

**Marg Beal
Ken Trowell
Bill White**

FICON CTC Implementation

This Redpaper provides an overview of the features and benefits of FICON CTC, and the information necessary to implement the CTC function with FICON channels. Using the information in this Redpaper, a configuration planner and/or systems programmer can design and define a high availability CTC configuration in a FICON environment. Migration from ESCON CTCs to FICON CTCs is described, along with information on operations and performance of FICON CTC.

The following topics are discussed in this Redpaper:

- ▶ Review of CTC support
- ▶ ESCON CTC review
- ▶ Overview and benefits of FICON CTC
- ▶ FCTC topology options
- ▶ FCTC hardware and software support
- ▶ Recommendations for FCTC device numbering scheme
- ▶ FCTC configuration and definition examples
- ▶ FICON CTC control unit function balancing
- ▶ Migration considerations
- ▶ FCTC operations and recovery - configuring for high availability recommendations
- ▶ FCTC functional characteristics and performance

Review of CTC support

Channel-to-channel (CTC) control units and associated devices provide the infrastructure for intersystem communication. z/OS, OS/390 and z/VM exploiters of CTC communication include:

- ▶ XCF (Cross-System Coupling Facility) pathin and pathout devices for sysplex intersystem communication (z/OS and OS/390)
- ▶ VTAM read/write devices
- ▶ TCP/IP read/write devices
- ▶ IMS read/write devices

In the past, CTC control unit support has been provided by the parallel channel-attached 3088 control unit, and with the introduction of ESCON in the early 1990s, by the ESCON CTC channel. Now, the FICON channel on the z900 at driver level 3C or later provides enhanced CTC control unit support, with increased connectivity and more flexible channel usage.

To provide background information for the description of the FICON CTC enhancements, a brief review of existing channel-to-channel control unit support follows.

Parallel 3088 CTC

Prior to the availability of ESCON, the 3088 control unit provided parallel channel-attached CTC support. The 3088 control unit, defined as “CTC”, is no longer supported.

ESCON CTC

CTC support in the ESCON environment is provided with a pair of connected ESCON channels (either in a point-to-point or switched point-to-point configuration), where one ESCON channel is defined as TYPE=CNC and the other ESCON channel is defined as TYPE=CTC. The CTC control unit (CU) function is implemented by the microcode supporting the ESCON channel TYPE=CTC. The ESCON CTC and CNC channels can be connected point-to-point, or switched point-to-point through an ESCON Director, which provides the switching function. Both shared (EMIF) and non-shared ESCON channels support the ESCON CTC function.

ESCON CTC Review

In this section, the topology options available for implementing ESCON CTC, along with sample configurations and supporting definitions are reviewed to provide a basis for the comparison with FICON CTC.

ESCON CTC topologies

ESCON CTC communication requires a pair of ESCON channels where one channel is defined as an ESCON *CTC* channel and the other channel is defined as an ESCON *CNC* channel. The ESCON CTC Control Unit (CU) function is provided in the microcode of the ESCON CTC channel.

SCTC control units are defined on the ESCON CTC channel to communicate with *SCTC* control units defined on the ESCON CNC channel, which in turn are defined to communicate to the *SCTC* control units defined on the ESCON CTC channel. The control units are defined as type *SCTC* and they support ESCON CTC devices where the devices are defined as type *SCTC*.

Important: For some older exploiters, *BCTC* devices were supported by ESCON SCTC control units. The FICON CTC CU function does *not* support the BCTC definition for FICON CTC devices.

Note the following:

- ▶ The ESCON channel defined as CTC can only support the ESCON CTC CU function and CTC devices. It cannot be used for other I/O device support, such as disk, tape and so on.
- ▶ The ESCON channel defined as CNC can support ESCON CTC control unit definitions and other I/O control unit type definitions, such as disk and tape, if the channel is connected in a *switched* point-to-point topology.

The ESCON CTC channel supports a maximum of 512 unit addresses (or devices) and up to 120 logical control units. This limitation means that multiple pairs of CTC and CNC channels must be used in an installation with a large number of interconnected LPARs on the same physical processor or to LPARS (images) on other processors.

Point-to-point

ESCON CTC and CNC channels can be connected point-to-point, as shown in Figure 1 and Figure 2.

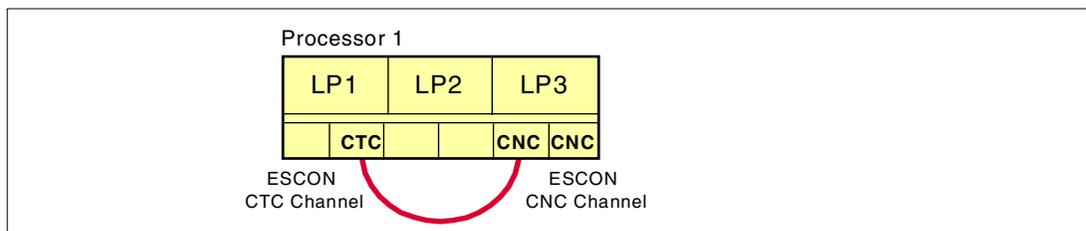


Figure 1 ESCON point-to-point CTC, single processor

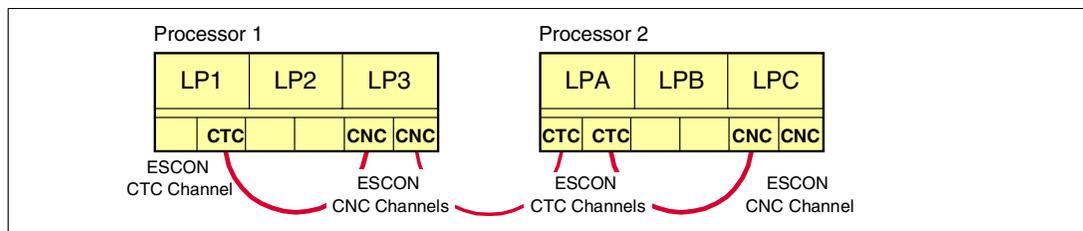


Figure 2 ESCON point-to-point CTC connection, multiple processors

In a point-to-point configuration, the ESCON CTC-CNC channels (either on the same or different processors) are directly connected to one another; neither channel is connected to an ESCON Director. A CNC channel connected point-to-point to a CTC channel can only support CTC control units. It cannot be used, for example, to support disk or tape control units. For this reason, point-to-point ESCON CTC-CNC configurations are not commonly used.

Switched point-to-point

ESCON CTC and CNC channels, on one or more processors, are usually connected in a switched point-to-point configuration, where the pairs of CTC-CNC channels are connected to the same ESCON Director. A simple ESCON switched point-to-point connection is shown in Figure 3 on page 4.

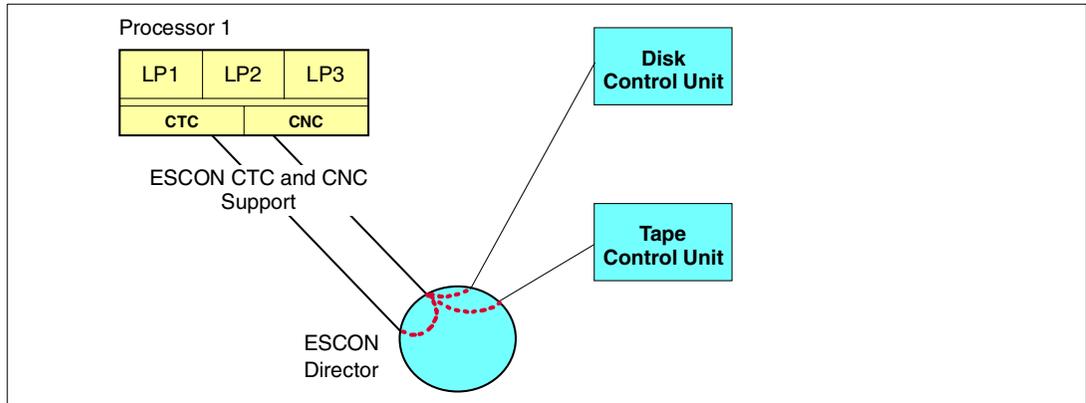


Figure 3 ESCON switched point-to-point CTC connections - one processor

An ESCON CTC and CNC channel pair is needed for communicating between images (LPARs) on the same processor. With the CNC channel connected to an ESCON Director, it can also be used to communicate with other ESCON control unit types, such as disk and tape control units.

A more common example of a switched point-to-point CTC configuration between systems on one processor and systems on a second processor is shown in Figure 4.

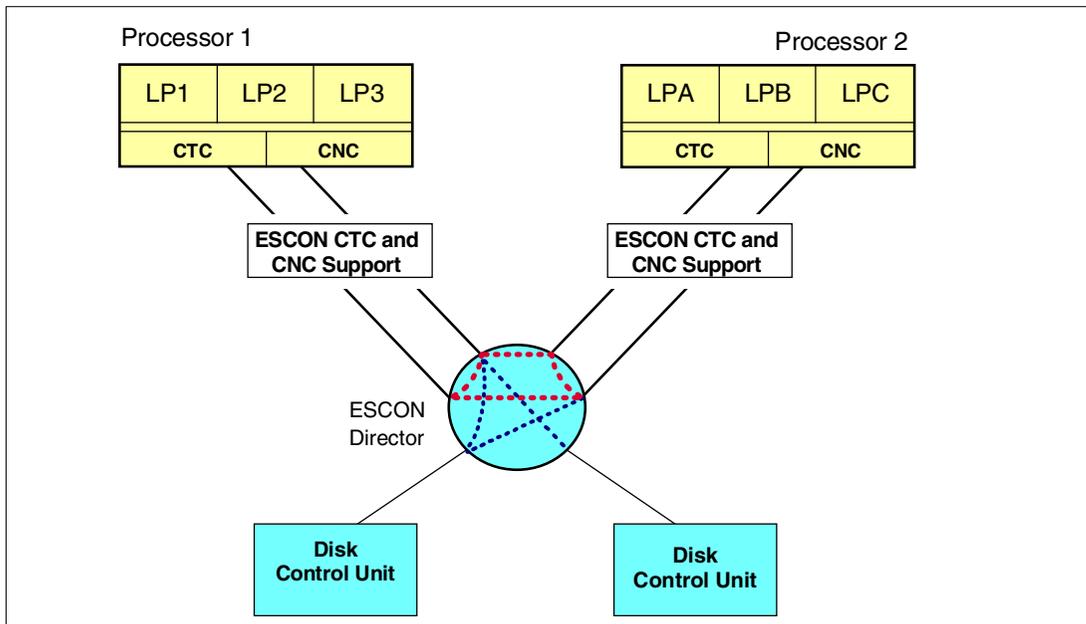


Figure 4 ESCON switched point-to-point connections

This figure, and the previous figures, show the minimum or *prime* CTC configuration. An ESCON CTC and CNC channel pair is needed on each processor for communicating between systems (logical partitions) on the same processor, or for communicating to logical partitions (LPARs) on other physical processors. A duplicated configuration (not shown) is required for availability reasons and should be connected through different ESCON channels and a different ESCON Director.

Sample ESCON CTC configurations and definitions

Defining ESCON or FICON CTC configurations requires an understanding of the use of CTCs by the different subsystems. For example, VTAM and TCP/IP require multiple read/write CTC device pairs, XCF requires multiple pathin/pathout CTC device pairs, and so on.

CTC hardware support allows the same CTC device to be used for both sending and receiving data, but most subsystems that use CTC devices for inter-system communications use a pair of devices: one for sending and the other for receiving. As a result, there could be up to eight CTC devices required for one image to communicate with one other image in the complex.

These eight CTC devices represent just the primary communication paths which should be duplicated for availability reasons. So, in a complex of 16 fully-interconnected images, up to 3,480 CTC devices may be required. This number is derived from the following calculation:

$$(((16 \text{ images}-1) \times 8 \text{ devices}) \times 16 \text{ images}) \times 2 \text{ for availability}) = 3840 \text{ CTC devices}$$

ESCON CTC device numbering scheme

Such a large number of CTC devices means that it can be both difficult and complex to design and define the CTC connections for a large CTC configuration. To help minimize the complexity, IBM developed a CTC device numbering scheme for use with ESCON CTC devices at the time when ESCON was introduced in the early 1990s. This device numbering scheme is documented in *MVS/ESA HCD and Dynamic I/O Reconfiguration Primer*, SG24-4037.

The method makes use of the 4-digit z/Architecture device number where:

- ▶ First digit
 - an even hexadecimal number is used for the send CTC control unit and device
 - an odd hexadecimal number is used for the receive CTC control unit and device
- ▶ Second and third digits

These represent an assigned CTC image-ID (that is assigned on paper) for the LPAR image and can be any unique value within the CTC complex. This CTC image-ID is used as a target identifier (for the image) that you use when defining how all other images access this image.
- ▶ Fourth digit

This digit is used to indicate whether the CTC connection is the primary or alternate (backup for availability purposes). A value of 0 to 7 is used for the primary CTC connection devices, and a value of 8 to F is used for the alternate CTC connection devices.

