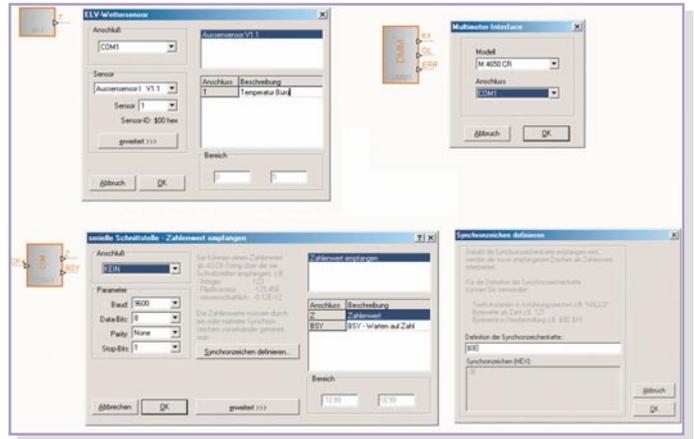


Bild 3: Programmierbare Bausteine – einfach Parameter eingeben – fertig!

deutlich länger geworden. Und mit jedem (per Internet verfügbaren) Update des Programms kommen auch wieder einige Peripheriegeräte hinzu. Und schließlich sind natürlich auch eigene Applikationen einzubinden, hier ist ja peripherieseitig USB-Technik stark im Kommen.

Das Herstellen der Verbindungen zwischen den Anschlüssen erfolgt denkbar einfach durch halbautomatische Zeichenhilfen. Eine Makrobibliothek, die selbstverständlich ebenfalls erweiterbar ist, gestattet das Einbinden eigener, komplexer Schaltungsteile als Sub-Schaltungs-Baustein in die Schaltung. Einfaches Beispiel dafür sind die in der Makrobibliothek mitgelieferten Logik-Bausteine. Die erscheinen wie ein IC-Gehäuse im Schaltplan – platzsparend und beim späteren, pingenaunen Nachbau leicht nachzuvollziehen.

So ist im Nu die gewünschte Schaltung entstanden, mit Peripherie „bestückt“ sowie Anzeigen und Messgeräte (vom schnellen 2-Kanal-Oszilloskop über 8-Kanal-Logik-Analysator bis zum Frequenzzähler) angebunden.

Besonders hervorzuheben sind aus Sicht der späteren Nutzung als reale Steuerung auch die neuen Tages- und Wochenschaltuhren, die zeitgesteuerte Abläufe wesentlich einfacher programmieren- und bedienbar machen. Auch das byteweise mögliche Schreiben und Auslesen von Dateien ist eine deutliche Komfortsteigerung.

Und schließlich soll auch noch die jetzt mögliche Zusammenschaltung von Bauteilanschlüssen zu Bussystemen per Wired-

Or genannt werden – ein Wunsch aus Praktikersicht, den wir seinerzeit beim Test der ersten Version geäußert hatten, ist erfüllt.

Und für das, was es noch nicht gibt, bietet ein DLL-Baustein eine Programmierschnittstelle für eigene Bauteile und Hardware-Treiber.

Alle Parameter der Schnittstellen (parallele, serielle, USB, Gameport, interne I/O-Karten) werden ganz bequem direkt aus der Schaltung heraus eingestellt, man muss keinen Extra-Treiber für die Geräte aktivieren.

Alle neuen Schaltungsteile des Systems hier zu diskutieren, würde an dieser Stelle den Rahmen sprengen, aber Tabelle 2 gibt einen kompletten Kurzüberblick über alle neuen Features der Version 3.0.

Das ganze Schaltungs-Entwurfssystem ist also eine Mischung aus traditionellem Schaltungsentwurf und der Einbindung programmierbarer Schaltungsteile (Abbildung 3). Programmieren heißt hier aber nicht, mühselig Programmzeilen zu schreiben, sondern es beschränkt sich lediglich auf die Parameterfestlegung etwa von Ein- und Ausgangsschnittstellen oder programmierbaren Bausteinen wie z. B. dem Textdisplay. Das zeigt aber auch, dass das System, wie wir noch sehen werden, vorrangig für die spätere tatsächliche Applikation auf PC-Anwendungen ausgerichtet ist. Einfache Schaltungen sind freilich direkt als Hardware nachzubauen, komplexere hingegen, und das beginnt schon bei der Auswertung der seriellen Schnittstellen, bedingen dann den Einsatz eines Mikrocontrollers, und hier ist wieder der normale Programmierer gefragt.

