



APU-2 Engagement Package **Release Notes**

UG-10301-03-07

Copyright

Copyright © 2015 CogniVue Corporation ("CogniVue") All rights reserved.

This document contains information which is proprietary to CogniVue and may be used for non-commercial purposes within your organization in support of CogniVue's products. No other use or transmission of all or any part of this document is permitted without written permission from CogniVue, and must include all copyright and other proprietary notices. Use or transmission of all or any part of this document in violation of any applicable Canadian or other legislation is hereby expressly prohibited.

User obtains no rights in the information or in any product, process, technology or trademark which it includes or describes, and is expressly prohibited from modifying the information or creating derivative works without the express written consent of CogniVue.

Disclaimer

CogniVue assumes no responsibility for the accuracy or completeness of the information presented which is subject to change without notice. In no event will CogniVue be liable for any direct, indirect, special, incidental or consequential damages, including lost profits, lost business or lost data, resulting from the use of or reliance upon the information, whether or not CogniVue has been advised of the possibility of such damages.

Mention of non-CogniVue products or services is for information purposes only and constitutes neither an endorsement nor a recommendation.

Revision History

Version	Details of Change	Author	Date
01	Initial Release	W. Hulme	Mar. 31, 2013
02	Update for APU2 Tools R2.1	W. Hulme	Apr. 17, 2013
03	Update for APU2 Tools R2.2	W. Hulme	July 3, 2013
04	Update for APU2 Tools 13R1.2	W. Hulme	Nov. 18, 2013
05	Update for APU2 Tools 13R1.4	W. Hulme	Mar. 5, 2014
06	Update for APU2 Tools 13R1.6	W. Hulme	Mar. 28, 2014
07	Update for APU2 Tools J-2014.09-SP1-1	Lee, Ki-ju	Feb. 10, 2015

Table of Contents

1	General Document Information	5
1.1	Acronyms	5
1.2	Scope	5
2	Release Identification	6
2.1	Associated Documentation	6
2.1.1	Tool User Guide	6
2.1.2	Programmer Guide	6
2.1.3	APU-2 Sample Code Overview	6
3	New In This Release	8
3.1	Changes in Release J-2014.09-SP1-1	8
3.2	Changes in Release 13R1.6	9
4	Known Issues	10
5	Miscellaneous	11
5.1	Compiler Warnings and Errors	11
5.1.1	CHESS Compiler Errors	11
5.1.2	ISS Warnings and Errors	12

Table of Tables

Table 1-1: Acronyms5

1 General Document Information

This document provides information specific to a given release of the APU-2 C tools. The specific release is identified, along with the versions of all associated software and documentation.

1.1 Acronyms

The following acronyms are used in this document, or in documents related to this release.

Acronym	Definition
ACP	Array Control Processor
APU	Array Processing Unit
CMEM	Computational Unit Memory
CPU	Central Processing Unit
CU	Computational Unit
DMA	Direct Memory Access
DMEM	Data Memory
IDE	Integrated Development Environment
IMEM	Instruction Memory
ISS	Instruction Set Simulator
MIMD	Multiple Instructions Multiple Data
OpenCV	Open Source Computer Vision
RISC	Reduced Instruction Set Computer
SIMD	Single Instruction Multiple Data

Table 1-1: Acronyms

1.2 Scope

This document applies only to a specific release at the time of delivery. The content of this document may be affected by subsequent patches and releases.

2 Release Identification

This document pertains to Release J-2014.09-SP1-1 of the APU-2 Tools

2.1 Associated Documentation

This section gives an overview of all the released documentations for APU-2 Tools.

2.1.1 Tool User Guide

This document contains information related to the installation and operation of the APU-2 compiler and simulation environment.

- Document Name(s) and Number(s):

UG-10301-01-08 – APU-2 Tool User Guide

- Required Background of Reader:

The reader of this document is expected to be a software developer, who intends to prototype, benchmark or develop APU-2 kernels or applications.

- Intended benefit of document:

This document provides guidance for the user of the APU-2 tools.

2.1.2 Programmer Guide

The Programmers Guide contains technical details regarding the vector extensions to C99 implemented in the APU-2 Simulator, and in the APU-2 hardware. This document provides guidance for the user of the APU-2 tools.

- Document Name(s) and Number(s):

UG-10301-00-05 –APU-2 C Programmer Guide

- Required Background of Reader:

The reader of this document must have a background in basic software application development, be aware general RTOS based infrastructures including memory management. This document outlines the differences between C99 and APU-2 Vector C. As such, the reader should be familiar with C99.

- Intended benefit of document:

This document provides detailed background information required to write APU-2 kernels.

2.1.3 APU-2 Sample Code Overview

This document contains information related to the understanding, compiling and profiling of a number of image processing algorithm kernels on the APU-2 platform.

- Document Name(s) and Number(s):

UG-10301-02-04 – APU-2 Sample Code Overview

- Required Background of Reader:

The intended audience is a software developer who is new to the APU-2 architecture, and would like to compile and execute existing code.

