

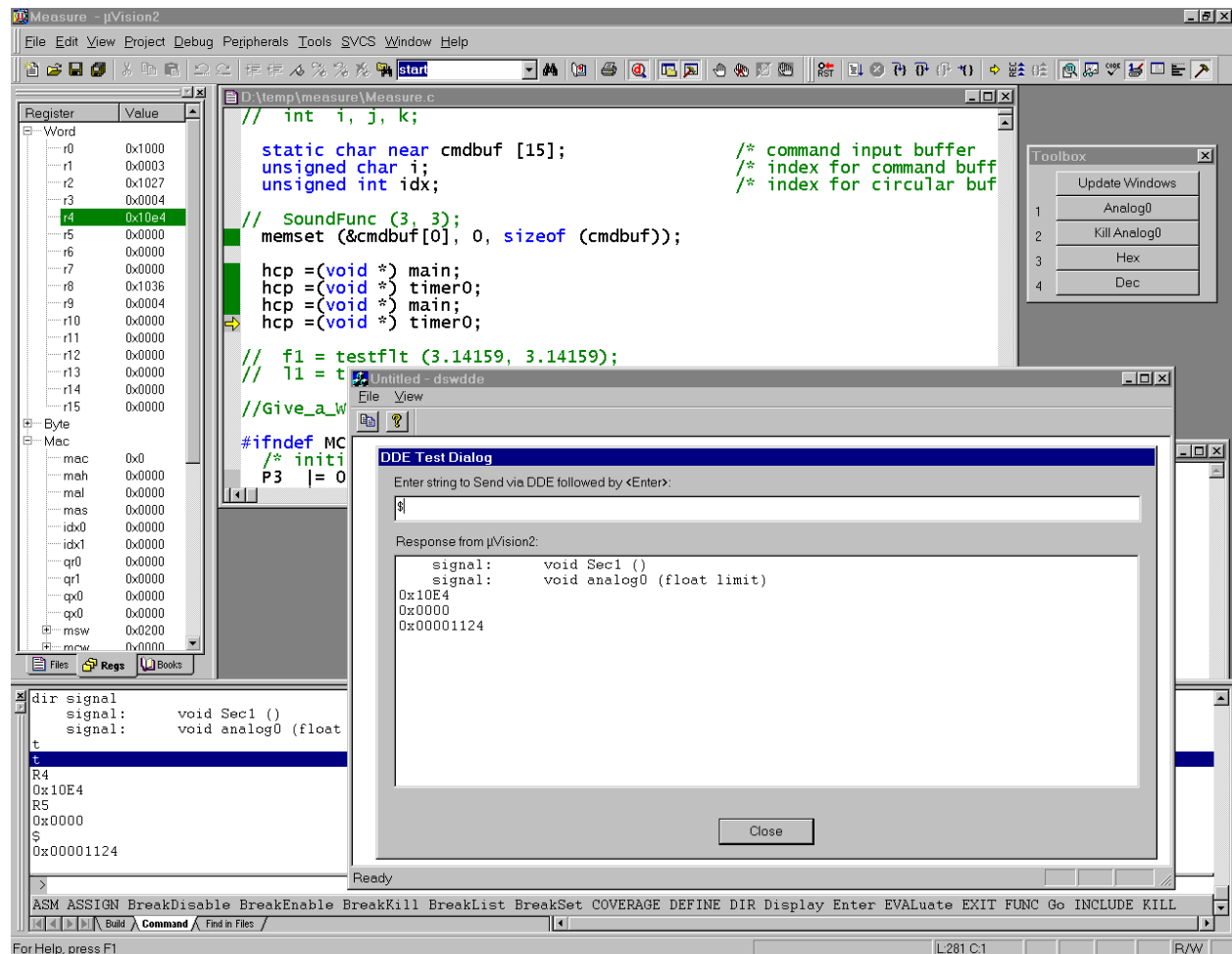
## Using the DDE Interface of the $\mu$ Vision2 Debugger

## Application Note 143

Sept 21, 1999, Munich, Germany

by Peter Holzer, Keil Elektronik GmbH support.intl@keil.com ++49 89 456040-0

The  $\mu$ Vision2 Debugger supports a DDE interface that allows remote control of all debugger functions from external applications. The demo application *DSWDDE.EXE* allows you to enter  $\mu$ Vision2 debugger commands. The debugger output is shown in the *Response from  $\mu$ Vision2* list box. The  $\mu$ Vision2 debugger command window can be still used to enter additional commands. The behavior of the command window does not change.



Sample  $\mu$ Vision2 and DSWDDE screen output

---

## DSWDDE Demo Application

The DSWDDE Demo is available in source form as Microsoft Visual C project and can be freely modified. The following table describes the relevant DDE functions that are implemented in the file MAINFRM.CPP.

---

Function:	Description:
InitDDE	Initialize the DDE interface. $\mu$ Vision2 uses the string "szKeilSW" as name for DDE string handle. This string must not be changed since it identifies the $\mu$ Vision2 debugger DDE connection.
StopDDE	Stops the DDE connection to $\mu$ Vision2 Debugger.
SendCommand	Creates a command string for $\mu$ Vision2 and sends the command via the DDE connection. The command string format is described in the MAINFRM.CPP source file in front of the SendCommand function.
DdeCallback	Is the standard DDE call back function. It handles in the <i>case XTYP_ADVDATA</i> : the command output of the $\mu$ Vision2 debugger. The array <i>&amp;iSt[6]</i> holds the received string from the $\mu$ Vision2 command window. The $\mu$ Vision2 debugger transmits only the output from the commands that are received via the DDE interface.

---

To implement the  $\mu$ Vision2 DDE interface, usually you copy the marked source code of the file **MAINFRM.CPP** to your application. You can use this code license and royalty free within your application.