Das Frontplatten-Tool

Sobald ein Bedien- oder Anzeigeelement auf der Arbeitsfläche platziert ist, kann man sich dieses auf der virtuellen Frontplatte anzeigen lassen. Hier ist es beliebig platzierbar, ja, man kann sogar mehrere Frontplatten (manuell oder programmgesteuert umschaltbar) generieren, es ist eine recht freie Gestaltung fast aller Anzeige- und Bedienelemente via Kontextmenü möglich. Auch hier sind zahlreiche Neuigkeiten verfügbar (siehe Tabelle 2). An hervorragender Stelle sollen hier nur

die realistischere Darstellung mit verbesserter Skalierbarkeit sowie das Textdisplay zur Anzeige von ASCII-Daten genannt werden. So ist man in der Lage, ein oder mehrere komplette Frontends für den späteren Anwender zu generieren. Die Freiheiten beim Entwurf gestatten auch später jede Änderung, etwa, wenn ergonomischere Anordnungen vom Nutzer verlangt werden. Und das Ganze geht auch schneller als früher, weil z. B. Anordnungswerkzeuge wie etwa Ausrichten direkt im Frontplattfenster verfügbar sind und nicht erst in Pull-down-Menüs gesucht werden müssen.

Was hier möglich ist, werden wir noch an einem Beispiel sehen.

Simulieren statt löten

Hat man seine Schaltung auf diese Wei-

se bedienbar gemacht, kann der Simulationslauf gestartet werden. Dieser ist gegenüber früheren Versionen ebenfalls weit komfortabler. So ist z.B. eine pinbezogene Statusanzeige für alle Leitungen integriert. So lassen sich Fehler schneller finden.

Dass die Simulation in Echtzeit abläuft, ist inzwischen Standard. Unterstützt wird der Entwickler durch die mögliche Einbindung der zahlreichen Messgeräte, wie 2-Kanal-Oszilloskop, Logik-Analyzer, Frequenzzähler oder Y(t)-Schreiber (Abbildung 4). Auch ein Signalgenerator mit verschiedenen Signalformen fehlt hier übrigens nicht.

Ist die Schaltung komplett lauffähig, optimiert und vollständig, kann sie gespeichert oder mit einem komfortablen Drucktool (mit stufenloser Zoommöglichkeit der Ausgabe) ausgedruckt werden.

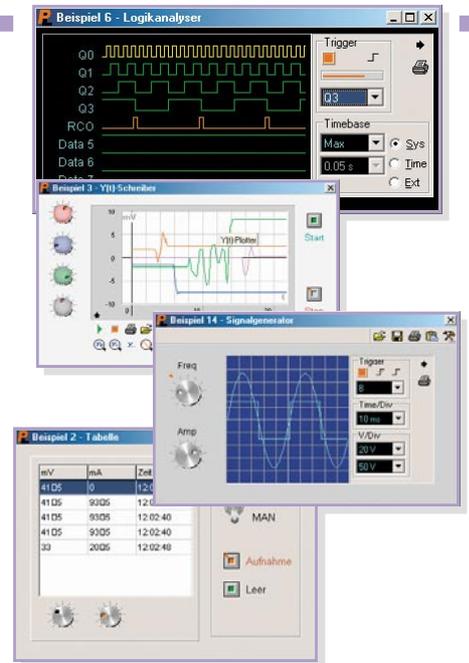


Bild 4: Zahlreiche integrierte Messgeräte helfen bei der Schaltungsentwicklung und können auch in das Frontend integriert werden.

Schaltung fertig? Kompilieren!

Soll die fertige Schaltung nicht als Grundlage eines eigenständigen Gerätes dienen, sondern tatsächlich als Applikation auf einem PC laufen, so ist nun der integrierte Compiler dran.

