

## OVERVIEW

MON51 supplied with C51 V5.50 adds support for target debugging of code banking application programs. You can easily configure MON51 to work with your code banking target. This Application Note explains the steps required to build and configure MON51 to work with your target hardware and code-banking support.

## CONFIGURING MON\_BANK.A51

The first step in building MON51 is to configure the MON\_BANK.A51 file found in the \C51\MON51 directory. The heading of this source, reproduced below, contains several Assembly EQU constant definitions which you use to configure the number of banks and manner of performing bank switching that MON51 uses on your target system.

```
$NOCOND DEBUGPUBLICS
;-----
; This file is part of the Monitor-51 Version 3 package
; Copyright KEIL ELEKTRONIK GmbH 1998
; Most of the settings in this file must conform with the settings
; in the file L51_BANK.A51 which is used for code banking with the
; BL51 Linker/Locator
;-----
;***** Configuration Section *****
?B_NBANKS      EQU      2      ; Define max. Number of Code Banks (not
;                               ; including XDATA or COMMON bank).
;                               ; The max. value for ?B_BANKS is 32
;                               ; possible values are: 1,2,3,...32
;
?B_MODE        EQU      0      ; 0 for Bank-Switching via 8051 Port
;                               ; 1 for Bank-Switching via XDATA Port
;
?B_BANKSTART   EQU      00000H ; defines the start address of the code
;                               ; banking area
;
?B_BANKEND     EQU      06FFFH ; defines the end address of the code
;                               ; banking area
;
?B_COMMON      EQU      0FFH   ; 0FFH if the COMMON area is not mapped into
;                               ; a code bank.
;                               ; otherwise ?B_COMMON must be set to the
;                               ; bank number which contains the COMMON area
;
?B_XRAM        EQU      0FFH   ; 0FFH if the XDATA RAM area is not mapped
;                               ; into a code bank.
;                               ; otherwise ?B_XRAM must be set to the bank
;                               ; number which contains the XDATA RAM area
?B_MON_DATA_BANK EQU      00H   ; Bank number where monitor data is stored
;
IF ?B_MODE = 0;
;-----
; if ?BANK?MODE is 0 define the following values
; For Bank-Switching via 8051 Port define Port Address / Bits
?B_PORT        EQU      P1      ; default is P1
?B_FIRSTBIT    EQU      5       ; default is Bit 3
;-----
ENDIF;
;
IF ?B_MODE = 1;
;-----
; if ?BANK?MODE is 1 define the following values
; For Bank-Switching via XDATA Port define XDATA Port Address / Bits
?B_XDATA_PORT  EQU      0FFFFH ; default is XDATA Port Address 0FFFFH
?B_FIRSTBIT    EQU      0       ; default is Bit 0
;-----
ENDIF;
;*****
```

## Debugging Code Banking Applications with MON51

APNT\_128

There are nine (9) different symbols you may need to change depending on your hardware configuration.

<b>?B_NBANKS</b>	Defines the number of code banks that your application uses. Valid values are <b>1-32</b> . This number is used to determine the number of additional address lines that are needed by the bank switch process.
<b>?B_MODE</b>	Selects the bank switching “mode”. Valid values are 0 or 1. A value of zero ( <b>0</b> ) indicates that you perform bank switching via a standard 8051 port (like P1 or P3). A value of one ( <b>1</b> ) indicates that bank switching is performed via a latched <b>XDATA</b> address.
<b>?B_BANKSTART</b>	Defines the starting address of your bank switching area. This is the range of ROM whose contents are dependent on the active bank at any particular point in execution. Valid values are <b>0000h-0FFFFh</b> . If your target requires that the debugger copy the common area into each bank, set this to <b>0000h</b> .
<b>?B_BANKEND</b>	Defines the ending address of your bank switching area. Combined with <b>?B_BANKSTART</b> , an address range for bank switching can be resolved. Valid values are <b>?B_BANKSTART-0FFFFh</b> .
<b>?B_COMMON</b>	Defines the bank number where the <b>COMMON</b> area is mapped. The common area contains code that must always be accessible and should not be banked. Valid values are <b>0-32</b> or <b>0FFh</b> (which indicates that the <b>COMMON</b> area is not mapped into a code bank). This switch is intended to give flexibility as to where application code can be located while the monitor is running on your target system. An independent application should not locate <b>COMMON</b> code to a code bank.
<b>?B_XRAM</b>	Defines the bank number where the <b>XDATA</b> area is mapped. Like <b>?B_COMMON</b> , this setting is intended to give some flexibility in allowing you to use external RAM along with the monitor’s requisite Von Neumann memory architecture.
<b>?B_MON_DATA_BANK</b>	Defines the bank number where the Monitor stores its <b>XDATA</b> variables.
<b>?B_PORT</b>	Defines the 8051 port that is used to perform bank switching. Valid values are <b>P0, P1, P2, or P3</b> .  This is only required if <b>?B_MODE</b> is set to zero (0).
<b>?B_XDATAPOINT</b>	Defines the address in <b>XDATA</b> of the latch used for bank-switching.  This is only required if <b>?B_MODE</b> is set to one (1).
<b>?B_FIRSTBIT</b>	Defines the first bit in <b>?B_PORT</b> or in the <b>XDATA</b> latch to use in selecting a bank. The total number of bits used is determined from <b>?B_NBANKS</b> .