There are many advantages to using this device numbering scheme. For instance, operations and systems programmers can easily identify the use (send or receive) and target system for any given CTC device. It also simplifies the CTC definition process in that the same operating system CTC device definitions can be used by all z/OS and OS/390 operating systems, independent of the image and processor where that operating system runs.

Because this numbering scheme is also being recommended for FICON CTC configurations, a sample configuration and definition is discussed. Figure 5 on page 6 shows the sample ESCON CTC configuration.

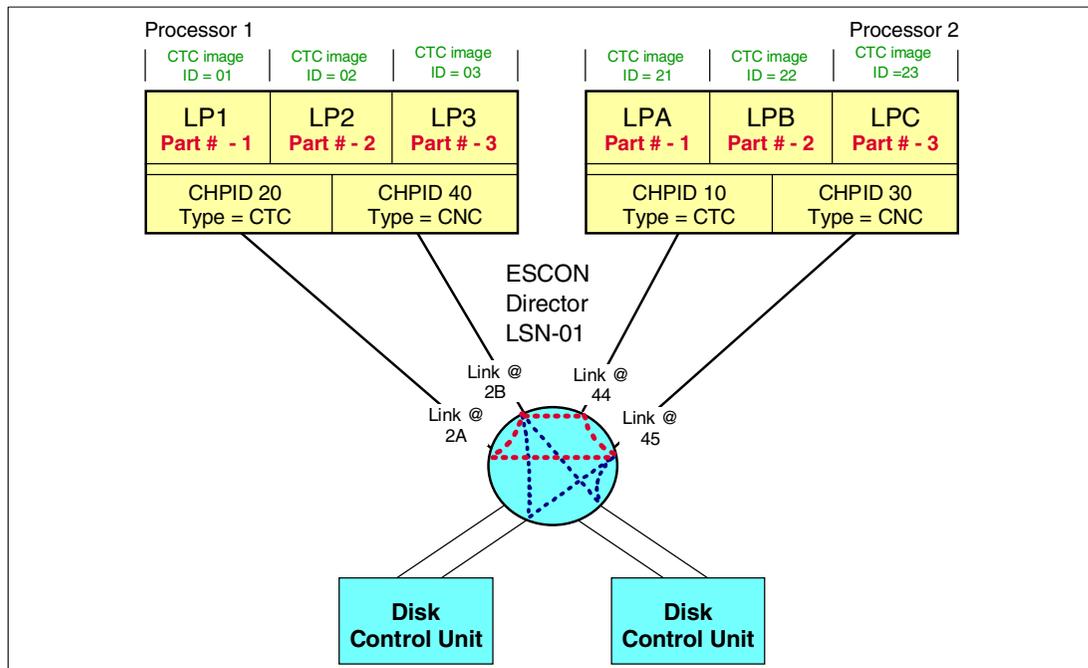


Figure 5 Sample ESCON CTC configuration

Two processors, each with three logical partitions, have full intersystem connectivity provided by an ESCON CTC and CNC channel on each processor. The channels are connected through an ESCON Director. Only the prime CTC configuration is shown.

The device number design for this configuration is shown in Figure 6. Note that the disk control units can be accessed only from the ESCON CNC channels. The ESCON CTC channels cannot be used to access normal I/O devices.

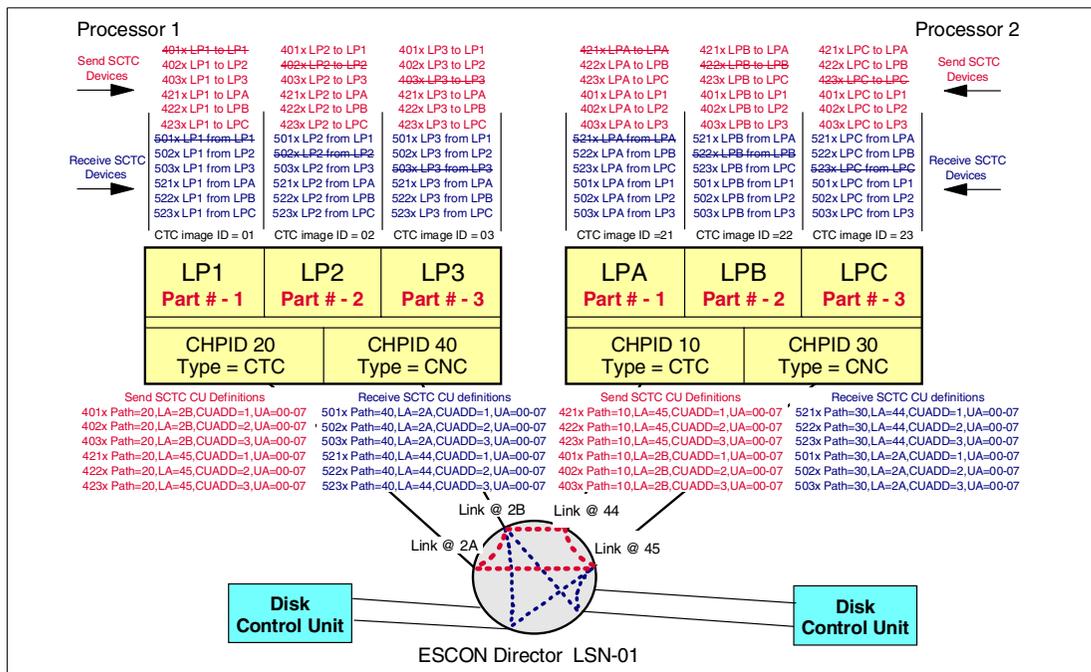


Figure 6 Sample ESCON CTC configuration CTC definitions

The diagram shows that all the send control units and devices are defined with numbers of the form 4xxx, and all the receive control units and devices are defined with numbers of the form 5xxx.

Specifically:

- ▶ Device number 4230 is used by LP1 on processor 1 to communicate as a send CTC device with logical partition LPC on processor 2.
- ▶ LPC has been assigned CTC image-ID 23, which is a unique CTC image-ID across the whole CTC complex.
- ▶ The CTC image-ID, here 23, forms the second and third digits of the device number.
- ▶ LPC has been defined with an LPAR partition number of 3, but the defined partition number is only unique on the processor and is therefore not used as part of the external device numbering scheme. It is used in the configuration definitions for the CTC CU - CUADD=x.

Using the same device numbering method, device number 5010 is used by LPC on processor 2 to communicate (as a receive CTC device) with logical partition LP1 on processor 1. LP1 on processor 1 has been assigned CTC image-ID 01, and here forms the second and third digits of the device number.

However, the choice of device numbers is not sufficient to ensure working CTC connections. The CTC control unit and device definitions must follow the z/Architecture and S/390 Architecture channel-to-device incoming request mapping rules. That is, the incoming source identifier is mapped to a defined destination identifier and unit address. This is explained in more detail in the ESCON SCTC-to-SCTC mapping from LP1 on processor 1 to LPC on processor 2, shown in Figure 7.

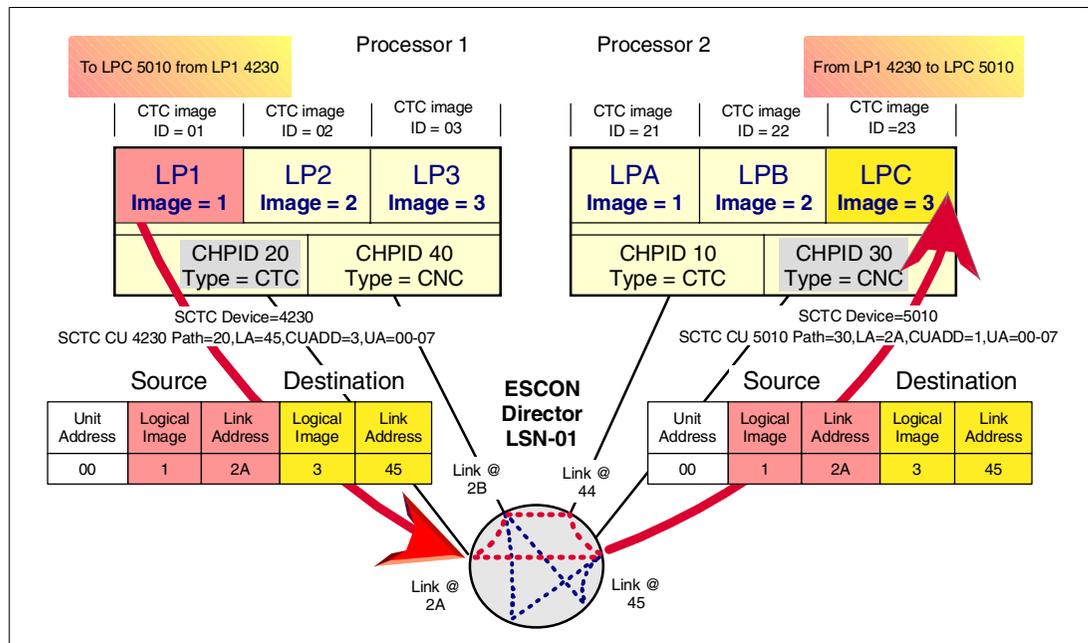


Figure 7 ESCON SCTL to SCTL mapping LP1 processor 1 to LPC processor 2

As shown in Figure 7:

- ▶ Device number 4230 is defined on processor 1 on SCTL control unit number 4230 with unit address 00. CU 4230 is defined to use CTC CHPID 20, link address (DLA) 45 and CUADD (destination control unit image) 3. On processor 2, SCTL control unit 5010 is

defined to use CHPID 30, which is connected to link address 45. LPC, which is in the access list for CHPID 30, has LPAR partition number (image number) of 3.

- ▶ Device number 5010 is defined on processor 3 SCTC control unit number 5010 with unit address 00. CU 5010 is defined to use CTC CHPID 30, link address (DLA) 2A and CUADD (destination control unit image) 1. On processor 1, SCTC control unit 4230 is defined to use CHPID 20, which is connected to link address 2A. LP1, which is in the access list for CHPID 20, has LPAR partition number (image number) of 1.

So, the incoming source ID is mapped to a defined destination ID consisting of destination link address (DLA), destination control unit image and unit address.

The ESCON SCTC-to-SCTC mapping from LPC on processor 2 to LP1 on processor 1 is shown in Figure 8. In this example, device 4010 in LPC on processor 2 has been defined to provide a source identifier matching device 5230 in LP1 on processor 1.

