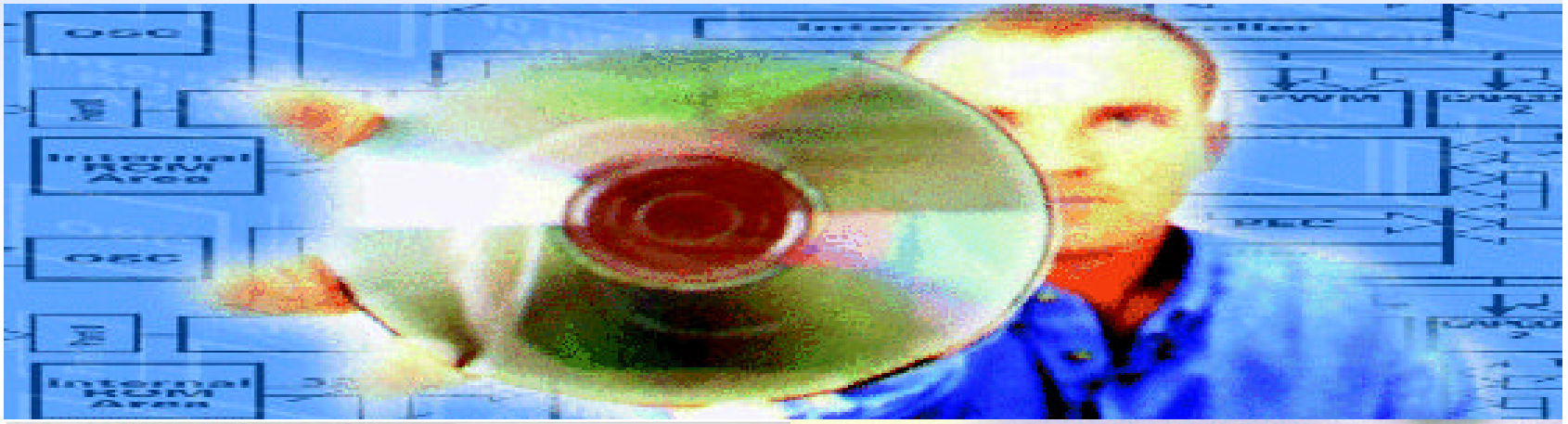
# HOT16x-3

## An Add-On to the HOT16x Hands-On Training Materials for the C166 Family using the fast-view66 Debugger

(from PLS; plus DAvE, the kitCON-16x Starter Kit, the Keil or Tasking C-Compiler, and an oscilloscope)



KEIL SOFTWARE

TASKING
Quality Development Tools Worldwide

pls
Programmierbare Logik & Systeme

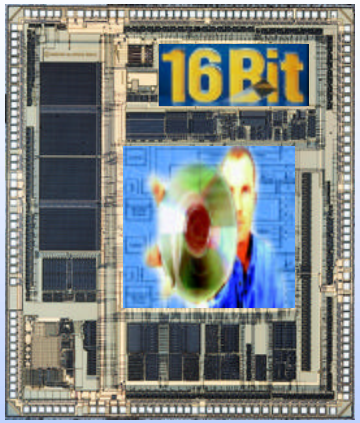
This version is based on DAvE Version 1.0,
Keil µVision2 V2.0, Keil C166 tool chain V4.0,
Tasking EDE V2.1, Tasking C166 tool chain V6.0,
16-bit Starter Kit CD ROM V4.1.
Please report any errors to axel.wolf@infineon.com

# Contents (I)

- **Introduction**
  - Introduction to HOT16x-3
  - Introduction to HOT16x
  - HOT Overview
  - HOT Add-On Overview
  - Short Introduction to DAvE
  - Short Introduction to the PLS fast-view66 HLL Debugger
  - Short Introduction to Keil µVision2
  - Short Introduction to the Tasking Embedded Development Environment EDE

