



TECHNICAL REPORT	EIA-EDIF-2™	
	Issued 1989-06 Stabilized 2015-07	
EDIF Connectivity, Volume 2		

NOTICE

This document has been taken directly from the original TechAmerica document and contains only minor editorial and format changes required to bring it into conformance with the publishing requirements of SAE Technical Standards. The release of this document is intended to replace the original with the SAE International document. Any numbers established by the original document remain unchanged.

The original document was adopted as an SAE publication under the provisions of the SAE Technical Standards Board (TSB) Rules and Regulations (TSB 001) pertaining to accelerated adoption of specifications and standards. TSB rules provide for (a) the publication of portions of unrevised specifications and standards without consensus voting at the SAE committee level, and (b) the use of the existing specification or standard format.

SAE Technical Standards Board Rules provide that: "This report is published by SAE to advance the state of technical and engineering sciences. The use of this report is entirely voluntary, and its applicability and suitability for any particular use, including any patent infringement arising therefrom, is the sole responsibility of the user."

SAE reviews each technical report at least every five years at which time it may be revised, reaffirmed, stabilized, or cancelled. SAE invites your written comments and suggestions.

Copyright © 2016 SAE International

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, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of SAE.

TO PLACE A DOCUMENT ORDER: Tel: 877-606-7323 (inside USA and Canada)
Tel: +1 724-776-4970 (outside USA)
Fax: 724-776-0790
Email: CustomerService@sae.org
<http://www.sae.org>

SAE WEB ADDRESS:

Full document details: [SAE EIAEDIF2](#)

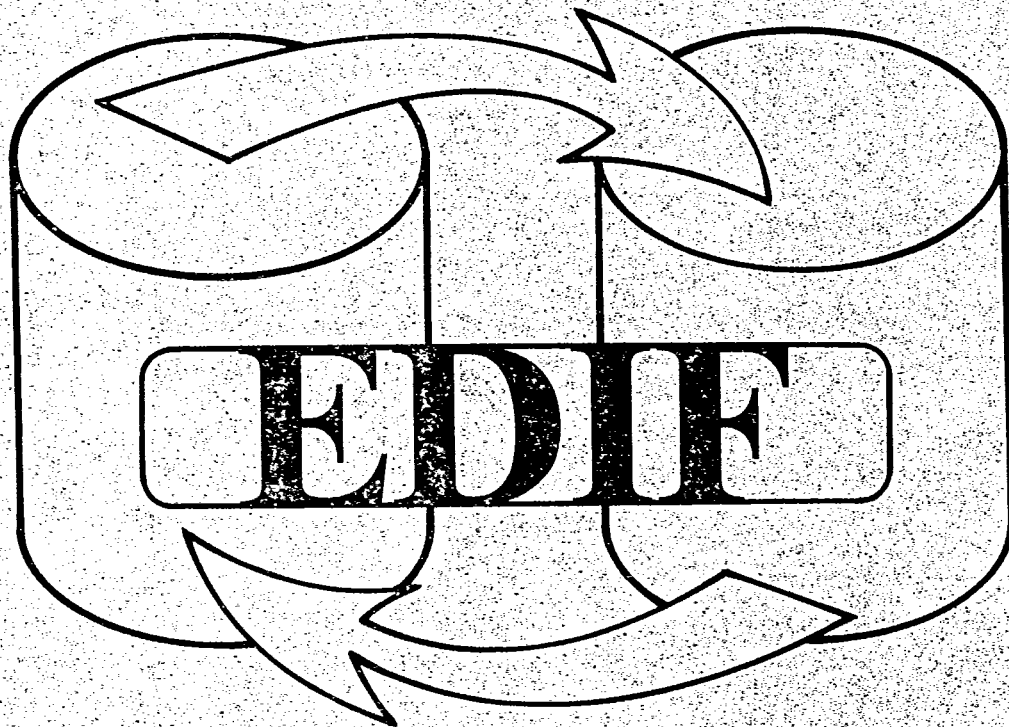
SAE values your input. To provide feedback on this Technical Report, please visit <http://www.sae.org/technical/standards/EIAEDIF2>