### NOTE

*These definitions must match those found in **L51\_BANK.A51** file as described in the 8051 Utilities Manual.*

## RUNNING INSTALL.BAT TO GENERATE A HEX FILE

After you configure **MON\_BANK.A51**, you're ready to build the HEX file to program your device with MON51. This is accomplished via a build batch file called **INSTALL.BAT**, located in the \C51\MON51 subdirectory. The calling convention for this file is as follows:

```
INSTALL serialtype [xdatastart [codestart [PROMCHECK][BANK]]]

serialtype := 0 using TIMER 1 9600 bps at 11.059 MHz CPU Clock
serialtype := 1 using baudr. gen. 9600 bps at 12.000 MHz (80515/80517)
serialtype := 2 using TIMER 2 9600 bps at 12.000 MHz CPU Clock
serialtype := 3 using serial interface 1 9600 bps at 12.000 MHz (80517)
serialtype := 4 using T2 9600 bps at 12 MHz for DALLAS 80C320/520/530
serialtype := 5 using Ser.Ch.1 9600 bps at 12 MHz (DALLAS 80C320/520/530)
serialtype := 6 using external UART 16450/16550
serialtype := 7 using TIMER 1 with self adjusting baudrate
serialtype := 8 using TIMER 2 with self adjusting baudrate
serialtype := 9 using baudr. gen. with self adjusting baudrate
serialtype := 10 using serial interface 1 with self adj. bdr. for 80517(A)
serialtype := 11 using TIMER 2 with self adj. bdr. for DALLAS 80C320/520/530
serialtype := 12 using Ser.Ch.1 with self adj. bdr. for DALLAS 80C320/520/530

xdatastart must be a page-no. between 0 and 0FFh inclusive.
codestart must be a block-no. between 0 and 0ECh inclusive.
PROMCHECK checks whether there is a PROM or RAM at address 0
DO NOT USE THIS OPTION WHEN codestart IS 00 !!!
BANK generates Monitor-51 with banking support

EXAMPLE: INSTALL 0 7F 80 BANK
```

**INSTALL.BAT** needs several parameters in addition to the information stored in **MON\_BANK.A51** to determine how to build the monitor. These are covered below in the order in which they appear on the command line.

<b>SERIALTYPE</b>	Indicates which of several predefined serial communication types should be used. See the descriptions above. Valid values are from 0 to 12.
<b>XDATASTART</b>	Indicates the XDATA page where MON51 should locate its XDATA variables. <b>MON51</b> requires one 256-byte page of XDATA for its own uses. This area must be mapped as both XDATA and CODE space. Valid values are from 00h to 0FFh.
<b>CODESTART</b>	Indicates the CODE page where the MON51 code should be stored. MON51 requires 5kb of code space for its own uses, starting on an even page boundary. Valid values are from 00h to 0FFh.
<b>PROMCHECK</b>	Causes MON51 to check whether ROM or RAM is present at address 0 by writing to XDATA address 0 and attempting to read the same value back from CODE address 0. This cannot be used when <b>CODESTART</b> is set to 0 or the PROMCHECK will corrupt MON51.
<b>BANK</b>	Generates MON51 with banking support. This is required for code banking applications.

## CONCLUSION

By configuring MON\_BANK.A51 and INSTALL.BAT correctly, you'll be on your way to target debugging with MON51 in no time. Questions about this application note or any other Keil Software product can be emailed to support@keil.com, or you may contact the Technical Support Department at 1-800-348-8051.

David Lively  
Keil Software, Inc.

---

Copyright © 1998 Keil Software, Inc. All rights reserved.

In the USA:  
**Keil Software, Inc.**  
16990 Dallas Parkway, Suite 120  
Dallas, TX 75248-1903  
USA

Sales: 800-348-8051  
Phone: 972-735-8052  
FAX: 972-735-8055

E-mail: sales.us@keil.com  
support.us@keil.com

Internet: <http://www.keil.com/>

In Europe:  
**Keil Elektronik GmbH**  
Bretonischer Ring 15  
D-85630 Grasbrunn b. Munchen  
Germany

Phone: (49) (089) 45 60 40 - 0  
FAX: (49) (089) 46 81 62

E-mail: sales.intl@keil.com  
support.intl@keil.com