Er wandelt die komplette Schaltung in eine ausführbare Datei um. Diese wird einfach auf dem PC, auf dem sie laufen soll, gestartet und funktioniert hier, einschließlich aller Peripherie-Anbindung, völlig im Hintergrund. Zu sehen ist nur eins – die Frontplatte mit den Bedien- und Anzeigeelementen (Abbildung 5). Bedient wird das Ganze mit der Maus oder, falls man einen Touchscreen-Bildschirm einsetzt, direkt durch Berühren der entsprechenden Sensorflächen. Ein solcher Touchscreen-Bildschirm bietet sich überall da an, wo die Präsenz von Tastatur und Maus störend wäre. Und er hat den Vorteil, dass man lediglich das Programm selbst bedienen kann, nicht aber das Betriebssystem dahinter. Er simuliert via USB und einem kleinen Programm einfach die Mausclicks, die ja auch der Bedienung auf der virtuellen Frontplatte dienen.

Doch zurück zum Compiler. Das erzeugte, ausführbare Programm hat diverse Vorteile. Zum einen ist die eigentliche Schaltung nicht darin enthalten und zum anderen auch nicht das Entwicklungsprogramm. So kann das .exe-File frei weitergegeben werden. Will der Anwender eine Änderung – nichts einfacher als das! Einfach die Schaltung ändern, simulieren und kompilieren. Dann wird sie per E-Mail versandt, und der Anwender kann sein „neues Gerät“ im Nu installieren.

Tabelle 2: Die Neuheiten von DMM-ProfiLab 3.0 in der Übersicht

- schnellere, einstellbare Taktgeber (1 ms)
- invertierbare Ein- und Ausgangspins für alle Bauteile
- neue, pinbezogene Statusanzeige (High/Low)
- neue, noch realistischere Bedienungselemente mit besserer Skalierbarkeit
- mehrere Frontplatten in einem Projekt. Manuell und programmgesteuert umschaltbar
- bauteilsensitive Hilfefunktion, direkt abrufbar
- Tastatursteuerung zur Bedienung der Frontplatte
- benutzerdefinierbarer Bitmap-Schalter
- Potenziometer wahlweise mit logarithmischer Skala und rastender Mittelstellung
- neue Zeigerinstrumente, wahlweise auch mit logarithmischer Skala
- Bildprojektor-Bauteil erlaubt das programmgesteuerte Einblenden von Bildern und GIF-Animationen
- mehrfarbige Duo-LED
- skalierbare Leuchtbalkenanzeige
- 16fach-Auswahlschalter (Combobox)
- Eigenschaften der Frontplattenelemente, wie z. B. Farben, Texte, etc., optional zur Laufzeit einstellbar
- editierbare Hints (Hinweistexte) für Bedienungselemente auf der Frontplatte
- Einstellungen der Frontplattenelemente können automatisch oder manuell gespeichert und geladen werden
- Druckfunktion/Zwischenablage für die Frontplatte
- elementare Hardware-Einstellungen in Abhängigkeit zur Laufzeit einstellbar
- Bauteilaustritte zu Bussystemen zusammenschaltbar (wired-or)
- Ausrichten-Funktionen für die Frontplatte
- komfortable Tages- und Wochenschaltuhren
- Bauteil zur einfachen Realisierung von Schaltverzögerungen
- 2-Kanal-Oszillograph zur Darstellung von Vorgängen im Millisekundenbereich
- schnellerer und verbesserter Y(t)-Schreiber
- Analog(de)multiplexer mit bis zu 16 Kanälen
- Verstärker mit steuerbarem Gain und Offset
- Korrekturtabelle mit linearer Interpolation zur Anpassung nichtlinearer Sensoren etc.
- Signalgenerator mit Sinus, Dreieck, Rechteck, Rampe
- Limiter zur Begrenzung analoger Größen
- Sampler-Bauteil zur Aufzeichnung und Wiedergabe von Schaltvorgängen und Wertänderungen mit bis zu 16 Kanälen
- Textdisplay zur Anzeige von ASCII-Daten
- Bustreiber und Adressdecoder
- übersichtliche Bauteildarstellung von Makrobibliotheken
- Frequenzzähler-Bauteil
- vereinfachtes Kompilieren mit Statusanzeige
- DLL-Baustein bietet Programmierschnittstelle für eigene Bauteile oder Hardware-Treiber