Documents
Document Sharing Hub

*Electronic Design
Interchange Format*

EIA/EDIF-2
JUNE 1989

EDIF Connectivity



EDIF Monograph Series

Volume 2



*Electronic Industries Association
EDIF Steering Committee*

EDIF Series

Volume 2

EDIF Connectivity

1989

Electronic Industries Association

EDIF Steering Committee



Documents

Document Sharing Hub

EDIF Connectivity

Published by
ELECTRONIC INDUSTRIES ASSOCIATION
Engineering Department
2001 Eye Street, N.W.
Washington, D.C. 20006
Copyright © 1989. 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, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the Electronic Industries Association.

Consulting Editor: Paul Stanford
Technical Editor: Paul Mancuso

Special thanks are due to Peter Lay for providing the material on which this monograph is based, as part of the Alvey CAD031 Project.

ISBN 0-07908-0002-0

Price: \$18.00

Foreword

This volume is the second in a series of monographs which will make the standard (ANSI/EIA 548-1988) easier to understand. It is intended to be used as a companion to the EDIF Reference Manual Version 2.0.0, which is also published by the EIA (ISBN 0-7908-0000-4).

This volume gives an introduction to the concept of connectivity in the format. Concepts are explained in a general way; constructs, such as *net*, appear in italics throughout this volume. Whenever more detail is desired about a topic the italicized constructs should be consulted in the EDIF Reference Manual. The EDIF Reference Manual contains the official definition of the format and should always be taken as the authoritative source of information. Further application guides will be provided in later volumes of this series.

Contents

Preface	i
Section 1: Connectivity	1-1
1.1 Introduction	1-1
1.2 Hierarchy and Nets	1-1
1.3 Compound Ports	1-5
1.4 Compound Nets	1-6
1.5 Connections between Different Sized Nets	1-11
1.6 Subnets within Nets	1-17
1.7 Exporting Connectivity Information	1-20
1.8 Weakly Joined Ports	1-22
1.9 Permutability of Ports	1-26
1.10 Additional Net Information	1-29
1.11 Summary	1-29
Appendix A	A-1
An Example EDIF Description	A-1



Documents

Document Sharing Hub

Figures

Figure 1. Master and instance ports.	1-2
Figure 2. Global port example.	1-3
Figure 3. Simple connection example.	1-4
Figure 4. Bus connections between master port A of a view and port B of instance I.	1-7
Figure 5. Circuit example.	1-8
Figure 6. Conceptual representation of net bundle.	1-10
Figure 7. Different connections to a bus.	1-11
Figure 8. Schematic of a bus ripper.	1-13
Figure 9. Bus/wire connection and the conceptual EDIF representation.	1-13
Figure 10. Connection of a net array to an instance of ripper cell ripper2.	1-15
Figure 11. A net with tied subnets.	1-19
Figure 12. Alternative connections to ports joined in the interface.	1-21
Figure 13. External connections of joined ports to two nets.	1-22
Figure 14. A valid connection to a pair of mustJoin ports.	1-23
Figure 15. The mustJoin of ports A and B satisfied higher in the hierarchy. ...	1-23
Figure 16. Illustration of connections for the EDIF fragment.	1-25
Figure 17. Valid connections for the EDIF mustJoin fragment.	1-26
Figure 18. Equivalent connections to permutable ports.	1-26
Figure 19. Permutable / nonPermutable example.	1-27
Figure 20. Nesting of permutable and nonPermutable.	1-28
Figure 21. Select cell schematic.	A-2

Preface

The increase in the number of silicon foundries and CAD/CAE system and workstation companies, and the problem of movement of data among in-house design systems has made a standard interchange format for electronic design data essential. The benefits of the general adoption of a standard interchange format are important and far reaching. The customer receives a much wider variety of feasible choices, and free competition in both CAE/CAD system/tool businesses and foundry businesses is enhanced. The electronic circuit designer and customer of CAE/CAD system and foundry service can expect compatibility among these products. The designer can choose equipment and services best suited for a particular task with a minimum of overhead. Foundries can work with customers regardless of the CAE/CAD equipment that the customer is using. The CAD/CAE tool or system vendor can maintain compatibility with a variety of foundries and other systems with a minimum investment in software development for that purpose.

