Working with Dallas 390 CPU in Contigious Addressing Mode

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-----------------------------------------------------------------------------

NOTE:

For supporting of the Dallas 390, 400, 5240 CPU in Contigious Mode,

the Keil PK51 offers the extended LX51 linker/locater and the extended

AX51 Macro Assembler. The instruction set itself is supported directly in

the Keil C51 Compiler with additional C51 run-time library sets.

These components are only part of the Keil PK51 package and not available in

the CA51 or DK51 toolkit.

-----------------------------------------------------------------------------

The following README file explains how to setup and how to use the Keil PK51

toolchain for the Contigious Addressing Mode of the Dallas 390 CPU and Dallas

5240 CPU.

Tool Setup

==========

The following steps are required to setup the toolchain for Dallas Contigious Mode:

a) Create a new Project with Project - New Project and select from the uVision

Device Database the Dallas DS80C390 (or DS5240) CPU. In the dialog Project -

Select Device for Target - CPU enable both options:

Use LX51 instead of BL51

Use AX51 instead of A51

b) Now you can select in the dialog Project - Options for Target - Target -

Code ROM Size: Contigious Mode: 512K program or 16MB program. This option

configures all tools including debugger/simulator for the Dallas contigious

mode. However, you need a proper startup sequence in your application code

that sets the SFR ACON in the Dallas CPU also to this operating mode.

c) The file Start390.A51 should be included for using the Dallas Contiguous Mode.

This file has a configuration section where you need to set hardware related

options. For example: extended memory map, chip enable output pins, or memory

size attached to chip enable output pins.

d) If your C program contains initialized variables, the file ..\C51\LIB\INIT.A51

must be included to your project. You must set XBANK = 1 when far variables

should be initialized and set DS390 = 1 when CPU runs in Dallas Contiguous Mode.

\*\*\* IMPORTANT NOTE \*\*\*: the file INIT.A51 needs to be the last file of the

linker input list. If you are using uVision this file should be therefore the

last file in your project tree.

Additional Memory Spaces

========================

In addition to the memory classes of the classic 8051 CPU, the extended 8051

toolchain adds three additional memory classes. These memory classes are

described in detail the Assembler/Utilities User's Guide (A51.PDF), Chapter 2

Architecture Overview, Extended 8051 Variants.

Class C51 memory type Allows Addressing of...

---------------------------------------------------------------------------------

HCONST const far complete CODE space C:0 - C:0xFFFFFF for constant

variables.

HDATA far complete XDATA space X:0 - C:0xFFFFFF for variables

ECODE C program code complete CODE space for program code.

Porting Existing 8051 Assembler Code to the Dallas Contigious Mode

==================================================================

As explained above the new toolchain supports additional memory classes for

addressing the extended memory areas of the Dallas 390 CPU. However, if you are

porting existing 8051 assembler code to the Dallas Contigious Mode, the existing

code is still used 16-bit address calculations. If you do not want to work

over the existing 8051 assembler code, you can still use the old traditional

memory classes CODE and XDATA. These memory classes ensure that the objects

are located in the lower 64KB of the CODE and XDATA region. In this way

existing 16-bit address calculation will still work correct.

Porting Existing C Code to the Dallas Contigious Mode

=====================================================

You can find a startup code for the Dallas Contigious Mode in the folder

Keil\C51\Examples\Dallas 390\C. It is recommended that you are using the

START390.A51 file also for own projects.

Function Pointers

=================

Since the Dallas 390 supports up to 16MB program space, it is no longer possible

to use code \* as function pointers. You need to use a generic pointer when

you define a function pointer.

Accessing 16MB memory on the Dallas 390

=======================================

C51 provides two new memory types that can be used to access the extended XDATA

and the extended CODE space. The memory types are: far and const far. The

following example shows you how to use these memory types:

char far farray[0x300]; // this is a variable in HDATA space

const char far ctext[] = "This is a string in ECODE space";

Downloading Programs to the Systronix uCAN2 Evaluation Board

============================================================

You can download the programs written with the Keil C51 Compiler to the

Systronix uCAN2 Evaluation Board. The following steps are required:

a) Connect the COM1(Loader) port on the Systronix uCAN2 Evaluation Board

to the PC as described in the Systronix user's guide.

b) Switch the board to the Loading Mode by pressing LOAD button.

c) Open a terminal emulation (like Hyperterminal) and start a connection

with 19200baud (8bits, no parity, 1 stop-bit, xon/xoff).

d) Press Enter to start the loader and send the command L <Enter> to the

board.

e) Now you can send the generated HEX-File to the board. In Hyperterminal

you can use the menu command 'Send - Text File' to perform this action.

f) After loading you can execute the program by pressing 'Reset'.

------------------------------------------------------------------------------

NOTE:

Currently is no Keil ROM Monitor available for the Dallas 390 contigious mode.

------------------------------------------------------------------------------