# Contents (II)

- **Using fast-view66 with the HOT16x Hands-On Training Materials**
  - How to set up the Hard- and Software used
  - Hints regarding DAvE and the Exercises
  - How to use fast-view66 with the Exercises from HOT16x-1 (using the Keil Compiler)
  - How to use fast-view66 with the Exercises from HOT16x-2 (using the Tasking Compiler)



# Introduction to HOT16x-3

- **HOT16x-3 is an add-on to the HOT16x-1 and HOT16x-2 training materials for the Infineon C166 family of 16-bit microcontrollers.**
- **While HOT16x-1 is using the integrated Keil debugger and HOT16x-2 is using the Tasking debugger CrossView Pro, HOT16x-3 is using the fast-view66 debugger from PLS.**
- **HOT16x-3 will show you how to run the exercises that are described in HOT16x-1 and HOT16x-2 with the fast-view66 debugger from PLS.**

# Introduction to HOT16x

- **HOT16x is a Hands-On Training material created for the C166 microcontrollers, using**
  - the kitCON-16x Starter Kits
  - the Keil µVision2 integrated development environment (IDE) including the C166 compiler, A166 Assembler, L166 Linker/Locator and Debugger (see HOT16x-1) or
  - the Tasking Embedded Development Environment (EDE) including the C166 compiler, A166 Assembler, L166 Linker/Locator and the Tasking CrossView Pro Debugger (see HOT16x-2)
  - DAvE, the Digital Application Engineer from Infineon Microcontrollers
  - an oscilloscope (for visualization purposes).
  - A Windows95/98 or Windows NT PC (to be able to run DAvE)

# Introduction to HOT16x (cont.)

- **HOT16x shows the user from the scratch how to generate software for the C167CR with DAvE and the Keil or Tasking tool chain:**
  - There are several exercises included, small tasks to be solved using every peripheral of the microcontroller.
  - The user creates a new project in DAvE and configures the device, following the detailed instructions.
  - After having generated the code, the user
    - switches to Keil µVision2 or Tasking EDE,
    - creates a new project (or uses a pre-configured project),
    - includes the C files created by DAvE and the assembler startup file,
    - adds some User Code,
    - compiles, assembles, links and locates the project.

# Introduction to HOT16x (cont.)

- After compilation with µVision2 or EDE, the user
  - switches to the debugger integrated in µVision2 or the CrossView Pro Debugger
  - connects to the kitCON-16x via bootstrap loader (loading the monitor),
  - loads, starts and debugs his example,
  - confirms his working program with a scope (screen shots are included in for most of the examples).

# Introduction to HOT16x (cont.)

- **HOT16x is made of two parts:**
  - An architectural overview introducing the C166 architecture and one specific derivative in particular.
  - The hands-on training part. In this part, the peripherals of the used microcontroller are described in detail followed by a dedicated example using this peripheral or a combination of already introduced peripherals.

  - The first HOT16x example shows the user from the scratch how to generate software with the used tool chain, including the setup of the µVision2 or EDE project.
  - The following examples in HOT16x-1 are based on an already prepared µVision2 project.
  - The following examples in HOT16x-2 include the generation of a new EDE project for each exercise.

# HOT Overview

| Name of Hands-On-Training | C166 Tools used | | | |
|---|---|---|---|---|
| | Compiler, Assembler, Linker, Locator | | Debugger | |
| | Keil µVision2 (Keil C166, A166, L166) | Tasking EDE (Tasking C166, A166, L166) | Keil Debugger in µVision2 | Tasking CrossView Pro |
| HOT161-1 (C161O) | X | - | X | - |
| HOT161RI-1 | X | - | X | - |
| HOT161CI-1 | X | - | X | - |
| HOT163-1 | X | - | X | - |
| HOT164-1 | X | - | X | - |
| HOT165-1 | X | - | X | - |
| HOT167-1 (C167CR/CS) | X | - | X | - |