Many interchange formats and hardware description languages have been developed over the past decade. However, each has suffered from one or more of the following drawbacks:

- Narrow focus – many formats have been developed to address only a specific aspect of the design data, such as mask artwork data or netlist information, but not both.
- Proprietary constraints – many formats are proprietary to specific semiconductor or CAD companies, severely limiting their effective use.
- Difficulty of Implementation – often, as a format evolves, it becomes increasingly difficult to develop parser/generator software for dealing with it; this is particularly true if upward compatibility is an issue.
- Difficulty of Extension – the rapid evolution of IC and PCB design methods causes new facets to be added to the data that must be transferred. Many languages or formats cannot grow to meet these new needs.

The Electronic Design Interchange Format has successfully addressed all of these problems and has made the movement of all design data a reality.

Section 1: Connectivity

1.1 Introduction

There are several aspects of an electronic design that must be captured and expressed by a design interchange format. One of these aspects is the connectivity of the circuit. The circuit connectivity or topology may be defined in several ways: as part of a schematic design, as an extraction from some layout description, or as a circuit description in some language such as the input language for a simulator. Connections need to be described between components in various views of the circuit. EDIF, thus, allows descriptions of circuit connections to be made in several views, most notably the mask layout, PCB layout, symbolic, schematic, and netlist views. The netlist view is designed primarily to describe the circuit connections; the layout and schematic views also describe other aspects of the circuit. The way in which the connectivity of a circuit is described is common to all such views.

1.2 Hierarchy and Nets

EDIF supports a hierarchical description of a circuit. The circuit is broken down into a number of cells, various aspects of which are described in views. A view may instantiate other views, forming a hierarchy. Within a view, the instantiated views of other cells may be regarded as black boxes or components which are connected together in some way.

When describing a view, we are less interested in the contents of the instantiated views than in the interface of those instantiated views. EDIF supports this abstract approach by dividing a view into two parts: the interface and the contents. The interface describes an abstract representation of the view which may be visible when that view is instantiated within the contents of other views. One aspect of the interface is that it describes what the view looks like when viewed from the outside; that is, it describes its shape and size and also the points to which connections can be made. These connection points are called ports by EDIF and represent the only places where a connection can be made between the contents of a view and its external environment.

When a view is described, we must define an interface with some ports; otherwise we cannot connect to the view. We may also describe the view itself in the contents. When another view is instantiated within the contents we are describing, we may also be able to connect to that particular instance. We can think of the instance importing copies of the ports in its definition into the instancing view.

We can classify the ports of a view in two ways: ports defined in the interface, which we shall refer to as master ports, and ports which appear as part of the instances within the contents, which we shall refer to as instance ports. These two types of ports are illustrated in Figure 1; the ports are shown as shaded rectangles within the boundary of a view.

