FarSite
COMMUNICATIONS

FarSync
SDK
Developers
Toolkit

FarSync® SDK - Developer Toolkit
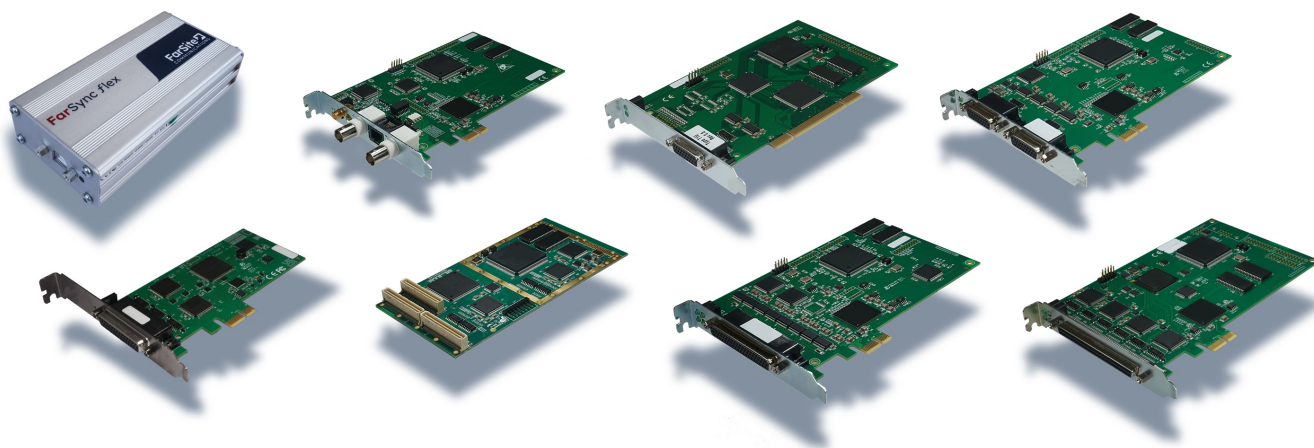
Key Features

- Fully documented APIs for Linux and Windows
- Includes free developer support from FarSite Engineering
- HDLC, V.120, G.704, LAPB and Transparent Bitstream
- Encoder/decoder for bitstream data formatting such as DAB / DMB
- LabVIEW application support
- Many Linux and Windows sample applications
- Supports the full range of FarSync adapters
- Line Monitor to assist rapid application development

Overview

The SDK supports writing applications to use the FarSync range of products on both Linux and Windows. It contains documentation, working sample applications, user libraries and test utility applications. There is everything a user needs to rapidly develop and test a wide variety of applications that use synchronous (HDLC framed) protocols, V.120, LAPB, asynchronous data or transparent bitstreams including Audio, G.704, MPEG Video T-DMB, DAB STI and DAB ETI with the bitstream encoder and decoders provided.

The SDK is intended to allow easy access to the set of hardware features offered by the various adapters in the range whilst also providing additional encoding and decoding to supplement that provided by the hardware.

Our Engineering department provides free email and telephone assistance to application developers using the API as part of the package provided when the FarSync SDK is purchased.

Note: This toolkit is not appropriate for the FarSync X25 range of products, they are supplied with their own specialist X.25 toolkit.

FarSync Windows API (FsWinAPI)

The Windows-based interface to the entire range of FarSync cards/devices is referred to as the FarSync Windows API (FsWinAPI). This is an extension of the MS Win32 COMM API and enables, for example, applications developed to support COM ports, to be easily ported to use FarSync support in both asynchronous or synchronous modes. This standardization enables the API to also be readily accessible from higher level environments e.g .NET (C#, VB etc.)

The majority of the API is line protocol-independent so applications can select between them easily (e.g. HDLC, Async, LAPB, DAB-ETI etc.)

The number and size of internal buffers can be configured to match application-specific latency requirements and scheduling characteristics.

The support for asynchronous I/O provides for optimised application designs without the need for polling.

FsWinAPI is a superset of the earlier SDCI API and so is backward-compatible with existing applications previously developed for the FarSync range of cards/devices.

Dynamic switching between Async and Sync line operation is supported.

Configuration of the various type of line signalling, NRZ, NRZI, Manchester Encoding, Conditioned Diphase, FM1 and FM0 are handled by various adapters in the range - see each particular adapter for a list of which line signals formats are supported.

Both 32 and 64-bit applications are supported.

The FsWinAPI User Guide manual in Adobe PDF format provides a comprehensive set of function call definitions and helpful advice on the best way to utilise the interface.

FarSync Windows API (FsWinAPI) continued

Synchronous HDLC, LAPB and V.120

The FsWinAPI allows connections both in raw HDLC frame mode, and the line error correcting LAPB and V.120 modes. The number and size of buffers use can be configured to achieve the optimum balance of throughput and latency.

