

---

# vDSP Complex Vector Conversion Reference

[Performance > Carbon](#)



2007-06-15



Apple Inc.  
© 2007 Apple Inc.  
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Apple's copyright notice.

The Apple logo is a trademark of Apple Inc.

Use of the "keyboard" Apple logo (Option-Shift-K) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Apple retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Apple-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Apple is not responsible for typographical errors.

Apple Inc.  
1 Infinite Loop  
Cupertino, CA 95014  
408-996-1010

Apple, the Apple logo, Carbon, Mac, and Mac OS are trademarks of Apple Inc., registered in the United States and other countries.

Simultaneously published in the United States and Canada.

**Even though Apple has reviewed this document, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.**

**IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY**

**DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.**

**THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.**

**Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.**

# Contents

---

## **vDSP Complex Vector Conversion Reference 5**

Overview 5

Functions 5

    vDSP\_ctoz 5

    vDSP\_ctozD 6

    vDSP\_ztoc 6

    vDSP\_ztocD 7

---

## **Document Revision History 9**

---

## **Index 11**

---



# vDSP Complex Vector Conversion Reference

---

<b>Framework:</b>	Accelerate/vecLib
<b>Declared in</b>	vDSP.h

## Overview

Describes the C API for the vecLib functions that convert complex vectors between interleaved and split forms.

## Functions

### vDSP\_ctoz

Copies the contents of an interleaved complex vector *C* to a split complex vector *Z*; single precision.

```
void vDSP_ctoz (const DSPComplex C[],
                vDSP_Stride strideC,
                DSPSplitComplex * Z,
                vDSP_Stride strideZ,
                vDSP_Length size);
```

#### Discussion

Performs the operation

$$A_{nI} = \text{Re}(C_{nK}) \quad ; \quad A_{nK+1} = \text{Im}(C_{nK}) \quad n = \{0, N-1\}$$

*strideC* is an address stride through *C*. *strideZ* is an address stride through *Z*. The value of *strideC* must be a multiple of 2.

For best performance, *C.realp*, *C.imagp*, *Z.realp*, and *Z.imagp* should be 16-byte aligned.

See also functions "[vDSP\\_ztoc](#)" (page 6) and "[vDSP\\_ztocD](#)" (page 7).

#### Availability

Available in Mac OS X v10.4 and later.

#### Declared In

vDSP.h

**vDSP\_ctozD**

Copies the contents of an interleaved complex vector *C* to a split complex vector *Z*; double precision.

```
void vDSP_ctozD (const DSPDoubleComplex C[],
                vDSP_Stride strideC,
                DSPDoubleSplitComplex * Z,
                vDSP_Stride strideZ,
                vDSP_Length size);
```

**Discussion**

This performs the operation

$$A_{nI} = \text{Re}(C_{nK}) \quad ; \quad A_{nK+1} = \text{Im}(C_{nK}) \quad n = \{0, N-1\}$$

*strideC* is an address stride through *C*. *strideZ* is an address stride through *Z*. The value of *strideC* must be a multiple of 2.

For best performance, *C*.*realp*, *C*.*imagp*, *Z*.*realp*, and *Z*.*imagp* should be 16-byte aligned.

See also functions ["vDSP\\_ztoc"](#) (page 6) and ["vDSP\\_ztocD"](#) (page 7).

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

vDSP.h

**vDSP\_ztoc**

Copies the contents of a split complex vector *A* to an interleaved complex vector *C*; single precision.

```
void vDSP_ztoc (const DSPSplitComplex * Z,
                vDSP_Stride strideZ,
                DSPComplex C[],
                vDSP_Stride strideC,
                vDSP_Length size);
```

**Discussion**

This performs the operation

$$C_{nK} = \text{Re}(A_{nI})$$

$$C_{nK+1} = \text{Im}(A_{nI}) \quad n = \{0, N-1\}$$

*strideC* is an address stride through *C*. *strideZ* is an address stride through *Z*.

For best performance, *C*->*realp*, *C*->*imagp*, *A*->*realp*, and *A*->*imagp* should be 16-byte aligned.

See also ["vDSP\\_ctoz"](#) (page 5) and ["vDSP\\_ctozD"](#) (page 6).

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

vDSP.h

**vDSP\_ztocD**

Copies the contents of a split complex vector A to an interleaved complex vector C; double precision.

```
void vDSP_ztocD (const DSPDoubleSplitComplex * Z,
                vDSP_Stride strideZ,
                DSPDoubleComplex C[],
                vDSP_Stride strideC,
                vDSP_Length size);
```

**Discussion**

This performs the operation

$$C_{nK} = \operatorname{Re}(A_n)$$

$$C_{nK+1} = \operatorname{Im}(A_n) \quad n = \{0, N-1\}$$

strideC is an address stride through C. strideZ is an address stride through Z.

For best performance, C->realp, C->imagp, A->realp, and A->imagp should be 16-byte aligned.

See also ["vDSP\\_ctoz"](#) (page 5) and ["vDSP\\_ctozD"](#) (page 6).

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

vDSP.h





# Document Revision History

---

This table describes the changes to *vDSP Complex Vector Conversion Reference*.

Date	Notes
2007-06-15	New document that describes the C API for the vDSP functionality for converting complex vectors between interleaved and split forms.

**REVISION HISTORY**

Document Revision History

# Index

---

## V

---

- vDSP\_ctoz **function 5**
- vDSP\_ctozD **function 6**
- vDSP\_ztoc **function 6**
- vDSP\_ztocD **function 7**