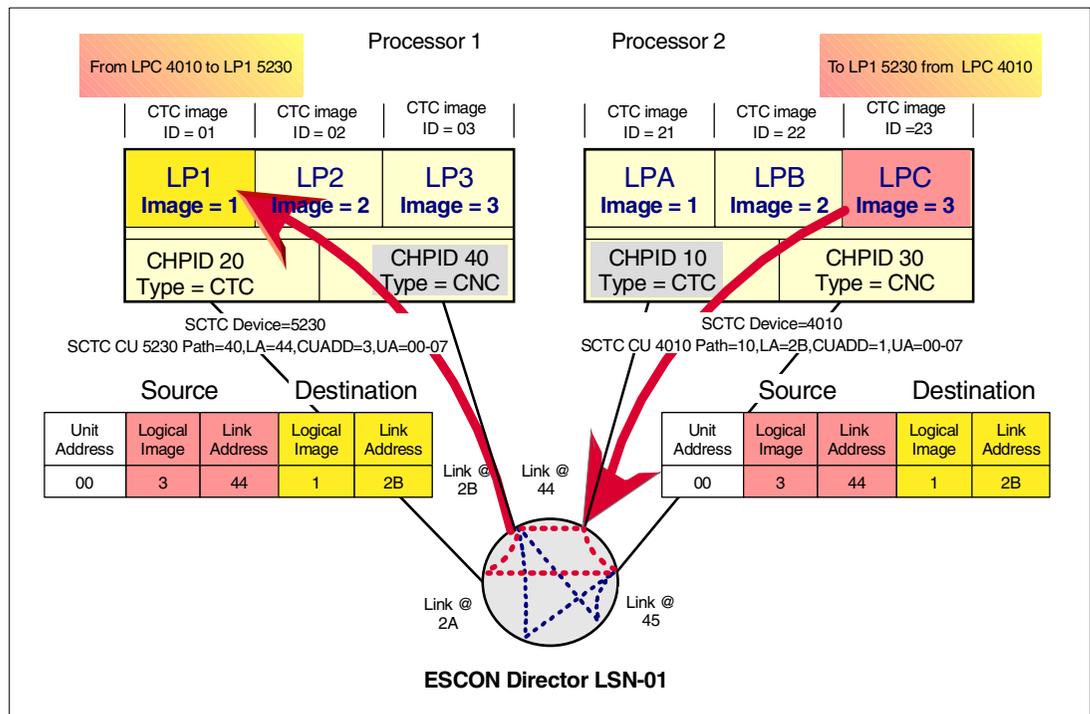


Figure 8 ESCON SCTC to SCTC mapping LPC processor 2 to LP1 processor 1

The definitions for this configuration are shown in Figure 9 on page 9 for processor 1, and in Figure 10 on page 10 for processor 2.

```

ID      MSG1='current escon 4xxx and 5xxx CTC Image_ID definition method',
        MSG2='escon does not support full duplex communication',
        SYSTEM=(2064,1)
RESOURCE PARTITION=((LP1,1),(LP2,2),(LP3,3))
CHPID  PATH=(20),SHARED,PARTITION=((LP1,LP2,LP3),(LP1,LP2,LP3)),SWITCH=01,TYPE=CTC
CHPID  PATH=(40),SHARED,PARTITION=((LP1,LP2,LP3),(LP1,LP2,LP3)),SWITCH=01,TYPE=CNC
CHPID  PATH=(70),SHARED,PARTITION=((LP1,LP2,LP3),(LP1,LP2,LP3)),SWITCH=02,TYPE=CTC
CHPID  PATH=(A0),SHARED,PARTITION=((LP1,LP2,LP3),(LP1,LP2,LP3)),SWITCH=02,TYPE=CNC
CNTLUNIT CUNUMBR=4010,PATH=(20),UNITADD=((00,008)),LINK=(2B),
        CUADD=1,UNIT=SCTC
CNTLUNIT CUNUMBR=4020,PATH=(20),UNITADD=((00,008)),LINK=(2B),
        CUADD=2,UNIT=SCTC
CNTLUNIT CUNUMBR=4030,PATH=(20),UNITADD=((00,008)),LINK=(2B),
        CUADD=3,UNIT=SCTC
CNTLUNIT CUNUMBR=4210,PATH=(20),UNITADD=((00,008)),LINK=(45),
        CUADD=1,UNIT=SCTC
CNTLUNIT CUNUMBR=4220,PATH=(20),UNITADD=((00,008)),LINK=(45),
        CUADD=2,UNIT=SCTC
CNTLUNIT CUNUMBR=4230,PATH=(20),UNITADD=((00,008)),LINK=(45),
        CUADD=3,UNIT=SCTC
CNTLUNIT CUNUMBR=5010,PATH=(40),UNITADD=((00,008)),LINK=(2A),
        CUADD=1,UNIT=SCTC
CNTLUNIT CUNUMBR=5020,PATH=(40),UNITADD=((00,008)),LINK=(2A),
        CUADD=2,UNIT=SCTC
CNTLUNIT CUNUMBR=5030,PATH=(40),UNITADD=((00,008)),LINK=(2A),
        CUADD=3,UNIT=SCTC
CNTLUNIT CUNUMBR=5210,PATH=(40),UNITADD=((00,008)),LINK=(44),
        CUADD=1,UNIT=SCTC
CNTLUNIT CUNUMBR=5220,PATH=(40),UNITADD=((00,008)),LINK=(44),
        CUADD=2,UNIT=SCTC
CNTLUNIT CUNUMBR=5230,PATH=(40),UNITADD=((00,008)),LINK=(44),
        CUADD=3,UNIT=SCTC
IODEVICE ADDRESS=(4010,008),UNITADD=00,CUNUMBR=(4010),STADET=Y,
        PARTITION=(LP2,LP3),UNIT=SCTC
IODEVICE ADDRESS=(4020,008),UNITADD=00,CUNUMBR=(4020),STADET=Y,
        PARTITION=(LP1,LP3),UNIT=SCTC
IODEVICE ADDRESS=(4030,008),UNITADD=00,CUNUMBR=(4030),STADET=Y,
        PARTITION=(LP1,LP2),UNIT=SCTC
IODEVICE ADDRESS=(5010,008),UNITADD=00,CUNUMBR=(5010),STADET=Y,
        PARTITION=(LP2,LP3),UNIT=SCTC
IODEVICE ADDRESS=(5020,008),UNITADD=00,CUNUMBR=(5020),STADET=Y,
        PARTITION=(LP1,LP3),UNIT=SCTC
IODEVICE ADDRESS=(5030,008),UNITADD=00,CUNUMBR=(5030),STADET=Y,
        PARTITION=(LP1,LP2),UNIT=SCTC
IODEVICE ADDRESS=(4210,008),UNITADD=00,CUNUMBR=(4210),STADET=Y,
        PARTITION=(LP1,LP2,LP3),UNIT=SCTC
IODEVICE ADDRESS=(4220,008),UNITADD=00,CUNUMBR=(4220),STADET=Y,
        PARTITION=(LP1,LP2,LP3),UNIT=SCTC
IODEVICE ADDRESS=(4230,008),UNITADD=00,CUNUMBR=(4230),STADET=Y,
        PARTITION=(LP1,LP2,LP3),UNIT=SCTC
IODEVICE ADDRESS=(5210,008),UNITADD=00,CUNUMBR=(5210),STADET=Y,
        PARTITION=(LP1,LP2,LP3),UNIT=SCTC
IODEVICE ADDRESS=(5220,008),UNITADD=00,CUNUMBR=(5220),STADET=Y,
        PARTITION=(LP1,LP2,LP3),UNIT=SCTC
IODEVICE ADDRESS=(5230,008),UNITADD=00,CUNUMBR=(5230),STADET=Y,
        PARTITION=(LP1,LP2,LP3),UNIT=SCTC

```

Figure 9 ESCON CTC channel, CU and device definitions for processor 1

Note that the SCTC CU starting unit address range for each control unit is defined to start at x'00'. The I/O devices are also defined with the starting unit address (for the range) of x'00'. This is a key point for this definition method. The device unit addresses are defined as starting at x'00' so that devices with different device numbers can be matched. For example, device number 4230 on processor 1 is defined with unit address x'00' and matches device number 5010 on processor 2, which is defined with a matching unit address of x'00'.

```

ID      MSG1='current escon 4xxx and 5xxx CTC Image_ID definition method',
        MSG2='escon does not support full duplex communication',
        SYSTEM=(2064,1)
RESOURCE PARTITION=( (LPA,1), (LPB,2), (LPC,3) )
CHPID  PATH=(10), SHARED, PARTITION=( (LPA,LPB,LPC), (LPA,LPB,LPC) ), SWITCH=01, TYPE=CNC
CHPID  PATH=(20), SHARED, PARTITION=( (LPA,LPB,LPC), (LPA,LPB,LPC) ), SWITCH=01, TYPE=CNC
CHPID  PATH=(54), SHARED, PARTITION=( (LPA,LPB,LPC), (LPA,LPB,LPC) ), SWITCH=02, TYPE=CNC
CHPID  PATH=(55), SHARED, PARTITION=( (LPA,LPB,LPC), (LPA,LPB,LPC) ), SWITCH=02, TYPE=CNC
CNTLUNIT CUNUMBR=4010, PATH=(10), UNITADD=((00,008)), LINK=(2B),
        CUADD=1, UNIT=SCTC
CNTLUNIT CUNUMBR=4020, PATH=(10), UNITADD=((00,008)), LINK=(2B),
        CUADD=2, UNIT=SCTC
CNTLUNIT CUNUMBR=4030, PATH=(10), UNITADD=((00,008)), LINK=(2B),
        CUADD=3, UNIT=SCTC
CNTLUNIT CUNUMBR=4210, PATH=(10), UNITADD=((00,008)), LINK=(45),
        CUADD=1, UNIT=SCTC
CNTLUNIT CUNUMBR=4220, PATH=(10), UNITADD=((00,008)), LINK=(45),
        CUADD=2, UNIT=SCTC
CNTLUNIT CUNUMBR=4230, PATH=(10), UNITADD=((00,008)), LINK=(45),
        CUADD=3, UNIT=SCTC
CNTLUNIT CUNUMBR=5010, PATH=(30), UNITADD=((00,008)), LINK=(2A),
        CUADD=1, UNIT=SCTC
CNTLUNIT CUNUMBR=5020, PATH=(30), UNITADD=((00,008)), LINK=(2A),
        CUADD=2, UNIT=SCTC
CNTLUNIT CUNUMBR=5030, PATH=(30), UNITADD=((00,008)), LINK=(2A),
        CUADD=3, UNIT=SCTC
CNTLUNIT CUNUMBR=5210, PATH=(30), UNITADD=((00,008)), LINK=(44),
        CUADD=1, UNIT=SCTC
CNTLUNIT CUNUMBR=5220, PATH=(30), UNITADD=((00,008)), LINK=(44),
        CUADD=2, UNIT=SCTC
CNTLUNIT CUNUMBR=5230, PATH=(30), UNITADD=((00,008)), LINK=(44),
        CUADD=3, UNIT=SCTC
IODEVICE ADDRESS=(4010,008), UNITADD=00, CUNUMBR=(4010), STADET=Y,
        PARTITION=(LPA,LPB,LPC), UNIT=SCTC
IODEVICE ADDRESS=(4020,008), UNITADD=00, CUNUMBR=(4020), STADET=Y,
        PARTITION=(LPA,LPB,LPC), UNIT=SCTC
IODEVICE ADDRESS=(4030,008), UNITADD=00, CUNUMBR=(4030), STADET=Y,
        PARTITION=(LPA,LPB,LPC), UNIT=SCTC
IODEVICE ADDRESS=(5010,008), UNITADD=00, CUNUMBR=(5010), STADET=Y,
        PARTITION=(LPA,LPB,LPC), UNIT=SCTC
IODEVICE ADDRESS=(5020,008), UNITADD=00, CUNUMBR=(5020), STADET=Y,
        PARTITION=(LPA,LPB,LPC), UNIT=SCTC
IODEVICE ADDRESS=(5030,008), UNITADD=00, CUNUMBR=(5030), STADET=Y,
        PARTITION=(LPA,LPB,LPC), UNIT=SCTC
IODEVICE ADDRESS=(4210,008), UNITADD=00, CUNUMBR=(4210), STADET=Y,
        PARTITION=(LPB,LPC), UNIT=SCTC
IODEVICE ADDRESS=(4220,008), UNITADD=00, CUNUMBR=(4220), STADET=Y,
        PARTITION=(LPA,LPC), UNIT=SCTC
IODEVICE ADDRESS=(4230,008), UNITADD=00, CUNUMBR=(4230), STADET=Y,
        PARTITION=(LPA,LPB), UNIT=SCTC
IODEVICE ADDRESS=(5210,008), UNITADD=00, CUNUMBR=(5210), STADET=Y,
        PARTITION=(LPB,LPC), UNIT=SCTC
IODEVICE ADDRESS=(5220,008), UNITADD=00, CUNUMBR=(5220), STADET=Y,
        PARTITION=(LPA,LPC), UNIT=SCTC
IODEVICE ADDRESS=(5230,008), UNITADD=00, CUNUMBR=(5230), STADET=Y,
        PARTITION=(LPA,LPB), UNIT=SCTC

```

Figure 10 ESCON CTC channel, CU and device definitions for processor 2

Overview and benefits of FICON CTC

In this section, FICON CTC is introduced along with a description of the benefits of implementing the CTC function using FICON channels.

FICON CTC

Channel-to-channel communication in a FICON environment is provided between two FICON (FC) channel FCTC control units, with at least one of the two FCTC CUs being defined on an FC channel on a z900 processor at driver level 3C or later, and the other defined on an FICON (FC) channel on any of the following processors:

- ▶ 9672 G5/G6 CPC
- ▶ z900 CPC
- ▶ z900 CPC at driver level 3C or later

The FICON CTC CU function is provided by the FICON (FC) channel FCTC control unit on the z900 CPC only at z900 hardware driver level 3C or later levels.

Differences between ESCON and FICON CTC

There are several differences between the ESCON and FICON CTC implementations, as shown in Table 1.