# HOT Overview (cont.)

| Name of Hands-On-Training | C166 Tools used | | | |
|---|---|---|---|---|
| | Compiler, Assembler, Linker, Locator | | Debugger | |
| | Keil µVision2 (Keil C166, A166, L166) | Tasking EDE (Tasking C166, A166, L166) | Keil Debugger in µVision2 | Tasking CrossView Pro |
| HOT161-2 (C161O) | - | X | - | X |
| HOT161RI-2 | - | X | - | X |
| HOT161CI-2 | - | X | - | X |
| HOT163-2 | - | X | - | X |
| HOT164-2 | - | X | - | X |
| HOT165-2 | - | X | - | X |
| HOT167-2 (C167CR/CS) | - | X | - | X |

# HOT Add-On Overview

| Name of Add-On | C166 Tools used: Compiler, Assembler, Linker, Locator: Keil µVision2 (Keil C166, A166, L166) | C166 Tools used: Compiler, Assembler, Linker, Locator: Tasking EDE (Tasking C166, A166, L166) | C166 Tools used: Debugger: Fast-view66 (PLS) | C166 Tools used: Debugger: Hitop (Hitex) |
|---|---|---|---|---|
| HOT16x-3 | X | X | X | - |
| HOT16x-4 | X | (X)* | - | X |
| HOT16x-5 (kitCON CAN/LCD) | X | X | - (Flash Tools used) | - (integrated Debuggers) |

* in preparation

# Short Introduction to DAvE

- **DAvE is your Digital Application Engineer from Infineon Microcontrollers.**
- **DAvE can help you compare and evaluate the different members of the Infineon C500 (8-Bit) and C166 (16-Bit) families of microcontrollers and help you find the right chip for your embedded control application.**
- **DAvE can be your one-stop access point to all standard knowledge associated with Infineon embedded technology expertise by offering you context sensitive access to user's manuals, data sheets, application notes etc. directly in your development environment.**
- **DAvE can help you program the Infineon microcontroller you want to use in your project, by offering you intelligent wizards that help you configure the chip to work the way you need it and automatically generate C-level templates with appropriate access functions for all of the on chip peripherals and interrupt controls.**
- **More DAvE info at www.infineon.com/DAvE.html**

# Short Introduction to the PLS fast-view66 Development Environment

- **Fast-view66/WIN is an effective development environment for C166 applications**
- **It offers a collection of state-of-the-art software tools including source file management, project building, powerful HLL debugger with various high-speed communication paths to the real hardware system with target monitor**
- **The workbench is open for other tools like configuration management or CASE, including structured debugging**
- **The goal: Minimized turn-around time for effective programming**
- **The fast-view66 HLL debugger is a useful tool for testing microcontroller applications created in the C-language and/or assembler**
- **The debugger is not dependent on a specific C166 hardware system**

- **More information is available on the Starter Kit CD ROM.**

# Short Introduction to the Keil µVision2 Integrated Development Environment

- **Keil µVision2:**
  - µVision2, the IDE from Keil Software, combines Project Management, Source Code Editing, and Program Debugging in one powerful environment. The Quick Start guide on the starter Kit CD ROM gives you the information necessary to use µVision2 for your own projects. It provides a step-by-step introduction of the most commonly used µVision2 features including:
    - Project Setup for the Make and Build Process
    - Editor facilities for Modifying and Correcting Source Code
    - Program Debugging and Additional Test Utilities

- **More information is available on the Starter Kit CD ROM or at www.keil.com.**

# Short Introduction to the Tasking Embedded Development Environment (EDE)

- **EDE:**
  - TASKING's Embedded Development Environment is a package of program building, editing, code generation and debugging tools that provides:
    - Accessible push-button control over a variety of development tasks spread over many tools
    - Tight integration of tools enabling a rapid edit-compile-debug process that leads to higher productivity by automating repetitive tasks

- **C Compiler:**
  - The C 166 compiler is designed and built specifically for the 80C166 microcontroller family. This means you get a very efficient compiler that takes full advantage of the microcontroller's architecture without violating the ANSI standard.

