



## **EM BEACON FIRMWARE**

# PORTING THE CODE TO GCC4.4 COMPILER

### Scope

Your EM Beacon code has been developed under a GCC toolchain based on Ride7 IDE. The GCC toolchain was based on GCC3.2 compiler. It now comes with the next release GCC4.4. For different reasons it might be useful to use GCC4.4 instead of GCC3.2.

- 1. GCC3.2 is not supported by Windows 10 and next coming Windows operation system
- 2. GCC3.2 maintenance is frozen.
- 3. All bug fixes and new features will only appear in upgrades of GCC version 4.4 (4.x, 5.x, etc.).
- 4. GCC4.4 offers some better optimization results (typically from 10 to 15% less code footprint produced)
- Next releases of the Beacon firmware will be offered for GCC4.4 and upgraded version of GCC4.4.

During this interim period or if you are working on an older Beacon firmware version, porting your Beacon firmware might be required.

This technical note describes the few modifications you need to implement in your project to comply with GCC4.4 (i.e. it describes the list of required modifications to port your Beacon firmware from GCC3.2 to GCC4.4).

Some aspects of the modifications are not deeply explained in this document. To get a deeper understanding, refer to the "Raisonance tools for C816 family – Getting Started" document [GettingStartedC816\_Ride7.pdf file] located under <RideInstallDir>\Raisonance \Ride\Doc\C816 folder.

This Technical Note comes with a special release of the Beacon firmware 3.0.0 prepared for GCC4.4 toolchain: this release has already been updated according to the description enclosed in this document. It can be used as an example you may refer to if needed.



#### Enable your GCC4.4 toolchain

To do so, go to the Ride\GNU directory and run the setGCC4.bat batch file (some systems require that you are signed in as Administrator). After this, Ride7 will compile using GCC 4.4.

After switching to GCC4, there are a few things that you will need to do to compile your Beacon project (originally developed under GCC3.2).

You must follow the instructions below carefully.

#### Modification of the linker script ctr0.ld

The file is located under your Beacon source (src) folder <BeaconDir>\EMBCxx\_beacon\src

1. Add sub-sections in all sections to allow code elimination and avoid linking error. It is required for all sections (text, data, etc ...)

Example: .rodata : /\* the constant values go after the code \*/ /\*----------\*/ { . = ALIGN(32); /\* align on row beginning \*/ \_srodata = .; \*(.rodata) \*(.rodata\*) /\* required with GCC4 dead code removal optis \*, \*(.rodata.\*) /\* required with GCC4 dead code removal optis erodata = .; } > data rom

Note : This change is backwards compatible with GCC3.2. Therefore you don't need to change anything in case you use this new version of crt0.ld under GCC3.2 toolchain.

2. The other crt0.ld modification concerns page0 allocation and linking section order.

To avoid allocation problems by the linker with page0, you must modify your linker script and move all ".page0\*" (.page0\_bss, .page0\_data, etc.) sections to the first position in the SECTIONS part of the file, before all non-page0 data (.bss, .data, etc.).

Note : This change is backwards compatible with GCC3.2. Therefore you don't need to change anything in case you use this new version of crt0.ld under GCC3.2 toolchain.



#### **Compiler options & settings**

1. Do not use GCC standard headers

Make sure this Compiler Dialect option is deactivated when using GCC4 – as shown on Figure 1: Compiler dialect settings for GCC4.4

perties				
Configuration:	CoinCell		•	
Application Of     Info     Directorie     Advanced C8     GCC compiler     Defines     Compiler     Code     Optimize     Warnings     More     As assembler     Dinker     Post Link     EM6819-Moni	es 116 Options Output	Dialect     ANSI restricted     Traditional C support     Default 'char' type     Use GCC standard includes (not reco	No No unsigned mmended) No	
	•			
• [				

Figure 1: Compiler dialect settings for GCC4.4

Note : This change is not backwards compatible with GCC3.2. Therefore under GCC3.2 toolchain the "use GCC standard includes (not recommended)" Dialect settings must be set to Yes.