Table 1 ESCON and FICON CTC differences

Characteristic	ESCON	FICON
# of required channels	At least 2	1 or 2
Channel dedicated to CTC function	Yes	No
# of unit addresses supported	Up to 512	Up to 16384
Data transfer bandwidth	12 - 17 MBps	60 - 90+ MBps
# of concurrent I/O operations	1	up to 32
Data transfer mode	Half duplex	Full duplex

The details of these differences are as follows:

- ▶ ESCON CTC connectivity is provided by a pair of ESCON channels, one defined as CTC and the other defined as CNC. At least two ESCON channels are required.
FICON CTC connectivity can be implemented using one or two FICON (FC) native channels.
- ▶ An ESCON channel defined as CTC can only support the CTC function. Only an SCTC control unit can be defined on an ESCON CTC channel.
The FICON native (FC) channel supporting the FCTC control unit can communicate with an FCTC control unit on either the z900 or 9672 G5/G6, and at the same time the same FICON (FC) channel can also support operations to other I/O control unit types such as disk and tape.
- ▶ An ESCON CTC channel supports a maximum of 512 unit addresses (devices).
A FICON native (FC) channel supports a maximum of 16,384 unit addresses (devices).
- ▶ An ESCON channel has a data transfer bandwidth of 12 - 17 MB, significantly less than the 60 - 90 + MB of a FICON channel.
- ▶ An ESCON channel supports only one actively communicating I/O operation at a time, whereas the FICON channel supports up to 32 concurrent I/O operations.
- ▶ An ESCON channel operates in half duplex mode, transferring data in one direction only at any given time. A FICON channel operates in full duplex mode, sending and receiving data at the same time.

FCTC topology options

A FICON CTC configuration can be implemented in different ways, as discussed in more detail later in this section. However, the following considerations apply to all FICON CTC configurations:

- ▶ The processor at each end of a FICON CTC connection uses a FICON native (FC) channel.

- The FICON channel is defined as “FC” (FICON native).
- ▶ The FC channel at each end of the CTC connection has a FICON CTC control unit defined. An FCTC control unit can be defined on a FICON native (FC) channel on a 9672 G5, 9672 G6 or z900, but the FCTC control unit function is provided only by a z900 at driver level 3C or later.
 - The FICON CTC control unit is defined as “FCTC”.
 - The FICON CTC devices on the FCTC control unit are defined as “FCTC”.
- ▶ The FCTC control function on the z900 at driver level 3C or later can communicate with an FCTC control unit defined on a FICON native (FC) channel on any of the following:
 - 9672 G5 or G6
 - z900
 - z900 at driver level 3C or later
- ▶ The FICON channel at each end of the FICON CTC connection, supporting the FCTC control units, can also communicate with other FICON native control units, such as disk and tape.

The diagram in Figure 11 summarizes the different CTC connectivity options available in the zSeries and 9672 G5/G6 environment.

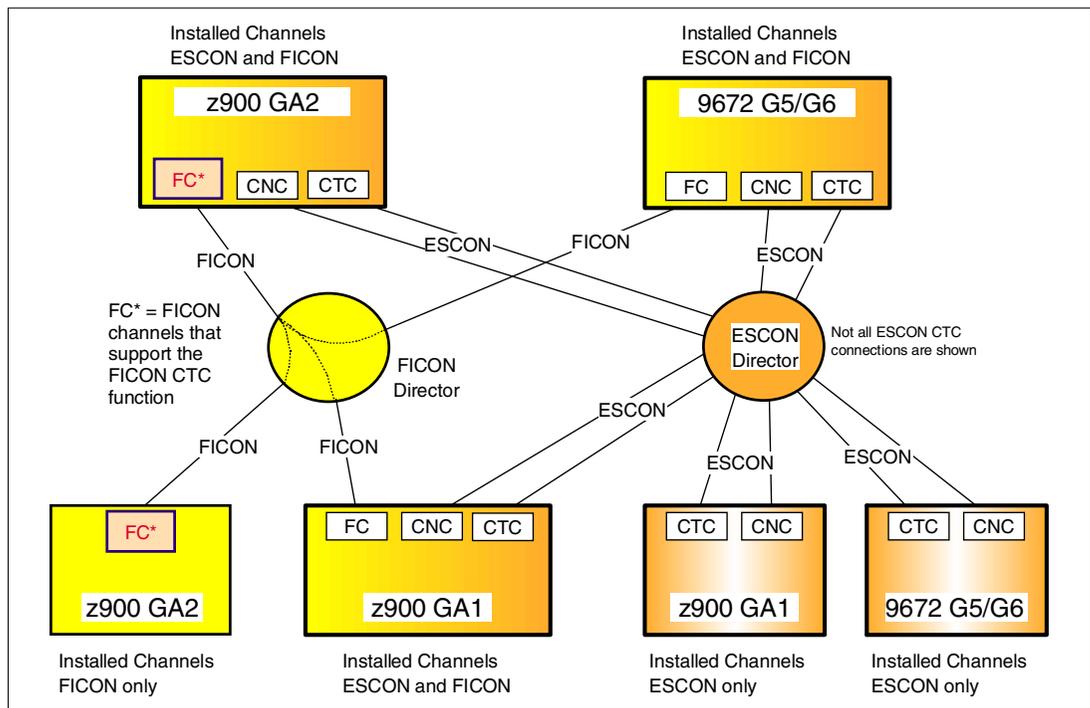


Figure 11 ESCON and FICON CTC connectivity options

In a FICON CTC configuration, although FICON CTC control units are defined at each end, only one end provides the FICON CTC control unit function. During initialization of the logical connection between two ends of a FICON CTC connection, the channel that will provide the FICON CTC control unit function is determined using an algorithm that, where possible, results in balancing of the number of FCTC CU functions that each end of the logical connection is providing. The establishment algorithm is discussed in more detail in “FICON CTC control unit function balancing” on page 24.

The FICON (FC mode) channel CTC configuration can use:

- ▶ One FICON (FC mode) channel in a point-to-point, or switched point-to-point configuration
- ▶ Two FICON (FC mode) channels in a point-to-point, or switched point-to-point configuration

CTC communication using one FICON channel per processor

Unlike the ESCON channel CTC communication, which uses a pair of ESCON CTC-CNC channels, the FICON (FC mode) channel CTC communication does *not* require a pair of channels because it can communicate with any FICON (FC mode) channel that has a corresponding FCTC control unit defined. This means that FICON CTC communications can be provided using only a single FICON (FC mode) channel per processor.

Single FICON FCTC CU function channel on one processor

A single FICON channel connected to a FICON Director can provide the FICON CTC communications between the systems in the logical partitions on a single processor, as well as images on other processors. It can also be used to communicate to other I/O control units. A sample configuration is shown in Figure 12.

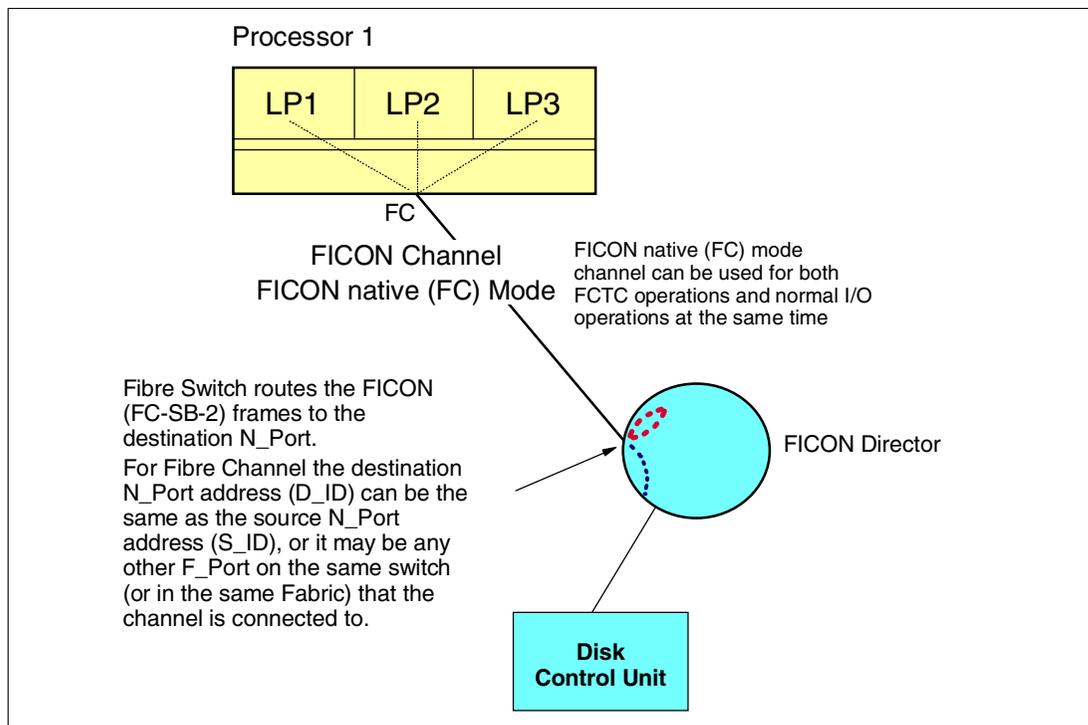


Figure 12 Single FICON channel on one processor

Single FICON channel between two or more processors

A single native FICON (FC) channel with FICON control units defined can be used to communicate between images (LPARs) on the same processor as well as images on other processors. It can also be used to communicate to other I/O control units. The FC channels are connected to a FICON Director. A sample configuration is shown in Figure 13 on page 14.

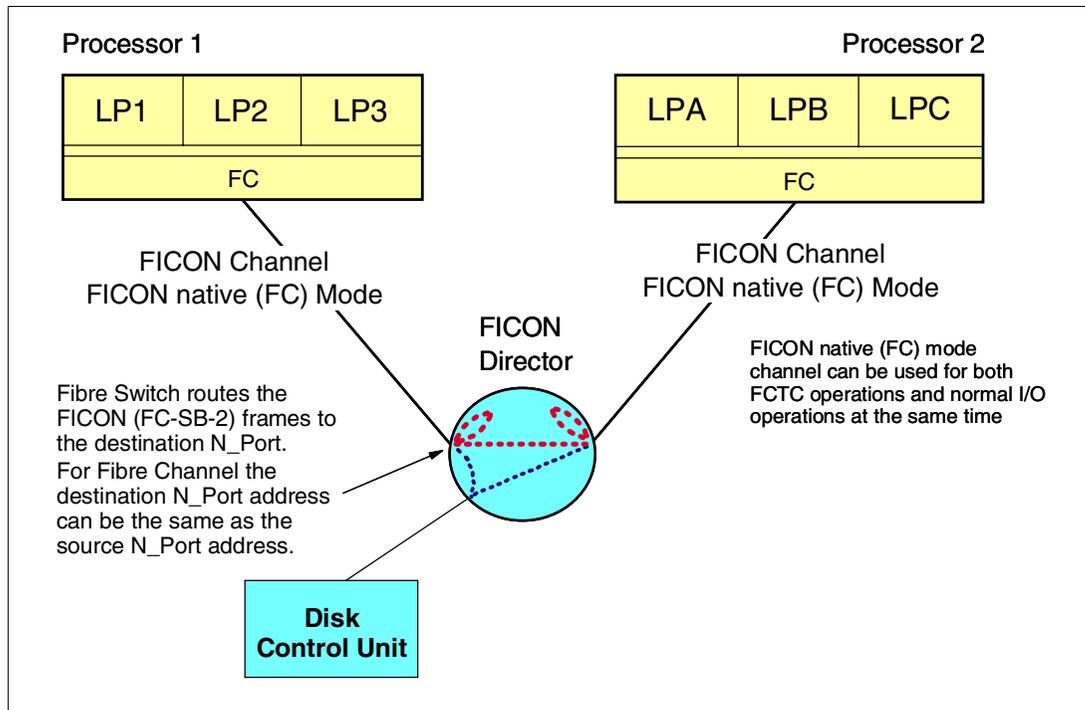


Figure 13 Single FICON channel per processor

CTC communication using two FICON channels per processor

Although a single FICON native (FC) channel per processor can provide CTC connections across multiple processors, for large FICON configurations, we recommend using at least one pair of FICON native (FC) mode channels. Using a pair of FICON native (FC) mode channels allows the installation to maintain the same CTC device definition methodology for FICON as was previously used for ESCON. But the FICON channels can support the definition of FCTC control units as well as other I/O control units, such as disk and tape, at the same time.

Two FICON channels on one processor

A sample configuration with two FICON (FC) mode channels providing FCTC-to-FCTC communications is shown in Figure 14 on page 15. A FICON (FC) channel supports up to 255 LCUs and up to 16K unit addresses per channel. Since the FICON native (FC) channel supports a larger number of devices, installations with a high number of logical partitions in an FCTC complex are easier to design and define.