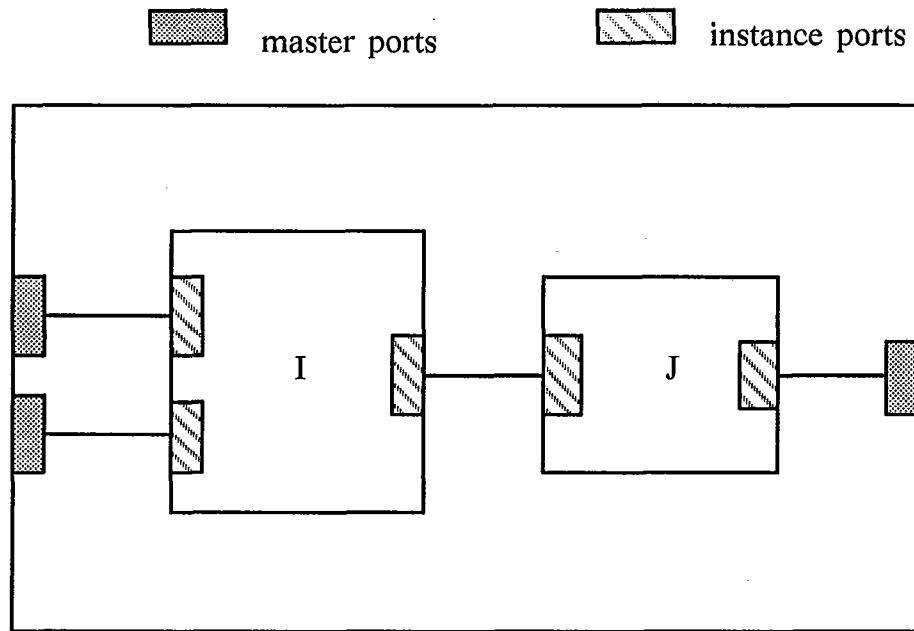


Figure 1. Master and instance ports.

For the connectivity of the circuit to be described we must have some mechanism to reference ports. This is achieved using the *portRef* or *globalPortRef* constructs. In the case of a master port we simply have to name the port within the *portRef*. For example, to refer to master port **MP** we would write:

(portRef MP)

In the case of an instance port, we must qualify the name with an instance name. For example:

(portRef A (instanceRef I))

means port **A** of the instance **I**.

On occasions it would also be useful to refer to all the ports with a particular name on all of the instances. EDIF provides a shorthand for this in the form of the *globalPortRef* construct. If we have two instances, **I1** and **I2**, each with a port **GP**, and another instance **I3** without this port **GP** then

(globalPortRef GP)

is equivalent to

(portRef GP)

(portRef GP (instanceRef I1))

(portRef GP (instanceRef I2))

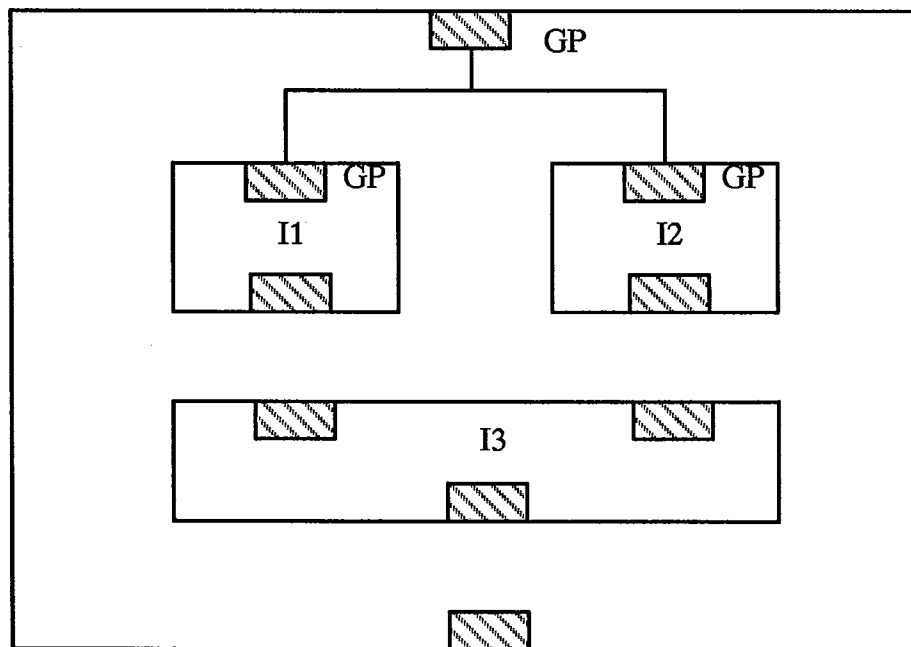


Figure 2. Global port example.