# Short Introduction to the Tasking Embedded Development Environment (EDE) (cont.)

- **Assembler:**
  - The TASKING assembler is an integral part of the tool set but delivers features that enable it to be used on its own. It is supplied complete with linker/locator, librarian and object format utilities.

- **Linker/Locator:**
  - The linker and locator is an essential part of the software building process that enables you to link and locate modules in target memory.

- **More information is available on the Starter Kit CD ROM or at www.tasking.com.**

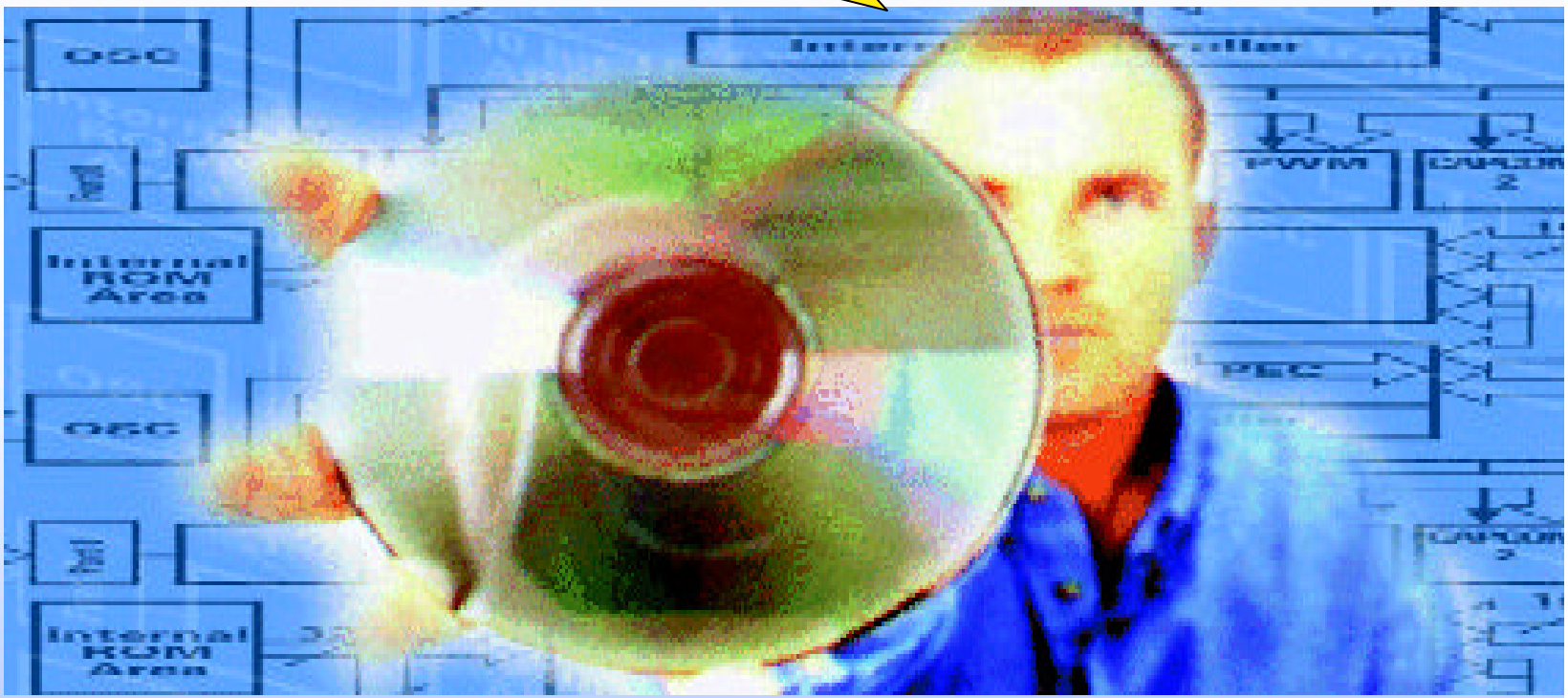
# Using fast-view66 with the HOT16x Hands-On Training Materials