- Intended benefit of document:

This document gives the reader a better understanding of the performance capabilities of the APU-2 architecture.

3 New In This Release

3.1 Changes in Release J-2014.09-SP1-1

This release is based on APU-2 Compiler release version J-2014.09-SP1-1

This section lists known issues resolved in versions J-2014.09-SP1-1 of the APU-2 Tools.

- (CLOVER-190) 13R1.3 PostInstall: APU Model Build Fail unless using chessmk -r first
- (CLOVER-336) Problem with vector extensions: break/continue

3.2 Changes in Release 13R1.6

This section lists known issues resolved in versions APU2_13R1.6 of the APU-2 Tools.

- (CLOVER-188) APU lockup on out-of-range address
- (CLOVER-229) Dependency hazard compiler error
- (CLOVER-261) ISS Command line argument support
- (CLOVER-263) Scalar kernel fails in release mode
- (CLOVER-270) Conditional vload issue during simulation
-

For CLOVER-261, "APU2 ISS command line argument support", currently it is the user's responsibility to provide the buffer size in the user application as below:

```
// add 1k memory space to pass main arguments...  
char _main_argv_area[1024];
```

4 Known Issues

This section lists known issues with the APU-2 Tools at the time of delivery.

- (CLOVER-32) The APU-2 Compiler may generate warnings which direct the user to consider using pragmas to help optimize the code.
- (CLOVER-166) vload() and vstore() execution time in simulation.
- (CLOVER-171) Vector condition stack overflow is not reported at compile time.
- (CLOVER-176) Wrongly issued run-time warning of CU selection is outside of range due to compiler optimization.
- (CLOVER-61) On windows, there is a system limitation that a command string can be at most 32767 chars. The file that is being compiled, standalone_instrinsics.c, has 656 functions, all with quite long names which results in a command string of over 40k chars.
For now you can set the following option (-F). This will result in shorter file names being generated.

Go and press "Settings Project settings → C/C++ front end → Extra arguments → Edit" in menu of ChessDE, And add an option -F, keeping with existing options.

5 Miscellaneous

5.1 Compiler Warnings and Errors

The following is intended to provide detailed explanations regarding certain known compiler errors which may arise due to improper usage of the APU-2 C language. For more information, see the “APU-2 C Programmer Guide”.

5.1.1 CHESS Compiler Errors

- Data structures

NB: The compiler will give the following error when trying to instantiate a structure containing both scalar and vector types:

```
Error in "file.c", line #, column #: cannot find suitable stack memory for
variable `var_name' of type `mixed_type'
```

or

```
Error in "file.c", line #, column #: cannot find suitable default memory for
pointer to type `mixed_type' (requires chess_storage(<memory>))
```

To resolve this error, ensure that structures contain only vector or only scalar entities. Note that pointers, whether to vectors or scalar values, are scalar, thus pointers to vectors and pointers to scalars CAN exist in the same structure.

- Scalar Data Structures

The `__cmem` storage qualifier is not allowed for scalar structure and will generate an error.

```
Error in "file.c", line #, column #: cannot allocate variable `var_name' of
type `scalar_type' to storage `VMb'
```

Allowing the compiler to choose the storage, i.e. omitting the storage qualifier altogether, will work in most cases. If a storage qualifier is needed for scalar values, specify `__dmem`.

- Vector Data Structures

The `__dmem` storage qualifier is not allowed for vector structure and will generate an error.

```
Error in "file.c", line #, column #: cannot allocate variable `v' of type
`vector_type' to storage `DMb'
```

Allowing the compiler to choose the storage, i.e. omitting the storage qualifier altogether, will work in most cases. If a storage qualifier is needed for scalar values, specify `__cmem`.

- Vector return value

Vectors can be used as return values, but only if they have register storage. Thus, a vector return value cannot be declared as `__cmem`, or the following error will be generated:

```
Error in "file.c", line #, column #: return value storage specifier `VMb' is
not a register
```

Remove any provided storage specifiers, allowing the compiler to store the variable in a register.

- Long path/directory names

The APU compiler generates temporary files with very long file names. Windows has a limitation on the maximum path/filename combination, which may result in the following error:

Internal error: cannot open file for output: apu_filter_generic-Fvoid_apu_filter_fir2__Pvec08u_sint_Pvec08u_sint_sint_sint_Pschar_sint_sint_sint_bndl.sfg.

Keep path names to source and object directories from being overly long.

5.1.2 ISS Warnings and Errors

- Vector Condition Stack Overflow

There is an architectural limitation on the number of nested levels of vector conditions. When the vector condition stack is overflowed, the B: The ISS issues the following warning to the console: blah:

```
WARNING : WRN (-2) : predicate stack pointer increment outside of increment
range : in function vcsptr_incr (APU2.p:L1772:C15)
```

```
WARNING : WRN (-2) : predicate stack pointer decrement outside of decrement
range : in function vcsptr_decr (APU2.p:L1778:C15)
```

To resolve this error, restructure the vector condition logic to reduce nested condition checks.