

Technical Support
Knowledge Center Open

VEE: How Can I Compute Recursive Equations?

Notices

© Keysight Technologies Incorporated, 2002-2020

1400 Fountaingrove Pkwy., Santa Rosa, CA 95403-1738, United States All rights reserved.

No part of this documentation may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Keysight Technologies, Inc. as governed by United States and international copyright laws.

Restricted Rights Legend

If software is for use in the performance of a U.S. Government prime contract or subcontract, Software is delivered and licensed as "Commercial computer software" as defined in DFAR 252.227-7014 (June 1995), or as a "commercial item" as defined in FAR 2.101(a) or as "Restricted computer software" as defined in FAR 52.227-19 (June 1987) or any equivalent agency regulation or contract clause.

Use, duplication or disclosure of Software is subject to Keysight Technologies' standard commercial license terms, and non-DOD Departments and Agencies of the U.S. Government will receive no greater than Restricted Rights as defined in FAR 52.227-19(c)(1-2) (June 1987). U.S. Government users will receive no greater than Limited Rights as defined in FAR 52.227-14 (June 1987) or DFAR 252.227-7015 (b)(2) (November 1995), as applicable in any technical data.

Portions of this software are licensed by third parties including open source terms and conditions.

For detail information on third party licenses, see [Notice](#).

Contents

The solution is as follows:

Refer to the Example Solutions.

This program works as follows:

There are three Real Constants driving the program: an initial value for X, the constant B, and the constant array U.

The output array X is produced dynamically by a collector that obtains the results of each iteration of the computation.

The number of iterations is given by the size of the array B (through the Totsize() function); this is the same as the size of the output array X. The size is loaded into a For Count object, which drives the computation through its iterations and then fires its Sequence Out pin when done.

A Formula box performs the actual computation, $X + B * U[N]$. The input pin X is fed through a Junction that connects to the Init constant (for the initial value in the first iteration) and to the output of the Formula itself (for all other iterations).

The outputs of the Formula box are also put into an array by a Collector and then output when the For Count object reaches the end of its count.

Note that if you are using feedback loops in versions of VEE before 4.0, you will need to put a Start button on the Init object ... it was one of the eccentricities of VEE that it insisted on having one for feedback loops.