- **How to set up your system: kitCON-16x HW setup**
  - see respective HOT16x material.
- **How to set up your system: DAvE installation**
  - see respective HOT16x material.
- **How to set up your system: Keil µVision2 installation**
  - see respective HOT16x material.
- **How to set up your system: Tasking EDE installation**
  - see respective HOT16x material.
- **How to set up your system: Exercise directory structure**
  - see respective HOT16x material.
- **Hints regarding DAvE and the Exercises**
  - see respective HOT16x material.

# How to set up your system: PLS fast-view66 installation

- **Insert the Starter Kit CD 4.0 or higher into your CD ROM drive.**
- **Wait for the Auto Start.**
  **(Acrobat Reader is required but also included on the CD: ...\install\reader\)**
- **Go to Third Party Development Tools**
- **Go to PLS**
- **Click setup.exe to start fast-view66/WIN installation. The following text will assume that the fast-view66 debugger is installed in the default directory c:\wview66**
- **Follow the instructions on the screen**
- **Choose your COM port you intend to use**
- **You will see four icons to start fast-view66, one for each starter_kit**

Let’s get started now!



# How to use the pls fast-view debugger with Keil µVision2


- **Select one of the exercises described in HOT16x-1, for example Exercise 7GPT1_1 from HOT167-1 for the C167CR**
- **Start DAvE and perform the DAvE configurations described in the exercise, generate the code**
- **Start µVision2 and perform the µVision2 configurations described in the exercise. Build the example (µVision2 creates the file "7GPT1_1.(noextension)")**
- **Start the fast-view66 debugger for the corresponding kitCON-16x starter kit (kitCON-167 in our example)**
- **Go to File | Program to download the program 7GPT1_1. Select the correct directory first. (Our example: c:\hot167_1\7gpt1_1\...)**

# How to use the pls fast-view debugger with Keil µVision2 (cont.)

- **The program will be loaded and displayed in the program window.**
- **If fast-view66 is offering to save everything as a fast-view66 project, press cancel**
- **Make your oscilloscope connections**
- **Hit GO**
- **Fast-view66 will show you that the program is running**
- **Press the red halt-button to stop program execution**
- **The program will halt in the endless loop in main()**

# How to use the pls fast-view debugger with Tasking EDE

TASKING
Quality Development Tools Worldwide

- Select one of the exercises described in HOT16x-2, for example Exercise 7GPT1_1 from HOT167-2 for the C167CR
- Start DAvE and perform the DAvE configurations described in the exercise, generate the code
- Start EDE and perform the EDE configurations described in the exercise.
- Make sure that EDE outputs an .OUT file by selecting this output format in the EDE menu EDE | Linker/Locator Options | Format
- Build the example (EDE creates the file 7GPT1_1.OUT)
- Start the fast-view66 debugger for the corresponding kitCON-16x starter kit (kitCON-167 in our example)
- Go to File | Program to download the program 7GPT1_1.OUT
  Select the correct directory first and change the default filename selection from "*. " to "*.out").
  (Our example: c:\hot167_1\7gpt1_1\...)

# How to use the pls fast-view debugger with Tasking EDE (cont.)

- **The program will be loaded and displayed in the program window.**
- **If fast-view66 is offering to save everything as a fast-view66 project, press cancel**
- **Make your oscilloscope connections**
- **Hit GO**
- **Fast-view66 will show you that the program is running**
- **Press the red halt-button to stop program execution**
- **The program will halt in the endless loop in main()**