

## ANSI-C, Datenkapselung

Wie können Daten mittels Funktionen, Strukturen (struct) und Pointer verpackt und ohne globale Variablen verschickt werden (z.B. an ein anderes Modul)? Beispiel: Eine RTC-Anwendung (Real-Time-Clock, Uhr).

Das Schlüsselwort **static** beschränkt die Sichtbarkeit einer Variablen auf das Modul (C-Datei), in dem sie verwendet wird. Mit **typedef** können eigene Datentypen "gebastelt" werden. **struct** beschreibt ein Datenpaket - eine Art Hülle -, in dem Datenelemente gekapselt sind. Datenelemente können neben Einfachtypen wie **char** oder **int** wiederum Structs oder Arrays sein. Der Adressoperator **&** liefert die Adresse einer Variablen.

Header: <i>rtc_time.h</i>	C-File: <i>rtc_time.c</i>
<pre> ...  // Datenpaket typedef struct {     BYTE bStunden;     BYTE bMinuten;     BYTE bSekunden } RTC_TIME; // Datentyp  // Dekl. Modulfunktion RTC_TIME * RTC_GetTime(void);  ... </pre>	<pre> ...  static RTC_TIME rtcTime;  // ----- // RTC Sekunden Interrupt // Interrupt_RTCSekunde(void) interrupt {     // Timerdaten abfüllen     rtcTime.bStunden = StdRegister;     rtcTime.bMinuten = MinRegister;     rtcTime.bSekunden = SekRegister; }  // ----- // Zugriff auf Timerdaten // RTC_TIME * RTC_GetTime(void) {     return &amp;rtcTime; // ret Adresse }  ... </pre>

Header: <i>main.h</i>	C-File: <i>main.c</i>
<pre> ...  #include "rtc_time.h"  ... </pre>	<pre> static RTC_TIME * pRTC;  ...  pRTC = RTC_GetTime();  printf("Time= %d : ", pRTC-&gt;bStunden); printf("%d : ", pRTC-&gt;bMinuten); printf("%d \n", pRTC-&gt;bSekunden);  ... </pre>