Transparent bitstream operation

The driver has built in transparent bitstream encoding-decoding support that handles a number of common protocols commonly passed over a continuous bitstream. The protocols handled include G.704, MPEG Video T-DMB, DAB ETI.

The driver can be configured to identify and extract most fixed-length byte-oriented user frames; where frames can be defined in terms of single or alternate synchronizing patterns, frame length and number of sync patterns to acquire sync. It provides, for example, low level frame handling for E1 and DAB-ETI.

Sample send and receive programs demonstrating common modes of operation are supplied with sample data files.

DAB modes supported

ETI(NI, G.703) and ETI(NI, G.704) - on the FarSync adapters of type TE1, TE1R and TE1e.

ETI(NI, V.11) - on the FarSync adapters of type Flex, T1U, T2U, T2Ue, T2Ee, T4U, T4Ue, T4E+, T4Ee and T2U-PMC.

The API also handles completely transparent bit streaming operation which could be used, for example, to support a non-standard protocol or, for instance, raw Audio.

For bitstream operations the driver supports the ability to define the amount of buffering used in the data path. This feature is vital to allow reliable continuous line transmissions required for bitstream operation while the applications driving them are rescheduled. This buffering requirement must be balanced with the latency constraints of the transmission path.

Note: For DAB STI modes the General Bitstream encoding - decoding library should be used.

Extended Clocking

The FsWinAPI supports the TE1e card's option to use an externally sourced rate adapted clock to drive the E1 line which is commonly used in DAB and DMB operations.

It also supports the extended clocking features of the FarSync T4E+ that allow external clocks to be remotely sourced, rate adapted and supplied to selected ports on the adapters. Alternatively a clock from one of the ports can be used to drive the clocks of all the other ports.

Asynchronous Operation

To operate a port with asynchronous data under Windows, the FarSync driver supports an extension of the standard MS Win32 COMM API. This enables applications that currently interface to standard serial ports on Windows to be easily ported to use FarSync ports in asynchronous mode.

The FarSync Flex actually appears as a "COM port" device when configured for async mode and so can be used directly with standard or ported COM port-based apps.

Asynchronous operation is available for the FarSync Flex, T4E, T4E+ and T4Ue+Async.
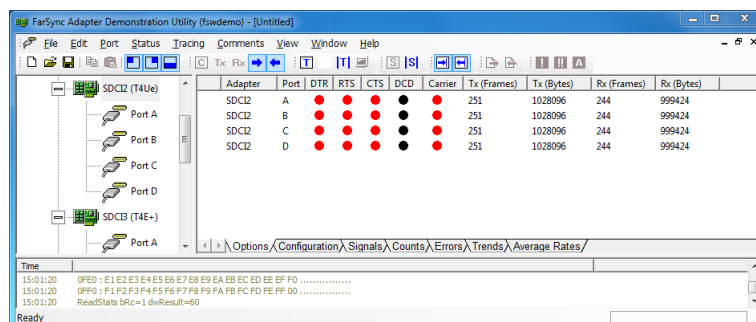
Sample Applications

A number of easy to follow sample applications are provided demonstrating the wide variety of features available with the adapters. GUI based demonstration and test utilities are supplied capable of utilizing the wide range of features supported by the adapters.

C, C#, VB and LabVIEW sample applications are supplied with the SDK. These support the range of communications modes supported by the FarSync adapters.

FSWDemo Test Utility

FSWDemo provided with the FarSync SDK is intended to demonstrate and test the operation of the Windows FsWinAPI interface and provide a really useful and powerful tool for checking various operating modes of the adapters.

Both HDLC frame and Transparent bitstream modes can be tested. Selection and testing of multiple lines and adapters is supported.

LabVIEW (Windows)

The FarSync API can also be used by LabVIEW applications/VIs. This support includes a set of LabVIEW-compatible methods and properties supported by a COM API interface layer which is supplied in extensible source form to enable even further customisation by the user if and when required to suit their particular application design.

An example VI (LabVIEW executable/program) for Windows is included in the FarSync SDK. This example demonstrates how to interface to the FarSync adapters from LabVIEW and illustrates control and configuration of the FarSync port as well as example data transfer usage.

Linux Char I/O API

The Linux API allows applications to send and receive HDLC format (bitsync) frames and also transparent bitstreams using the Char I/O API.

Sample applications supplied with the Developers Toolkit demonstrate both modes of operation. The source for the applications and drivers are included, with the Linux API manuals provided in Adobe PDF format. A monitor program is included in the Linux support that allows data sent and received by the application to be recorded in hex format.

Synchronous and E1 Operation

The API can supports data rates to 10 Mbits/s per port, with all the ports individually selectable for speed, clocking direction and mode of operation. For the maximum speeds supported by a particular adapter check the FarSync products particular technical specification.