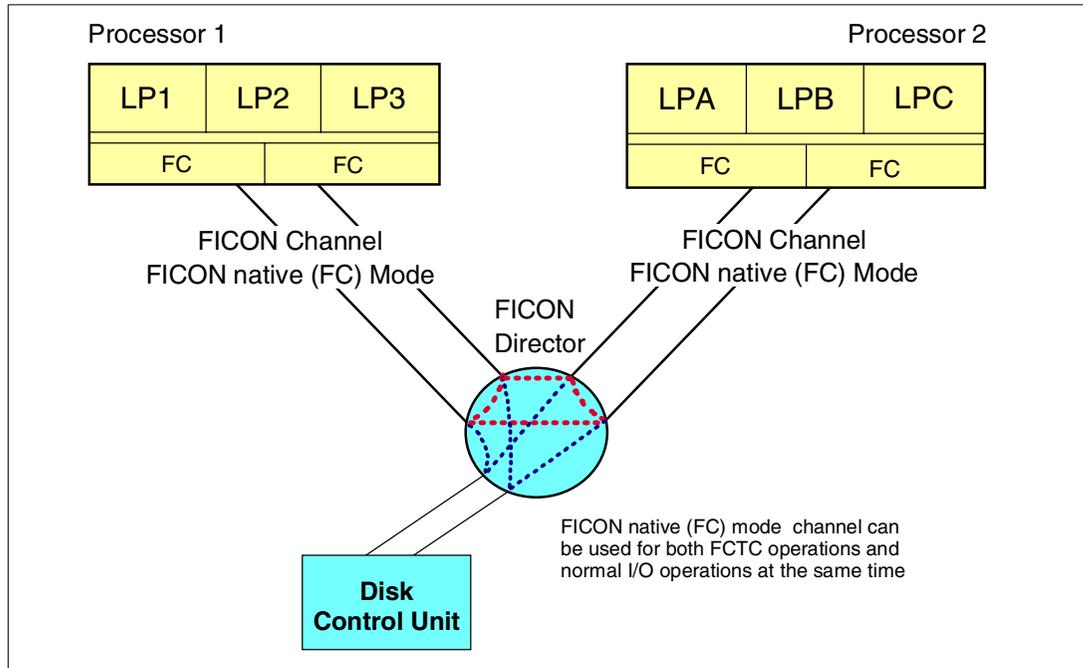


Figure 14 Two FICON channels between two processors

FCTC hardware and software support

Hardware support

The FICON native (FC) channel provides FICON CTC (FCTC) support. FICON CTC connections require that at least one end of the connection be on a z900 CPC at driver level 3C or later. The other end of the FICON CTC connection can be on any processor that supports FICON native (FC) channel paths, including:

- ▶ 9672 G5/G6 FICON channel defined with a FCTC control unit
- ▶ z900 GA1 FICON channel defined with a FCTC control unit
- ▶ z900 driver level 3C or later FICON channel defined with a FCTC control unit

Software support

IOS

IOS provides the UIM required for FCTC definition and initialization support in APAR OW48283.

HCD

HCD support for the definition of FCTC control units and devices is provided in APAR OW45976. This APAR is applicable to FMIDs HCS6051 and HCS6091, and the PTFs that implement this APAR are listed in Table 2 on page 16.

Table 2 HCD FCTC APAR OW45976 PTFs

HCD Release	PTF
R051	UW99391
R053	UW99392
R054	UW99393
R091	UW99394

HCM

HCM APAR IR45358 provides support for FCTC definitions. The PTFs that implement this APAR are listed in Table 3.

Table 3 HCM FCTC APAR IR45358 PTFs

HCM Release	PTF
R210	UR90332
R310	UR90333
R410	UR90334

IOCP

IYP IOCP Version 1 Release 1 Level 1 (1.1.1) provides support for FICON channel-to-channel (FCTC) communication. Note that one end of the FICON CTC connection must be on a CPC at EC J10638 for the connection to be successfully established.

There is a new release of the publication *zSeries 900 Input/Output Configuration Program User's Guide for IYP IOCP*, SB10-7029-01. The publication is available using Resource Link at <http://www.ibm.com/servers/resourcelink> by selecting section: **Library > zSeries 900 > Hardware**.

Information on equivalent standalone IOCP, OS/390 IOCP, VM IOCP, and VSE IOCP follows:

- ▶ Equivalent standalone IOCP for IYP IOCP version 1.1.1 is in EC J10638 for machine type 2064 (1xx models).
- ▶ Equivalent z/OS, OS/390, VM, and VSE IOCP is shipped via the APARs listed in Table 4.

Table 4 IOCP APARs for FCTC Support

Operating System	APAR
z/OS and OS/390	OW50966
VM	VM62942
VSE	DY45702

Recommendations for FCTC device numbering scheme

There are a number of schemes that can be used for developing the device numbers for the FICON CTC devices. This section discusses different schemes and provides recommendations for different configurations. Defining ESCON or FICON CTC configurations requires an understanding of CTC operations and the different methods available for defining the connections. It is also necessary to take into consideration the subsystem requirements for using CTC connections. As the number of images in a complex grows, the CTC definition process becomes increasingly complex.

The factors affecting the design and definition of the FCTC configuration are similar to those for ESCON SCTC control units and devices, but with these additional considerations:

- ▶ A single FICON (FC) channel per processor can be used for supporting FCTC connections between images on one or more processors, while an ESCON CNC-CTC channel requires a pair.
- ▶ The FICON (FC) channel can operate in full duplex mode.

FICON CTC device number recommendations

There are a number of different device numbering schemes that can be used for the FICON CTC configuration. The choice depends on the installation's hardware configuration supporting the FICON CTC configuration and the degree of simplicity required in the definition process. The following discussion offers different methods that depend on whether the installation chooses to implement FCTC with one FICON (FC) channel per processor or two.

ESCON device numbering scheme enhancement for FICON

One of the device numbering schemes that can be used for FICON CTC devices is based on the method developed by IBM and recommended for use with ESCON CTC devices when ESCON was introduced in the early 1990s. This device numbering scheme is discussed in detail in "ESCON CTC device numbering scheme" on page 5.

There are many advantages to continuing to use this device numbering scheme when the CTC configuration includes FICON CTC devices. Apart from ease of migration, operations and systems programmers can easily identify the use (send or receive) and target system for any given CTC device. It also simplifies the CTC definition process in that the same operating system CTC device definitions can be used by all z/OS and OS/390 operating systems, independent of the image and processor where that operating system runs.

In a FICON environment, the ESCON device numbering technique can be modified in the following ways:

- ▶ Using two FICON (FC) channels per processor, define all send FCTC control units and devices (with numbers of the form 4xxx) on one FICON (FC) channel and define all receive FCTC control units and devices (with numbers of the form 5xxx) on a second FC channel. This technique provides the easiest migration path.
- ▶ Using two FICON (FC) channels per processor, evenly spread the send CU/device definitions (4xxx) across the two FC channels, and evenly spread the receive CU/device definitions (5xxx) across the two FC channels. This method exploits the Fibre Channel full duplex data flow used by FICON, but has the disadvantage that there is a large amount of customization required for this technique.

Matching device number scheme

Another approach is to define matching device numbers for each end of the CTC connection. For example, device number 4300 in image LP-1 on processor 1 is used to communicate with device number 4300 in image LP-B on processor 2.

This scheme has a number of disadvantages. It does not enable the operator or systems programmer to identify the use or target system for a given CTC device number. It also makes it difficult to relocate operating systems between logical partitions on the same or a different processor.

Although this scheme appears simple to use at first, it is the most impractical of all possible approaches for defining CTC configurations and is not recommended.

FCTC configuration and definition examples

Based on the device numbering scheme discussed in “FICON CTC device number recommendations” on page 17, this section provides some sample FICON CTC configurations and I/O definitions.

FICON CTC using two FC channels per processor

A sample configuration with FICON CTC connections between six operating systems, three in logical partitions on each of two processors, is shown in Figure 15. Note that this implementation scheme, using two FC channels per processor, is based on the ESCON device numbering method and significantly eases the migration to FICON CTC.

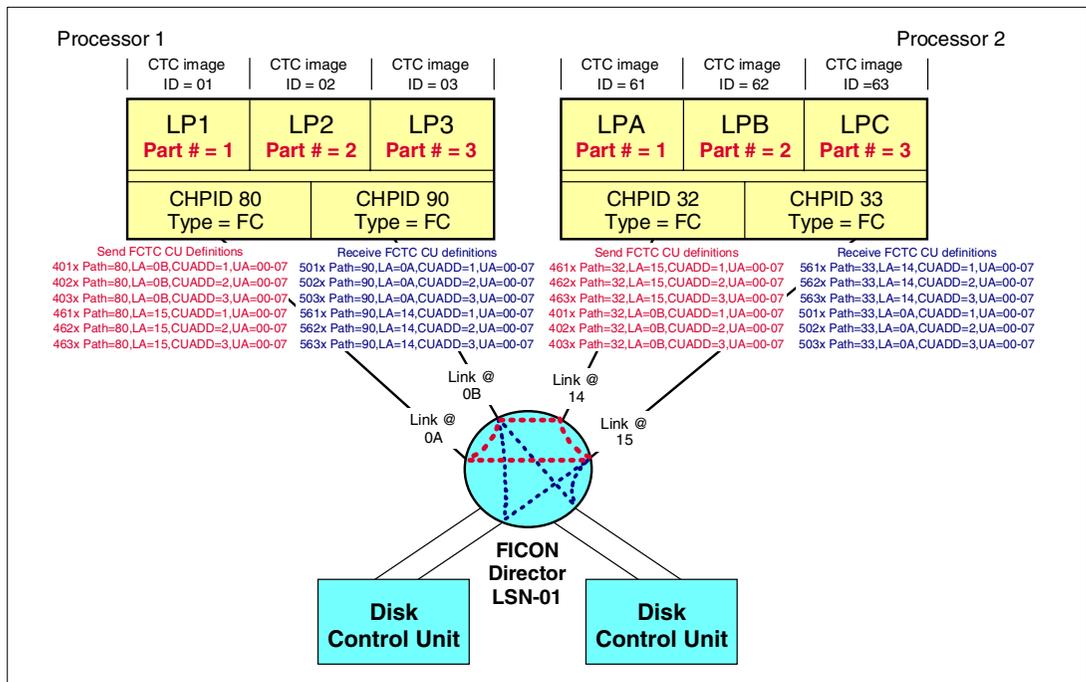


Figure 15 FICON CTC configuration using two FC channels per processor

This configuration and definition example has the following characteristics:

- ▶ All sending FCTC control units and devices (with FCTC CUs and FCTC Device using definition numbers 4xxx) are configured on one FICON (FC) channel and all the receiving

FCTC control units and devices (with FCTC CUs and FCTC Devices using definition numbers 5xxx) are configured on another FICON (FC) channel.

- ▶ The FICON (FC) channel may operate in half duplex mode for the CTC communication (either all the I/O requests will be for sending data, or all the I/O requests will be for receiving data) but it will operate in full duplex mode for communication to other (non-CTC) control units and devices, such as disk and tape (the I/O requests will be for both sending and receiving data).

The device number method shown here uses the middle two digits of the 4xxx and 5xxx device numbers to represent a processor's image CTC-image-ID. This allows the same operating system's subsystem CTC definitions to be used by all operating system instances, regardless of the supporting logical partition that is used.

A detailed example of the FCTC mapping for two devices in this configuration is shown in Figure 16.