2. Use –Os Optimization level

Make sure this level is set when using GCC4 – as shown on Figure 2: Compiler Optimize settings for GCC4.4 3. Enable the Ignore Inline option

Make sure this level is enabled when using GCC4 – as shown on Figure 2: Compiler Optimize settings for GCC4.4 4. Enable the per function section option

Make sure this level is enabled when using GCC4 – as shown on Figure 2: Compiler Optimize settings for GCC4.4

Properties		
Configuration: Coir	Cell	•
<ul> <li>Application Options</li> <li>Advanced C816 Op</li> <li>GCC compiler</li> <li>Dialect</li> <li>Defines</li> <li>Compiler Outpu</li> <li>Code</li> <li>Optimize</li> <li>Warnings</li> <li>More</li> <li>As assembler</li> <li>D LD Linker</li> <li>Post Link</li> <li>EM68 19-Monitor Compiler Compiler</li> </ul>	Ignore Inline Force Inline Opti size Per function sections	-Os Yes No 4 Yes
< III	•	
	Reset t	to default this group options Close

Figure 2: Compiler Optimize settings for GCC4.4

This document is the property of EM MICROELECTRONIC-MARIN SA upon the condition that all rights originating in the information, whether patented or not, will be respected.



Note : These changes are not backwards compatible with GCC3.2. Therefore under GCC3.2 toolchain the "Compiler Optimize settings" must be set as shown on Figure 3: !!! Compiler Optimize settings for GCC3.2 !!!

Properties			x
Configuration: CoinCell		•	
Application Options     Advanced C816 Options     GCC compiler     Dialect     Compiler Output     Code     Optimize     Warnings     More     As assembler     Du Linker     Dest Link     EM6819-Monitor Configura	Coptimize Optimization level Ignore Inline Force Inline Opti size Per function sections GCC3.2 previous settin t	-01 No A No Prosent	
4 III >	Rese	et to default this group options	e

Figure 3: !!! Compiler Optimize settings for GCC3.2 !!!



### Linker options & settings

- Disable Data Init feature Make sure this option is disabled when using GCC4 – as shown on Figure 4: Linker General settings for GCC4.4
- 2. Remove unused sections Make sure this option is enabled when using GCC4 – as shown on Figure 4: Linker General settings for GCC4.4
- 3. Remove Program Headers Alignment as shown on Figure 1: Compiler dialect settings for GCC4.4 Make sure this option is deactivated when using GCC4 – as shown on Figure 4: Linker General settings for GCC4.4

perties			
Configuration:	CoinCell		•
Application C     Advanced CE     GCC compile     S- AS assemble     LD Linker     General     Startup     St	16 Options	General Generate MAP file Include Cross References Generate Data Init Warn undefined symbols Remove unused sections Nign Headers	Yes No No Yes No
•	4		
		Reset to de	efault this group options Close

Figure 4: Linker General settings for GCC4.4

Note : This change is not backwards compatible with GCC3.2. Therefore under GCC3.2 toolchain the "General linker settings" must be set as shown on Figure 5: !!! Linker General settings for GCC3.2 !!!

Properties				x
Configuration:	CoinCell		•	
Application Op Info Directories Advanced C8 GCC compiler AS assembler LD Linker LD Linker Scripts Libraries More Post Link B- EM6819-Monit	s 16 Options	General Generate MAP file Include Cross References Generate Data Init Warn undefined symbols Remove unused sections Align Headers GCC3.2 previous	Yes No Yes No No No Settings !	
<	•	Reset	to default this group options	2

Figure 5: !!! Linker General settings for GCC3.2 !!!



#### Post link settings

Disable the Post Link option.

Make sure this option is deactivated when using GCC4 – as shown on Figure 6: Post Link setting

Properties					×
Configuration:	CoinCell			•	
- Application Op - Info - Directorie - Advanced C8 - GCC compiler - AS assembler - LD Linker - EM6819-Monit	s 16 Options	Settings     Post Link Action     Command line		No cmd /C move /Y *.d .\bui	ld
•	4				
			Reset to defa	ault this group options	Close

Figure 6: Post Link setting

Note : This change is not backwards compatible with GCC3.2. Therefore under GCC3.2 toolchain the "Post Link option" setting must be defined as shown on Figure 7: !!! Post Link settings for GCC3.2 !!!

Properties					×
Configuration:	CoinCell			•	
Application Op     Advanced C8:     GCC compiler     As assembler     D Linker     D Linker     Erost Link     E- EM6819-Monit	16 Options	Post Link Action Command line		Yes cmd /C move /Y *.d .\build s !	
٠ III					
			Reset to defau	ult this group options	Close

Figure 7: !!! Post Link settings for GCC3.2 !!!

This document is the property of EM MICROELECTRONIC-MARIN SA upon the condition that all rights originating in the information, whether patented or not, will be respected.



Contact Information

**Technical support:** 

http://www.emdeveloper.com support@emdeveloper.com