Frame sizes up to 32 Kbytes are supported in HDLC mode to allow support for a wide variety of **specialist synchronous protocols**.

The transparent bitstream mode is suitable for **transparent data requirements such as Audio, MPEG Video T-DMB, DAB STI and DAB ETI.** See also the General Bitstream encoding and decoding library which can help shortcut the development time needed for these types of applications.

The API allow the full range of internal clock speeds to be selected and also selection of extended clocking options supported by FarSync adapters such as the FarSync T2Ee, FarSync T4E+ and FarSync T4Ee.

Configuration of the various type of line signalling, NRZ, NRZI, Manchester Encoding, Conditioned Diphase, FM1 and FM0 are handled by various adapters in the range - see each particular adapter datasheet for a list of which line signals formats are supported.

LAPB Operation

An API to the LAPB protocol is available though the hdlcX interface, all the ports on the adapters can be addressed by this method.

Asynchronous Operation

Asynchronous operation is available from the standard TTY interface for the FarSync adapters of types T4E+, T4Ue+Async, T4Ee and Flex. Dynamic switching between Async and Sync operation is supported.

Note: A Raw Sockets API is also available on Linux to support the adapters, though this interface is deprecated.

General Purpose Bitstream Encoding and Decoding Library

A library is provided that allows low level encoding and decoding of the transparent bitstream for speedy application development of your Linux application. The library can be configured to identify and extract most fixed-length byte-oriented user frames; where frames can be defined in terms of single or alternate synchronizing patterns, frame length and number of sync patterns to acquire sync. It provides, for example, low level frame handling for E1, DAB-ETI or DAB-STI.

The library is available on request for Windows for compatibility with legacy applications or for use with DAB STI mode. For new applications we recommend using the low level encoding and decoding built into the Windows driver.

Sample send and receive programs demonstrating common modes of operation are supplied with sample data files.

DAB modes supported

ETI

ETI(NI, G.703) - on the FarSync adapters of type TE1, TE1R and TE1e

ETI(NI, V.11) - on the FarSync adapters of type Flex, T1U, T2U, T2Ue, T2Ee, T4U, T4Ue, T4E+, T4Ee, & T2U-PMC.

STI

STI(PI, G.703) - on the FarSync adapters of type TE1, TE1R and TE1e

STI(PI, V.11) - on the FarSync adapters of type Flex, T1U, T2U, T2Ue, T2Ee, T4U, T4Ue, T4E+, T4Ee & T2U-PMC.

The library is available in 32 and 64 bit form.

Transparent bitstream mode - further details

The FarSync adapters can be run in Transparent Mode, also called bitstream mode. Transparent Mode disables zero insertion and deletion, CRC generation and checking, abort generation, and opening/closing flag generation. Generally for data transmission the application should continuously supply data at a sufficient rate for the line speed so that there are no breaks in transmission. For data reception the application should provide empty buffers at a sufficient rate for the line speed to ensure that no received data is lost. The FarSync drivers provide considerable buffering to overcome reductions in the data supply due to application rescheduling.

The communications hardware transmits data exactly as it is loaded in the transmit FIFO, when the transmitter has no more data to send it transmits mark idle, no abort sequence is sent. Abort has no meaning in Transparent Mode as all one's is as valid as is any other data sequence. Transparent Mode is useful for transmitting and receiving raw data streams such as MPEG video and audio such as T-DMB (Digital Multimedia Broadcasting) and DAB (Digital Audio Broadcasting) ETI (Ensemble Transport Interface - ETSI EN 300 799) and STI (Service Transport Interface - ETSI EN 300 797).

Many of the less common protocols for which hardware support is not commonly available can be handled in transparent mode with software handling the encoding and decoding of the particular frame format.

Transparent mode also disables the receive byte counter; therefore, short frame and long frame errors are not reported.

Receive data may not be byte aligned, as the receiver knows nothing of the data format or any synchronising sequences. If receive data needs to be byte aligned then this can be achieved in software on the Host PC.

The FarSync adapters normally send and receive the LSB of a byte first, this is however configurable to MSB if required. The receiver will begin receiving as soon as the port is started. It will continue receiving as long as there are buffers available in the receive buffer ring. Each receive block is as big as the buffer allocated for it. If the receive data is marking, then the receive buffers just keep being filled with 0xFF's until there are no buffers available or the port is stopped.

HDLC mode - further details

HDLC frames use flags to determine the beginning and end of a frame. These flags - 01111110 (or 0x7E) provide frame synchronisation. One flag may be used as both an end flag for one frame and the start flag for the next frame. Although FarSync adapters do not transmit such shared flags, they can receive and correctly handle them.

HDLC frames typically consist of an opening flag, followed by an address field, a control field, an information field (the data payload), a cyclic redundancy check (CRC) field, and, finally, a closing flag.