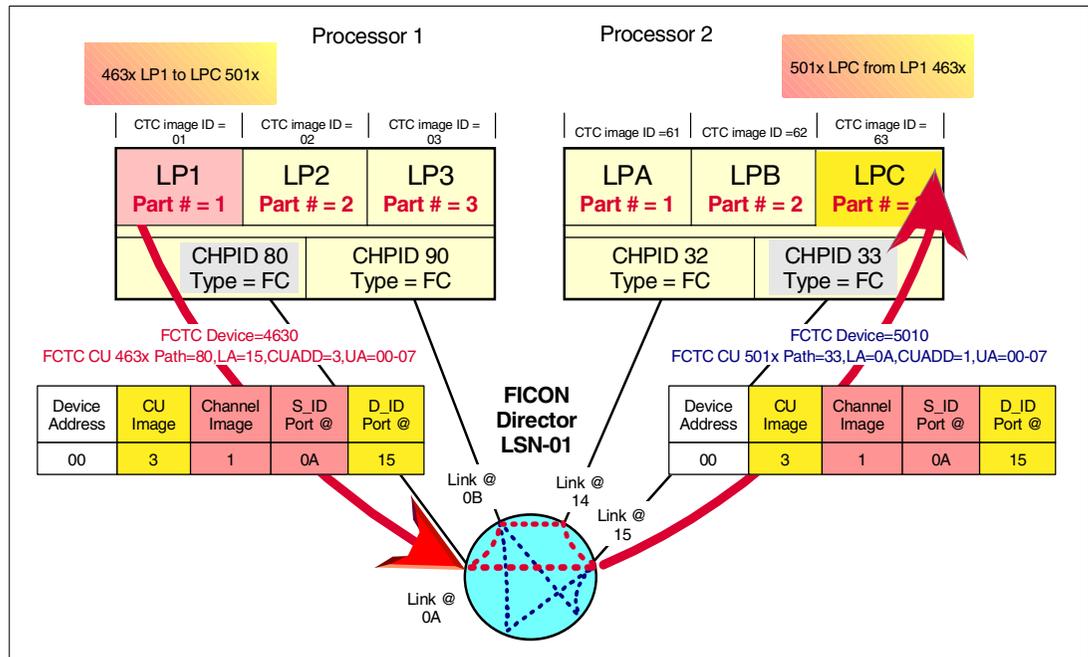


Figure 16 FCTC mapping example - from LP1 on processor 1 to LPC on processor 2

Here, sending device number 463x in LP1, defined to logical partition number 1 on processor 1, communicates with device number 501x in LPC (defined logical partition number 3 on processor 2). The definition for the control unit supporting device numbers 463x includes the following essential elements:

- ▶ Path=80, a Type=FC channel connected to LSN-01
- ▶ DLA=14, the FICON (FC) channel one-byte destination link address
- ▶ CUADD=3, the target logical partition number in processor 2 (the processor associated with the CHPID that is connected to FICON switch LSN 01 and Link Address 14 (switch Port Address 14))
- ▶ UA=00, the unit address, which must match the target device unit-address

Communication in the opposite direction, from LPC on processor 2 to LP1 on processor 1, is shown in Figure 17 on page 20. Here, the sending device number 401x on LPC, defined to logical partition number 3 on processor 2, communicates with device number 563x on LP1, defined to logical partition number 1 on processor 1. The definition for the control unit supporting device numbers 401x includes the following essential elements:

- ▶ Path=32, a Type=FC channel connected to LSN-01
- ▶ DLA=0B, the FICON (FC) channel one-byte destination link address
- ▶ CUADD=1, the target logical partition number in processor 1 (the processor associated with the CHPID that is connected to FICON switch LSN 01 and Link Address 0B (switch Port Address 0B))
- ▶ UA=00, the unit address, which must match the target device unit address

The z/Architecture and s/390 Architecture channel-to-device mapping rules dictate that the incoming source identifier (S_ID-Channel Image) is mapped to a defined destination identified (D_ID-Control Unit Image), along with the device address (UA).

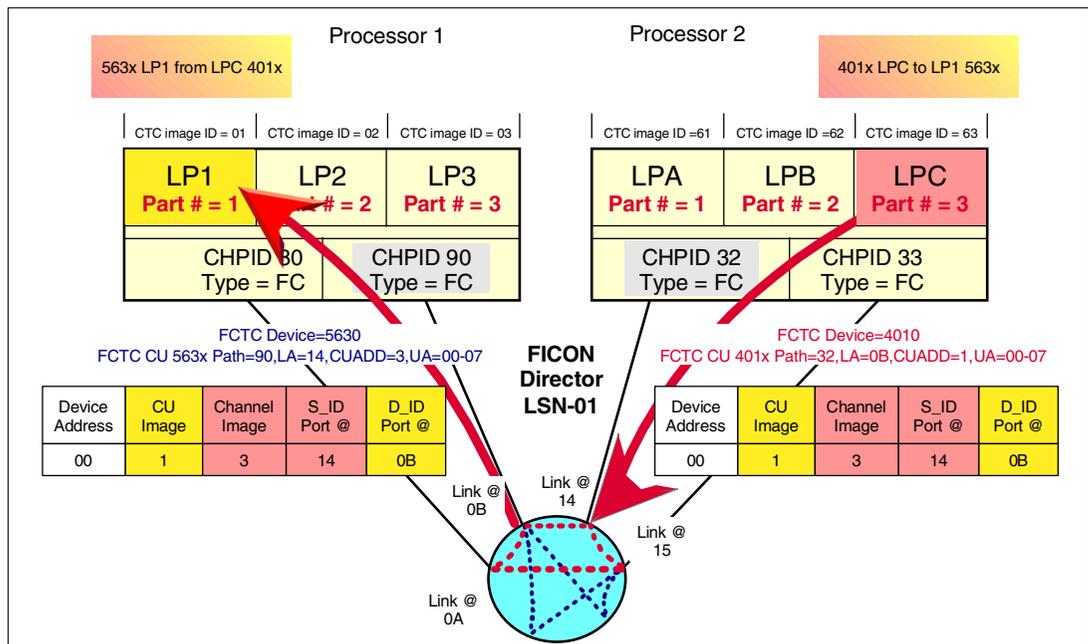


Figure 17 FCTC mapping example - from LPC on processor 2 to LP1 on processor 1

The definition requirements for FICON CTC control units and devices are:

- ▶ Processor:
 - Mode can be either Basic or LPAR
 - In LPAR mode, each logical partition is assigned a name and a logical partition number (single hexadecimal digit, unique within the processor)
- ▶ CHPID:
 - Define CHPID type as FC
 - Mode can be shared or un-shared in LPAR mode
- ▶ Control unit:
 - Define CU address (CUADD), which is the target logical partition's partition number
 - Type is FCTC

- ▶ I/O device:
 - Type is FCTC
 - Device Unit Address, which will be matched to the same unit address value at the target FCTC CU end

The I/O definitions for processor 1 in this configuration are shown in Figure 18 on page 21. The I/O definitions for processor 2 in this configuration are shown in Figure 19 on page 22.

```

ID      MSG1='ficon fctc definition using the 4xxx and 5xxx CTC Image_ID definition method',
        MSG2='does not take advantage of the FICON channel full duplex capability',
        SYSTEM=(2064,1)
RESOURCE PARTITION=( (LP1,1) , (LP2,2) , (LP3,3) )
CHPID  PATH=( 80) , SHARED, PARTITION=( (LP1,LP2,LP3) , (LP1,LP2,LP3) ) , SWITCH=21, TYPE=FC
CHPID  PATH=( 90) , SHARED, PARTITION=( (LP1,LP2,LP3) , (LP1,LP2,LP3) ) , SWITCH=21, TYPE=FC
CHPID  PATH=( 20) , SHARED, PARTITION=( (LP1,LP2,LP3) , (LP1,LP2,LP3) ) , SWITCH=22, TYPE=FC
CHPID  PATH=( 30) , SHARED, PARTITION=( (LP1,LP2,LP3) , (LP1,LP2,LP3) ) , SWITCH=22, TYPE=FC
CNTLUNIT CUNUMBR=4010, PATH=( 80) , UNITADD=( (00,008) ) , LINK=( 0B) ,
          CUADD=1, UNIT=FCTC
CNTLUNIT CUNUMBR=4020, PATH=( 80) , UNITADD=( (00,008) ) , LINK=( 0B) ,
          CUADD=2, UNIT=FCTC
CNTLUNIT CUNUMBR=4030, PATH=( 80) , UNITADD=( (00,008) ) , LINK=( 0B) ,
          CUADD=3, UNIT=FCTC
CNTLUNIT CUNUMBR=4610, PATH=( 80) , UNITADD=( (00,008) ) , LINK=( 15) ,
          CUADD=1, UNIT=FCTC
CNTLUNIT CUNUMBR=4620, PATH=( 80) , UNITADD=( (00,008) ) , LINK=( 15) ,
          CUADD=2, UNIT=FCTC
CNTLUNIT CUNUMBR=4630, PATH=( 80) , UNITADD=( (00,008) ) , LINK=( 15) ,
          CUADD=3, UNIT=FCTC
CNTLUNIT CUNUMBR=5010, PATH=( 90) , UNITADD=( (00,008) ) , LINK=( 0A) ,
          CUADD=1, UNIT=FCTC
CNTLUNIT CUNUMBR=5020, PATH=( 90) , UNITADD=( (00,008) ) , LINK=( 0A) ,
          CUADD=2, UNIT=FCTC
CNTLUNIT CUNUMBR=5030, PATH=( 90) , UNITADD=( (00,008) ) , LINK=( 0A) ,
          CUADD=3, UNIT=FCTC
CNTLUNIT CUNUMBR=5610, PATH=( 90) , UNITADD=( (00,008) ) , LINK=( 14) ,
          CUADD=1, UNIT=FCTC
CNTLUNIT CUNUMBR=5620, PATH=( 90) , UNITADD=( (00,008) ) , LINK=( 14) ,
          CUADD=2, UNIT=FCTC
CNTLUNIT CUNUMBR=5630, PATH=( 90) , UNITADD=( (00,008) ) , LINK=( 14) ,
          CUADD=3, UNIT=FCTC
IODEVICE ADDRESS=( 4010,008) , UNITADD=00, CUNUMBR=( 4010) , STADET=Y,
          PARTITION=( LP2,LP3) , UNIT=FCTC
IODEVICE ADDRESS=( 4020,008) , UNITADD=00, CUNUMBR=( 4020) , STADET=Y,
          PARTITION=( LP1,LP3) , UNIT=FCTC
IODEVICE ADDRESS=( 4030,008) , UNITADD=00, CUNUMBR=( 4030) , STADET=Y,
          PARTITION=( LP1,LP2) , UNIT=FCTC
IODEVICE ADDRESS=( 5010,008) , UNITADD=00, CUNUMBR=( 5010) , STADET=Y,
          PARTITION=( LP2,LP3) , UNIT=FCTC
IODEVICE ADDRESS=( 5020,008) , UNITADD=00, CUNUMBR=( 5020) , STADET=Y,
          PARTITION=( LP1,LP3) , UNIT=FCTC
IODEVICE ADDRESS=( 5030,008) , UNITADD=00, CUNUMBR=( 5030) , STADET=Y,
          PARTITION=( LP1,LP2) , UNIT=FCTC
IODEVICE ADDRESS=( 4610,008) , UNITADD=00, CUNUMBR=( 4610) , STADET=Y,
          PARTITION=( LP1,LP2,LP3) , UNIT=FCTC
IODEVICE ADDRESS=( 4620,008) , UNITADD=00, CUNUMBR=( 4620) , STADET=Y,
          PARTITION=( LP1,LP2,LP3) , UNIT=FCTC
IODEVICE ADDRESS=( 4630,008) , UNITADD=00, CUNUMBR=( 4630) , STADET=Y,
          PARTITION=( LP1,LP2,LP3) , UNIT=FCTC
IODEVICE ADDRESS=( 5610,008) , UNITADD=00, CUNUMBR=( 5610) , STADET=Y,
          PARTITION=( LP1,LP2,LP3) , UNIT=FCTC
IODEVICE ADDRESS=( 5620,008) , UNITADD=00, CUNUMBR=( 5620) , STADET=Y,
          PARTITION=( LP1,LP2,LP3) , UNIT=FCTC
IODEVICE ADDRESS=( 5630,008) , UNITADD=00, CUNUMBR=( 5630) , STADET=Y,
          PARTITION=( LP1,LP2,LP3) , UNIT=FCTC