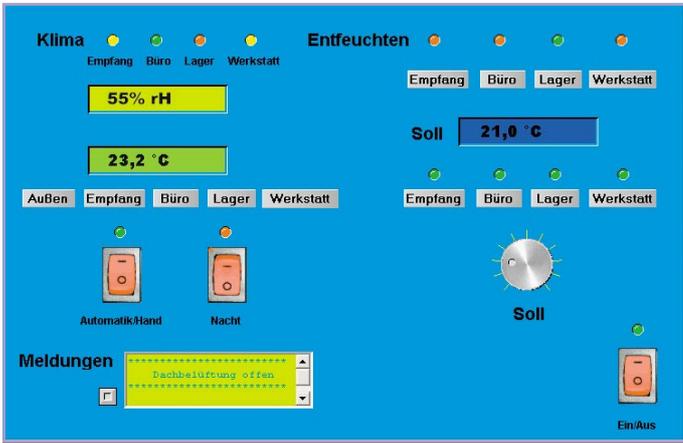


Bild 5: Ein fertiges Projekt in Funktion – hier sind zahlreiche Möglichkeiten des Frontplatten-Tools zu sehen.

Applikation ganz einfach

Unser Anwendungsbeispiel zeigt die Steuerung einer dezentral arbeitenden Klimaanlage einer kleinen Firma (Abbildung 6), die aus einem zentralen Außengerät und vier individuell steuerbaren Raum-Klimageräten besteht. Hier kann man exemplarisch sehr gut erkennen, wie sehr einfach handelsübliche Peripherie eingebunden werden kann.

Als E/A-Baustein dient hier die ELV-I/O-Baugruppe des Typs IO 88. Dieser Interface-Baustein verfügt über ein serielles Interface und je 8 Ein- und Ausgänge.

Die Temperaturerfassung erledigt unser Wettersensor-Empfänger, der die Daten der normalen Temperatursender unserer Wetterstationen „einsammelt“. Er liefert ja Hex-Daten innerhalb eines einfachen Protokolls. Diese können mit den Bausteinen des Entwicklungssystems auf mehrfache Weise ausgewertet werden. Zum einen ist die direkte Aufbereitung zu einer Werteanzeige, hier von Temperatur und Luftfeuchte, möglich, zum anderen sorgt ein Interface-Programmbaustein dafür, dass bei Erreichen bestimmter, programmierbarer Werte, Schalt- und Anzeigevorgänge ausgelöst werden.

So konnte man hier durch relativ einfache Verknüpfungen von Temperatur und Luftfeuchte die Mehrfarb-LEDs oben als Klimakomfort-Anzeigen ansteuern.

Da der Wettersensor-Empfänger die Daten aller Sensoren empfängt, ist er, auch wenn verschie-

dene Orte abgefragt werden sollen, nur einmal erforderlich. Lediglich im Schaltplan wird er virtuell so oft installiert, wie man ihn für verschiedene Verknüpfungen benötigt.

Die Steuerung und Programmierung der einzelnen Klima-Innengeräte erfolgt im rechten Teil der Frontplatte. Sie ist dem eingesetzten Klimagerät, das über eine Kabelfernsteuerung verfügt, angepasst.

Oben wird bei Bedarf die separat schaltbare Entfeuchtungsfunktion per Hand geschaltet, sie kann aber auch, über die beschriebene Klimaauswertung gesteuert, automatisch geschaltet werden.

Für den Kühl- bzw. Heizbetrieb ist ein Temperatursoll, für jeden Raum getrennt, einstellbar. Das wird mit einem recht großen Hysteresebereich kontrolliert, um die interne Klimaautomatik der Klimageräte nicht zu oft zum Schalten zu zwingen. Allerdings hat sich hier die externe reale Raumtemperaturmessung über die Temperatursensoren bewährt, da die Raum-Klimageräte durch die interne Temperaturmessung oft nur die Temperatur unmittelbar am Gerät halten, aber nicht für den ganzen Raum.

Die LED-Anzeigen über den Tasten zeigen an, ob das jeweilige Klimagerät aktiv ist.

Eine Nachtschaltung für energiesparen-

den Betrieb und eine zentrale Aus-Schaltung ergänzen die Steuerung.