As far as the FarSync driver is concerned the address, control and information fields are just data.

Frames maintain data transparency by a process called zero-insertion and deletion. To ensure transparency, a 0 bit is inserted by the transmitter after 5 continuous 1s within the frame contents, and removed by the receiver whenever it detects 5

Flag	Address Field	Control field	Information field	Frame Check Sequence	Flag

continuous 1s followed by a 0. This prevents data within the frame being confused with the flags, and is known as bit-stuffing, unstuffing and also ZBID.

Normal frames are terminated with a flag - any frame which terminates with 7 1s or more is assumed to be aborted, and discarded. The transmitter can deliberately abort a frame if it chooses - this can sometimes happen if frames need to be retransmitted, and the transmitter knows that the frame currently being sent will be discarded by the receiver.

Linux Line Monitoring

The Farmon utility allows all data sent and received on the line to be recorded, very handy for debugging application behaviour.

Windows Line Monitor

The high performance multi-channel line monitor that allows the user to record, display, store and replay line traffic with protocol decoding. Comments can even be inserted into the line trace to assist later analysis.
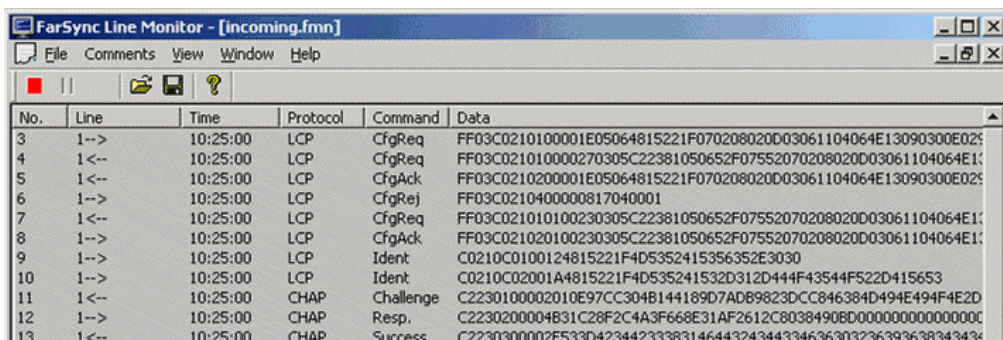
The FarSync SDK supports APIs to all the products listed in this table				
Products supported by the SDK	**Adapter Type**	**Product Codes**	**Windows API**	**Linux API**
FarSync Flex	USB	FS4100	Yes	Yes
FarSync T1U	PCI	FS4140	Yes	Yes
FarSync T2U	PCI	FS4240	Yes	Yes
FarSync T4U	PCI	FS4440	Yes	Yes
FarSync T4E+	PCI	FS4444 / FS4446	Yes	Yes
FarSync TE1	PCI	FS4150	Yes	Yes
FarSync TE1e	PCIe	FS4155	Yes	Yes
FarSync T2Ue	PCIe	FS4250	Yes	Yes
FarSync T2Ee	PCIe	FS4256	Yes	Yes
FarSync T4Ue	PCIe	FS4450 / FS4451	Yes	Yes
FarSync T4Ee	PCIe	FS4456	Yes	Yes
FarSync T2U-PMC	PMC	FS4280 / FS4281	Yes	Yes

FarSync SDK—Developers Toolkit, what's included	
Reference manuals	Windows FsWinAPI User Guide manuals (set of 4), Linux Char I/O API user manuals, Linux LAPB API manual, RS485 application guidelines (for the FarSync Flex, FarSync T2Ee, FarSync T4E+, FarSync T4Ee), Bitstream encoder/decoder User Manual, FarSync Flex Monitor User Manual .
Sample programs	Sample applications supplied that use FsWinAPI, Char I/O APIs and the bitstream encoder/decoder. There are also adapter performance check and adapter function test applications.
Source code	Sample C .NET (C# and VB) and LabVIEW VI samples.
Bitstream encoding and decoding	Library and sample programs supporting decoding of bitstream data, can be configured to support many different fixed length bitstreams such as ETI, STI and E1 framing.
Customer support	We provide free email and telephone assistance to the application developer using the API as part of the package provided when the FarSync SDK is purchased.

Order Information		
Name	**Description**	**Product Code**
FarSync SDK	Linux and Windows Developers Toolkit for the FarSync adapters - this is required if you want to write software to use the FarSync adapter APIs. *Note: This toolkit is not appropriate for the FarSync X25 range of products or the FarLinX Gateways, they are supplied with their own toolkit.*	FS9610

FarSync ® SDK Version 1.40

Tel: +49 (8241) 91 83 501
Fax: +49 (8241) 96 00 263
Email: farsite@B-NetCons.de
Web: www.B-NetCons.de

B-NetCons GmbH
Business - NetzwerkConsulting

FarSite
COMMUNICATIONS