```

Figure 18 Processor 1 definitions

```

ID      MSG1='ficon fctc definition using the 4xxx and 5xxx CTC Image_ID definition method',
        MSG2='does not take advantage of the FICON channel full duplex capability',
        SYSTEM=(2064,1)
RESOURCE PARTITION=(LPA,1),(LPB,2),(LPC,3)
CHPID  PATH=(32),SHARED,PARTITION=((LPA,LPB,LPC),(LPA,LPB,LPC)),SWITCH=21,TYPE=FC
CHPID  PATH=(33),SHARED,PARTITION=((LPA,LPB,LPC),(LPA,LPB,LPC)),SWITCH=21,TYPE=FC
CHPID  PATH=(54),SHARED,PARTITION=((LPA,LPB,LPC),(LPA,LPB,LPC)),SWITCH=22,TYPE=FC
CHPID  PATH=(55),SHARED,PARTITION=((LPA,LPB,LPC),(LPA,LPB,LPC)),SWITCH=22,TYPE=FC
CNTRLUNIT CUNUMBR=4010,PATH=(32),UNITADD=((00,008)),LINK=(0B),
          CUADD=1,UNIT=FCTC
CNTRLUNIT CUNUMBR=4020,PATH=(32),UNITADD=((00,008)),LINK=(0B),
          CUADD=2,UNIT=FCTC
CNTRLUNIT CUNUMBR=4030,PATH=(32),UNITADD=((00,008)),LINK=(0B),
          CUADD=3,UNIT=FCTC
CNTRLUNIT CUNUMBR=4610,PATH=(32),UNITADD=((00,008)),LINK=(15),
          CUADD=1,UNIT=FCTC
CNTRLUNIT CUNUMBR=4620,PATH=(32),UNITADD=((00,008)),LINK=(15),
          CUADD=2,UNIT=FCTC
CNTRLUNIT CUNUMBR=4630,PATH=(32),UNITADD=((00,008)),LINK=(15),
          CUADD=3,UNIT=FCTC
CNTRLUNIT CUNUMBR=5010,PATH=(33),UNITADD=((00,008)),LINK=(0A),
          CUADD=1,UNIT=FCTC
CNTRLUNIT CUNUMBR=5020,PATH=(33),UNITADD=((00,008)),LINK=(0A),
          CUADD=2,UNIT=FCTC
CNTRLUNIT CUNUMBR=5030,PATH=(33),UNITADD=((00,008)),LINK=(0A),
          CUADD=3,UNIT=FCTC
CNTRLUNIT CUNUMBR=5610,PATH=(33),UNITADD=((00,008)),LINK=(14),
          CUADD=1,UNIT=FCTC
CNTRLUNIT CUNUMBR=5620,PATH=(33),UNITADD=((00,008)),LINK=(14),
          CUADD=2,UNIT=FCTC
CNTRLUNIT CUNUMBR=5630,PATH=(33),UNITADD=((00,008)),LINK=(14),
          CUADD=3,UNIT=FCTC
IODEVICE ADDRESS=(4010,008),UNITADD=00,CUNUMBR=(4010),STADET=Y,
          PARTITION=(LPA,LPB,LPC),UNIT=FCTC
IODEVICE ADDRESS=(4020,008),UNITADD=00,CUNUMBR=(4020),STADET=Y,
          PARTITION=(LPA,LPB,LPC),UNIT=FCTC
IODEVICE ADDRESS=(4030,008),UNITADD=00,CUNUMBR=(4030),STADET=Y,
          PARTITION=(LPA,LPB,LPC),UNIT=FCTC
IODEVICE ADDRESS=(5010,008),UNITADD=00,CUNUMBR=(5010),STADET=Y,
          PARTITION=(LPA,LPB,LPC),UNIT=FCTC
IODEVICE ADDRESS=(5020,008),UNITADD=00,CUNUMBR=(5020),STADET=Y,
          PARTITION=(LPA,LPB,LPC),UNIT=FCTC
IODEVICE ADDRESS=(5030,008),UNITADD=00,CUNUMBR=(5030),STADET=Y,
          PARTITION=(LPA,LPB,LPC),UNIT=FCTC
IODEVICE ADDRESS=(4610,008),UNITADD=00,CUNUMBR=(4610),STADET=Y,
          PARTITION=(LPB,LPC),UNIT=FCTC
IODEVICE ADDRESS=(4620,008),UNITADD=00,CUNUMBR=(4620),STADET=Y,
          PARTITION=(LPA,LPC),UNIT=FCTC
IODEVICE ADDRESS=(4630,008),UNITADD=00,CUNUMBR=(4630),STADET=Y,
          PARTITION=(LPA,LPB),UNIT=FCTC
IODEVICE ADDRESS=(5610,008),UNITADD=00,CUNUMBR=(5610),STADET=Y,
          PARTITION=(LPB,LPC),UNIT=FCTC
IODEVICE ADDRESS=(5620,008),UNITADD=00,CUNUMBR=(5620),STADET=Y,
          PARTITION=(LPA,LPC),UNIT=FCTC
IODEVICE ADDRESS=(5630,008),UNITADD=00,CUNUMBR=(5630),STADET=Y,
          PARTITION=(LPA,LPB),UNIT=FCTC

```

Figure 19 Processor 2 definitions

FICON CTC using one FC channel per processor

A sample FICON CTC configuration using one FC channel per processor is shown in Figure 20 on page 23.

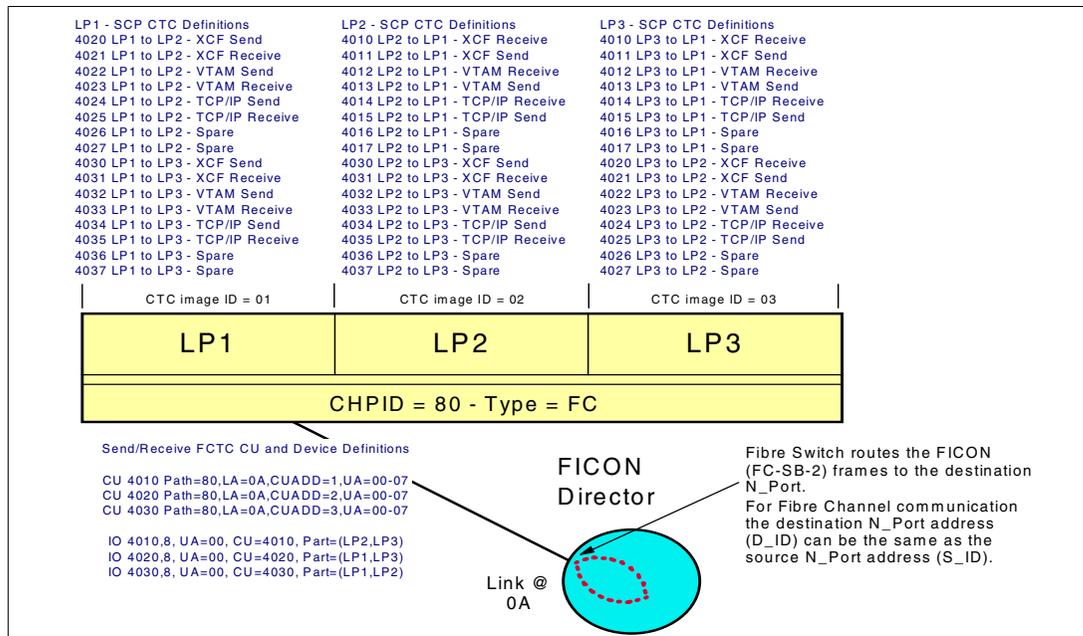


Figure 20 FICON CTC configuration using one FC channel per processor

In the Fibre Channel switch environment, it is the Fibre Channel switch that routes the Fibre Channel FICON (FC-SB-2) frames to the destination N_Port (D_ID).

For Fibre Channel communication the destination N_Port address (D_ID) (the target port) can be the same as the source N_Port address (S_ID) (the originating port).

This means that an FCTC CU destination link address (the fibre channel switch port address where the FICON frame is being sent to) will be the same link address (port address) that the channel is connected to. For Fibre Channel, because the destination port address can be the same as the source port address, this allows communication between LPARs on the same processor using only a single FICON (FC) channel.

When communicating between LPARs (on the same processor), data will flow as follows:

- From the originating LPAR (source LPAR) to the FICON (FC) channel
- As a Fibre Channel frame with a FICON (FC-SB-2) payload from the FICON (FC) channel
- To the Fibre Channel switch port (that the channel is connected to)
- Where the destination port address (in the Fibre Channel frame) is the same as the source port address
- The Fibre Channel switch port will route the frame back up the same link to the same channel
- The channel will then deliver the data to the target LPAR (destination LPAR)

FCTC definition using a single FICON (FC) channel

The FCTC hardware definition method used when there is only a single FICON (FC) channel in the FCTC configuration can be the same as when using two FICON (FC) channels for FCTC to FCTC communication. See the FCTC CU and device definitions within Figure 20 on page 23.

However, when only a single FICON (FC) channel is used in a FCTC configuration, each operating system's CTC usage definition would need to be customized for the partition that it is running on. These are also shown within Figure 20 on page 23.

Because of the requirement of having to define the operating system's CTC use for the image that the operating system is running in, the same flexibility of movement of operating systems using a common operating system CTC definition does not exist. Most large installations like the idea of having a common operating system CTC definition, as well as the flexibility of movement of the operating system between LPARs on the same processor (or even to LPARs on any processor) without having to change the operating system definition.

If installations still want the flexibility of movement of the operating system using a single operating system CTC definition, then it is recommended to use two FICON (FC) channels on each processor for CTC configurations. Remember that when using two FICON (FC) channels for FCTC communication, these same two FICON (FC) channels can still be defined to be used for I/O operations to other CU types, in addition to FCTC CUs (at the same time). In reality, it does not cost the large system user more FICON channels in total, because the existing FICON (FC) channels can be used by CTC I/O, as well as for disk and tape I/O requests.

FICON CTC control unit function balancing

In a FICON CTC configuration, although FICON CTC control units are defined at each FICON (FC) channel end, only one of the two ends (that form the FCTC logical connection) provides the FICON FCTC control unit function. During initialization of the logical connection between two ends of a FICON CTC connection, the FICON (FC) channel that is to provide the FICON CTC control unit function (one of the two that makes up the logical connection) is determined using an algorithm that takes the following into consideration:

- ▶ FICON CTC control unit support

FICON CTC control unit support is only provided in the FICON native (FC) channel on a z900 at driver level 3C or later. If one end of the connection is a z900 CPC at driver level 3C or later, and the other end is a z900 GA1 or 9672 G5/G6 processor, the z900 driver level 3C processor provides the FICON CTC control unit function.

- ▶ FICON CTC CU function, count

If both ends of the connection support the FICON CTC control unit function, that is, both ends of the FICON CTC connection are on a z900 CPC at driver level 3C or later, then the count of FCTC CU functions already supported is taken into consideration. The FC channel with fewer FCTC CU functions being supported is selected to provide the FICON CTC control unit function.

- ▶ Fibre Channel Node/Port, WWN

If both ends of the connection are on a z900 CPC at driver level 3C or later and each has an equal CTC CU function count, then the FC channel with the lower WWN (FC World Wide Name) provides the FICON CTC control unit function. The WWN is provided during Port Log_in (PLOGI) time. In the initialization sequence, the PLOGI precedes the ELP and the channel remembers the WWN for each N_Port that has logged into it.

The FICON CTC control unit function establishment algorithm is summarized in Figure 21 on page 25.

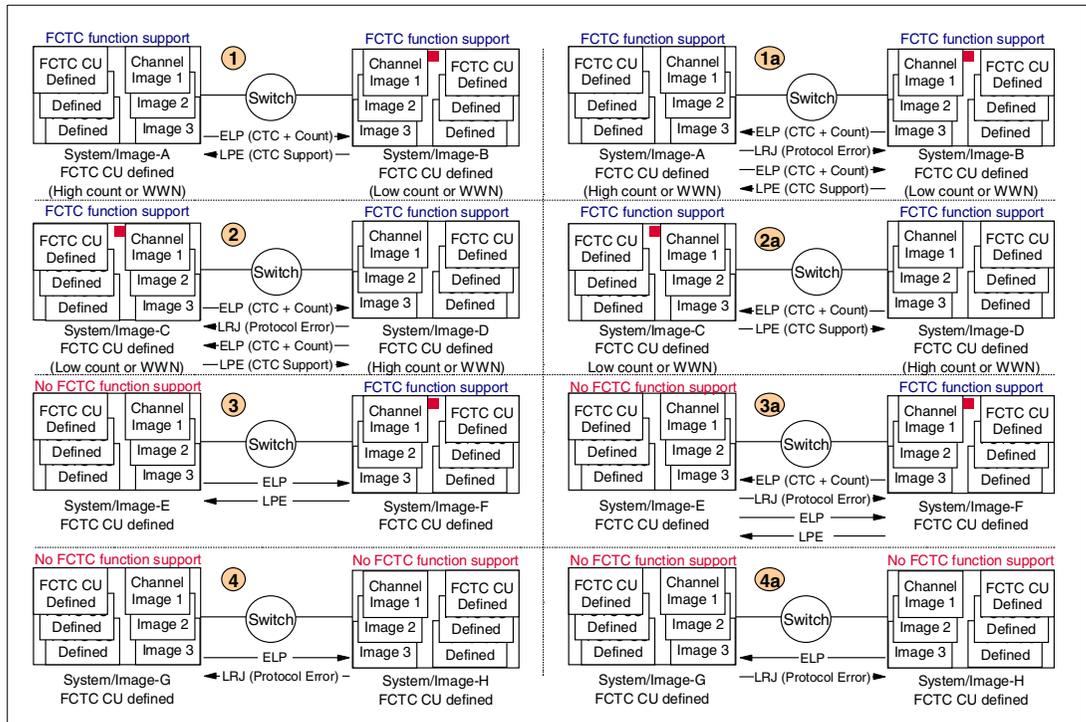


Figure 21 FICON CTC control unit establishment algorithm

The eight sections of the figure illustrate the possible scenarios for establishing the FICON CTC CU function. For each sequence, note that the ELP originating from a z900 CPC at driver 3C or later contains both the CTC bit, which indicates FCTC CU function support, and the count of FCTC CU functions already supported by that channel.

1. In diagram 1, systems A and B are both FCTC CU function capable. System A has a high FCTC CU function count and a higher WWN than system B.