Da der IO 88 auch 8 Eingänge enthält, wurde zusätzlich noch eine kleine Zustandsmeldung über solche Funktionen wie Öffnungszustand der Dach-Lüftungsluke im abseits gelegenen Personalraum, über den Öffnungszustand der Außentüren und ein paar weitere Funktionen integriert. Hier werden, durch die verschiedenen Zustandsmeldungen verknüpft, einfach per Texteditor erstellbare Texte abgerufen und angezeigt.

Die gesamte Kommunikation des Projektes erfolgt drahtlos per Funk über FS20-Sender und -Empfänger sowie die Funk-Wettersensoren.

Die Bedienung wird über den o. g. Touchscreen vorgenommen, dabei wurden die Standorte der Bedienelemente den berührungssensitiven Feldern angepasst. Die Touchscreen-Steuerung tangiert das eigentliche Programm nicht, da es, wie gesagt, lediglich die normale Maus emuliert.

Das Ganze wird durch einen ausgedienten Büro-PC gesteuert, der mit seinen gerade einmal 100 MHz, 32 MB RAM und einer 4-GB-Festplatte für dieses Projekt fast schon überdimensioniert scheint, aber dafür sehr leise arbeitet. Heute bietet sich hier ein extrem preiswertes ITX-System an.

Gerade die abschließende Vorstellung des Klimaprojektes zeigt, dass solch eine Entwicklungsumgebung durchaus keine Elektronik-Spielerei ist, sondern eine äußerst preiswerte Möglichkeit darstellt, auch komplexere Steuerungen schnell und kostengünstig mit Standard-Peripherie realisieren zu können.

Zudem eignet sich das System hervorragend für die Ausbildung. **ELV**

Digital ProfiLab Expert V3.0
45-559-63

€ 99,⁹⁵

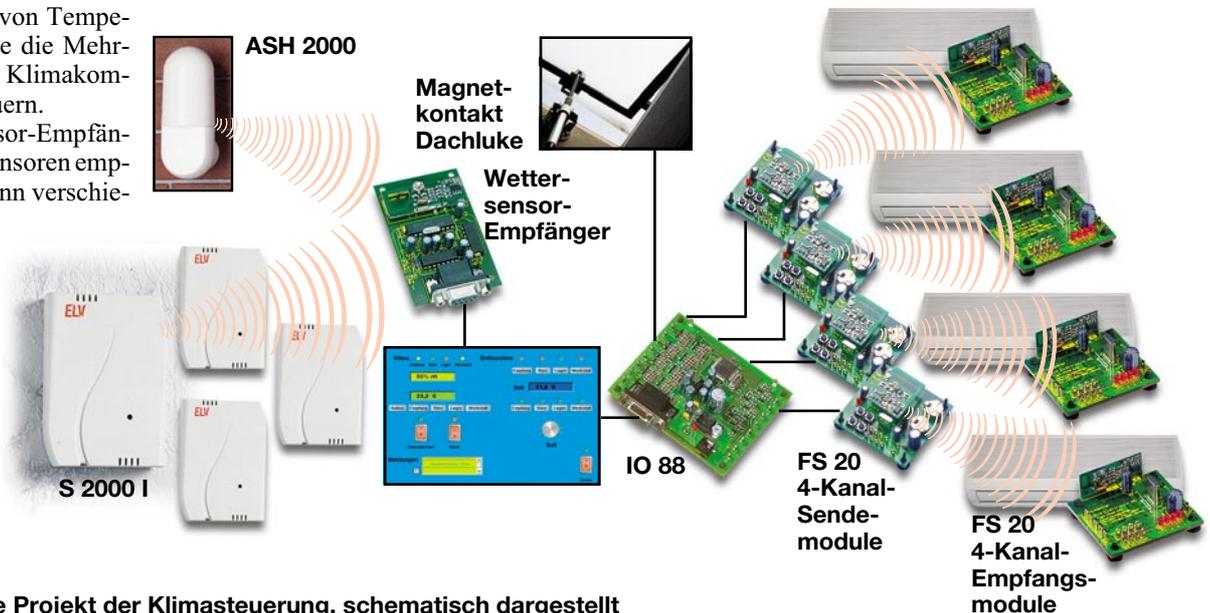


Bild 6: Das gesamte Projekt der Klimasteuerung, schematisch dargestellt