We list all instance ports **GP**. Instance **I3** has no port **GP**, and therefore it does not appear in this list. The reference to the master port **GP** is implied by the *globalPortRef*. *GlobalPortRef* does not imply connections across the hierarchy of a design; it has no implications beyond its use as a shorthand notation.

Within a view some of the ports could be connected or shorted together. If the circuit is considered to be a number of components wired together, there could be a piece of wire joining the connected ports. Since in an EDIF description of a circuit we are taking a hierarchical view, we only need to consider the description of these connections within the contents of one view. All the connected ports lie within a net. A net is the set of ports which are directly connected together, with associated attributes and possibly corresponding graphics. For example, in Figure 3 there are six nets, five of which include master ports and one, **net4**, which consists only of a pair of instance ports.

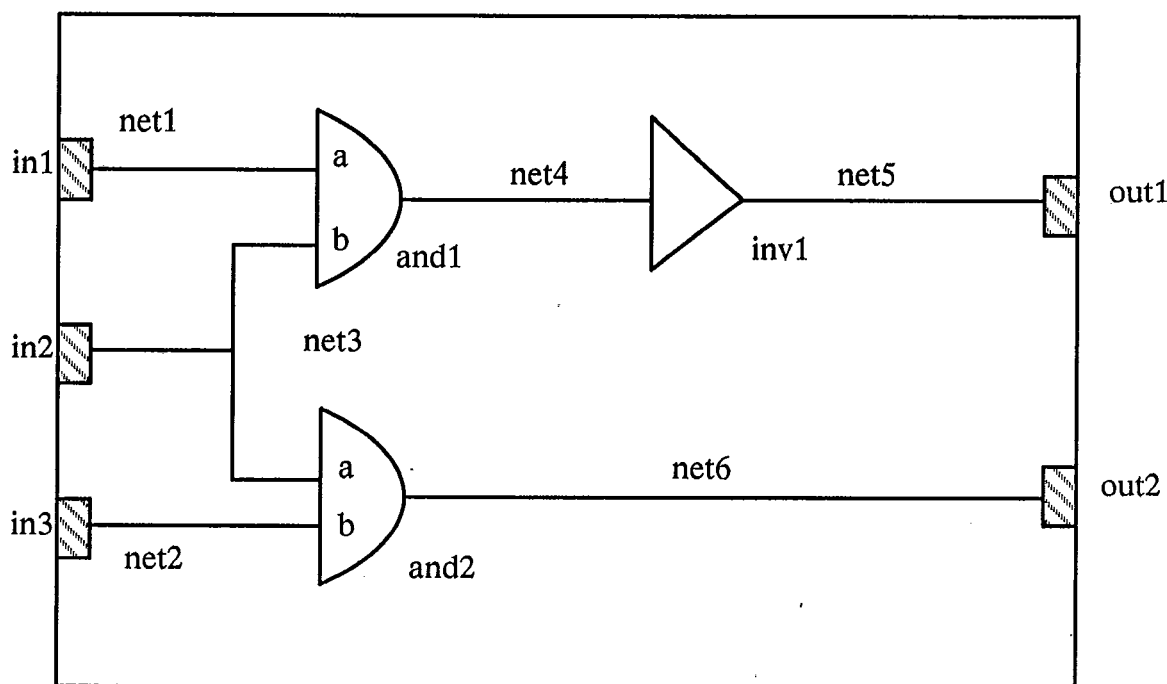


Figure 3. Simple connection example.

A simple net is described by listing its constituent ports. In general, however, there will often be other information associated with the net. The listing of the ports belonging to a net forms the basis of the EDIF description of connectivity. All aspects of connectivity are collected together in a *net* construct, which must contain one *joined* construct in which the ports are listed. In Figure 3, **net1** would be described as

```
(net net1
  (joined
    (portRef in1)
    (portRef a (instanceRef and1))))
```

and **net3** as

```
(net net3
  (joined
    (portRef in2)
    (portRef b (instanceRef and1))
    (portRef a (instanceRef and2))))
```