If an ELP is sent from system A with the CTC bit on and if the receiving end (system B) accepts the ELP with an LPE with the CTC bit set, then the receiving end (system B) provides the FCTC CU function and a logical path is established. The receiving end (system B) accepts the ELP, indicated by sending the LPE back to system A. System A on receipt of the LPE, will then know that the FCTC CU function is being provided at the B end. This happens because all of the following conditions are met:

- It has a matching FCTC CU definition. That is, at the receiving channel, the incoming frame sources, namely the port-identifier (source port address) and the channel image identifier (source logical partition number) each match a defined destination; the port-identifier (destination port address) and the channel image identifier (source logical partition number).
 - It supports the FCTC CU function.
 - It has the lower FCTC CU function count, or the counts were equal and the receiving end has the lower WWN.
- a. Diagram 1a shows the flow when system B initiates the establishment sequence by sending an ELP with the CTC bit on. The ELP also contains the established FCTC CU function count for the system B channel.

System A is capable of establishing the FCTC CU function. However, because either its FCTC CU function count is higher or the counts are equal and its WWN is higher, system A (the receiving end) rejects the ELP with an LRJ indicating a protocol error. System A then sends an ELP with the CTC bit set and the FCTC CU function count. System B, the receiving end, accepts the ELP with an LPE with the CTC bit set. System B, the receiving end, provides the FCTC CU function and a logical path is established.

2. In diagram 2, systems C and D are both FCTC CU-capable. System C has a low FCTC CU function count or WWN.

If an ELP is sent from system C with the CTC bit on and if the receiving end (system D) rejects the ELP with an LRJ indicating a protocol error, then the sending end (system C) provides the FCTC CU function. In this scenario, the receiving end (system D) rejects the ELP because of one of the following conditions:

- It supports the FCTC CU function but there is no matching CU definition; the incoming source does not match a defined destination.
- It has the higher FCTC CU function count.
- The FCTC CU function counts are equal, but the sending end (system C) has the WWN advantage.

- a. In diagram 2a, if system D initiates the sequence by sending an ELP with the CTC bit and count, the receiving end (system C) accepts the ELP with an LPE because it had the lower FCTC CU function count or WWN.

3. In diagram 3, system E is not FCTC CU-capable, and system F is FCTC CU-capable.

If system E sends an ELP without the CTC bit, indicating the FICON channel does not support the FCTC CU function, and system F, the receiving system, has been defined to support the FCTC CU function and has a matching CU definition, then the receiving end accepts the ELP with an LPE without the CTC bit.

System F, the receiving end of the ELP, now provides the FCTC CU function, and a logical path is established between the ends of the FICON CTC logical connection. The sending end (system E) treats this as a normal ELP-LPE sequence.

- a. In diagram 3a, if system F initiates the sequence with an ELP with the CTC bit and count, then system E, where the FICON channel does not support FCTC CU function, responds with an LRJ indicating protocol error. Eventually system E will send an ELP without the CTC bit, and system F responds with an LPE. A logical path is established and the FICON channel on system F will provide the FCTC CU function.

4. Neither system G nor H are FCTC CU-capable.

If system G sends an ELP without the CTC bit, indicating the FICON channel does not support the FCTC CU function, and system H, the receiving system, also does not support the FCTC CU function, then the receiving system (system H) responds with an LRJ. This means that neither end of the logical connection supports the FCTC CU function. No logical path is established.

- a. If system H initiates the flow with an ELP without the CTC bit, then system G (the receiving end) rejects it with an LRJ because it does not support the FCTC CU function.

The output of the z/OS operator command **D M=DEV(dddd)** for any FICON CTC device can be used to determine which FICON CHPID of the pair is providing the FCTC control unit function. Sample output is shown in “z/OS commands” on page 27.

Migration considerations

Migration from ESCON SCTC devices (using ESCON channels) to FICON FCTC devices (using FICON channels) can be achieved with a minimum of disruption. The migration only requires that different control unit numbers and device numbers are used. As stated previously, the hardware support of the FCTC CU function by the z900 requires the installation of, or upgrade to, a z900 at hardware driver level 3C or later. The newly defined FICON FCTC configuration can co-exist with the existing ESCON SCTC configuration. Exploiters, such as XCF, can use both FICON CTCs and ESCON CTCs at the same time (FCTC versus SCTC). For example, XCF supports signalling paths on both FCTC devices and SCTC devices at the same time.

Operations, availability and recovery

This section describes the external operations, configuring for availability, and recovery of error conditions, for FICON FCTC devices.

z/OS commands

The Display Units command for FICON CTC devices shows the device type FCTC. Sample outputs are shown in the following two screens.

The device type shown for FICON CTC devices is FCTC, and the device type for ESCON CTC devices would be shown as SCTC. BCTC devices may also be defined on an ESCON CHPID for older applications using basic CTC operating mode (basic CTC command set).

Note: The FICON FCTC CU function only supports the extended CTC operating mode (extended CTC command set); it does not support the basic CTC operating mode (basic CTC command set).

The FCTC device types for FICON, along with the SCTC and BCTC device types for ESCON, all belong in the IOS CTC device class.

The first display shows the usage of the display unit CTC command:

```
D U,CTC,,4000,4
IEE457I 20.32.46 UNIT STATUS 372
UNIT TYPE STATUS          VOLSER      VOLSTATE
4010 FCTC A-BSY
4011 FCTC O
4012 FCTC O
4013 FCTC O
```

The next display shows the usage of the display device command for four devices, starting from unit address 4000.

```

D U,,,4000,4
IEE457I 20.32.46 UNIT STATUS 372
UNIT TYPE STATUS          VOLSER      VOLSTATE
4010 FCTC A-BSY
4011 FCTC O
4012 FCTC O
4013 FCTC O

```

Apart from the device type (FCTC) and the channel type (FC) shown in the z/OS and OS/390 command displays, there is no difference operationally in the management of FICON FCTC devices from ESCON SCTC devices. For example, FICON FCTC devices can be managed using z/OS or OS/390 VARY and SETXCF commands, just like ESCON SCTC devices.

The z/OS or OS/390 Display Matrix Device command for FICON FCTC devices is similar to that for ESCON SCTC devices, but has the enhancement of also displaying information in the device Node Element Descriptor (NED), indicating at which end of the FCTC logical connection the FCTC CU function is provided.

```

D M=DEV(5630)
IEE174I 20.33.27 DISPLAY M 376
DEVICE 5630 STATUS=ONLINE
CHP                90
DEST LINK ADDRESS  14
DEST LOGICAL ADDRESS 03
PATH ONLINE        Y
CHP PHYSICALLY ONLINE Y
PATH OPERATIONAL   Y
ND                = NOT AVAILABLE
DEVICE NED = 002064.CTC.IBM.02.148AA0010ECB
                *

```

The preceding screen shows the output of the z/OS (or OS/390) Display Matrix Device command, which includes the display of the device Node Element Descriptor (NED) for the FICON FCTC device 5630 in image LP1 on processor 1. Alpha character A in the device NED (which we marked with an * underneath it) indicates that the FCTC control unit function is supported by the CHPID (90) at this end of the FCTC-to-FCTC logical connection (LP1 to LPC) for the image that the command was entered from.

The following screen shows the display of the device NED for the FICON CTC device number 4010 in image LPC on processor 2. Alpha character B in the device NED (which we marked with an * underneath it) indicates that the FCTC control unit function is supported by the CHPID at the other end of the FCTC connection for this image, 3, (LPC) to the other image, 1, (LP1).

```

D M=DEV(4010)
IEE174I 20.38.44 DISPLAY M 386
DEVICE 4010 STATUS=ONLINE
CHP                32
DEST LINK ADDRESS  0B
DEST LOGICAL ADDRESS 01
PATH ONLINE        Y
CHP PHYSICALLY ONLINE Y
PATH OPERATIONAL   Y
ND                 = NOT AVAILABLE
DEVICE NED = 002064.CTC.IBM.02.143AB0013487
*
```

There is no difference in the output of the z/OS Display Matrix CHPID command for a FICON native (FC) channel supporting FICON FCTC control units from that of the same FICON (FC) channel supporting disk or tape control units.

```

D M=CHP(32)
IEE174I 20.34.06 DISPLAY M 378
CHPID 32:  TYPE=1B, DESC=FICON SWITCHED, ONLINE
DEVICE STATUS FOR CHANNEL PATH 32
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
4010 +  +  +  +  +  +  +  +  .  .  .  .  .  .  .  .
4020 +  +  +  +  +  +  +  +  .  .  .  .  .  .  .  .
4030 +  +  +  +  +  +  +  +  .  .  .  .  .  .  .  .
4610 +  +  +  +  +  +  +  +  .  .  .  .  .  .  .  .
4620 +  +  +  +  +  +  +  +  .  .  .  .  .  .  .  .
4630 +  +  +  +  +  +  +  +  .  .  .  .  .  .  .  .
SWITCH DEVICE NUMBER = NONE
***** SYMBOL EXPLANATIONS *****
+ ONLINE      @ PATH NOT VALIDATED  - OFFLINE      . DOES NOT EXIST
* PHYSICALLY ONLINE  $ PATH NOT OPERATIONAL
```

Configuring for availability

The recommendations for configuring FICON FCTC devices for high availability are the same as those for ESCON SCTC devices:

- ▶ Provide redundant FICON FCTC connections with as few common hardware elements as possible. That is, configure redundant FICON FCTC connections on:
 - Different channels
 - Channels in different channel groups
 - Through different switches/fabrics
- ▶ Spread the receiving CTC devices across different physical hardware CTC connections.
- ▶ Spread the sending CTC devices across different physical hardware CTC connections.
- ▶ Spread the CTC devices for an exploiter across different physical hardware CTC connections.

Redundant CTC connections can be mixed across ESCON and FICON channels and provide the same functionality. Exploiters of CTC infrastructure for intersystem communication, such as XCF and VTAM, have no reliance on whether the CTC device is FCTC or SCTC; however, FICON CTCs provide a number of benefits over ESCON CTCs, including increased performance and connectivity.

FCTC operational and functional characteristics

This section covers some of the FCTC operational and functional characteristics.

Operational characteristics

The FICON native (FC) channel supporting a FICON CTC control unit has the same operational and functional characteristics as any FC channel supporting other types of control units. To summarize:

- ▶ The FICON native (FC) channel supports a maximum of 32 concurrent I/O connections.
- ▶ The link bandwidth for the FICON native (FC) channel is 100 MBps.
- ▶ The distance from the channel to control unit or channel to switch or switch to control unit link is up to 10 km (20 km with an RPQ) for FICON channels using long wavelength laser FOSAs.
- ▶ The channel-to-CU end-to-end distance before data droop is up to 100 km for FICON.
- ▶ The FICON channel supports up to 16,384 devices.
- ▶ FICON channels use frame and Information Unit (IU) multiplexing control to provide greater exploitation of the priority I/O queueing mechanisms within the FICON-capable control units, such as the IBM ESS 2105, and the 2064 Channel Subsystem priority queueing hardware function.
- ▶ Frame multiplexing support by the FICON channels, FICON switches, and FICON control units offers optimum utilization of the FC links.

More detailed information can be found in *FICON Native Implementation and Reference Guide*, SG24-6266.

Considerations for mixing control units on FICON channels

As described in “FICON CTC control unit function balancing” on page 24, if the FICON channel at each end of a FCTC connection is on a processor that supports the FICON CTC CU function, the channel that initiates the establishment of the FCTC CU function indicates the number of FCTC CUs that it currently supports. The receiving FC channels uses this number to determine which channel provides the FICON CTC CU function. The channel with the fewest FCTC number of FCTC CU functions supported will provide the FCTC CU function for the connection. This is an attempt to balance the load that FCTC connections place on each channel.

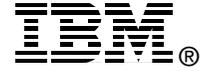
The FICON native (FC) channels supporting FCTC control units can also be used to support I/O operations to other control units, such as disk and tape. Note, however, that the establishment of logical paths to control units other than FCTC control units are *not* taken into consideration in the automatic FCTC CU function establishment balancing algorithm.

Performance

Detailed performance information will be provided by IBM at a later date. This information will include recommendations for the amount of CTC activity and mix of CTC and other I/O activity that is supported on a native FICON or FICON Express (FC) channel.

In general, z900 FICON Express channels have better performance characteristics than previous versions of FICON channels; therefore, it is recommended that FICON Express be implemented wherever possible, particularly for performance-critical applications.

This document created or updated on January 11, 2002.



Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HYJ Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this document was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

CUA®	MVS™	VTAM®
e (logo)® 	OS/390®	z/Architecture™
ESCON®	Perform™	z/OS™
FICON™	Redbooks (logo)™ 	z/VM™
IBM®	Resource Link™	zSeries™
IMS™	S/390®	
MVS/ESA™	SP™	

Other company trademarks

The following terms are trademarks of other companies:
C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

Other company, product, and service names may be trademarks or service marks of others.