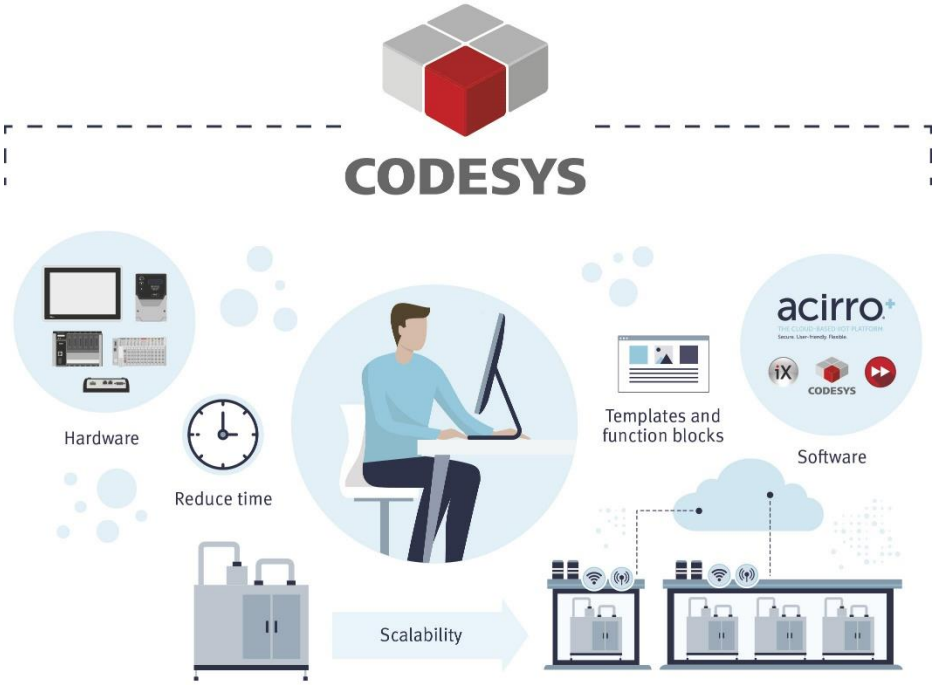
Quick start guide

Serial comms FBs - CODESYS library

SER0001 - Quick start Serial communication



Beijer ELECTRONICS

1 Function and area of use

This document explains the CODESYS library for serial communication.
Target device X2 / BoX2 control series, with embedded CODESYS runtime.

2 About this document

This quick start document should not be considered as a complete manual. It is an aid to be able to startup a normal application quickly and easily.

Use the following hardware, software, drivers and utilities in order to obtain a stable application:

In this document we have used following software and hardware

- CODESYS 3.5 SP13 patch 3
- X2 control and BoX2 control

For further information refer to

- CODESYS online help
- Installation manual X2 control (MAxx202)
- Beijer Electronics knowledge database, HelpOnline

This document and other quick start documents can be obtained from our homepage.
Please use the address *support.europe@beijerelectronics.com* for feedback.

3 Table of Contents

4 Serial communications with CODESYS function blocks

This library is compatible with X2Control and BoX2Control devices (DeviceId 0x1024)

This library simplifies serial communications from the X2 Control to serial devices such as barcode readers, weigh scales, and printers.

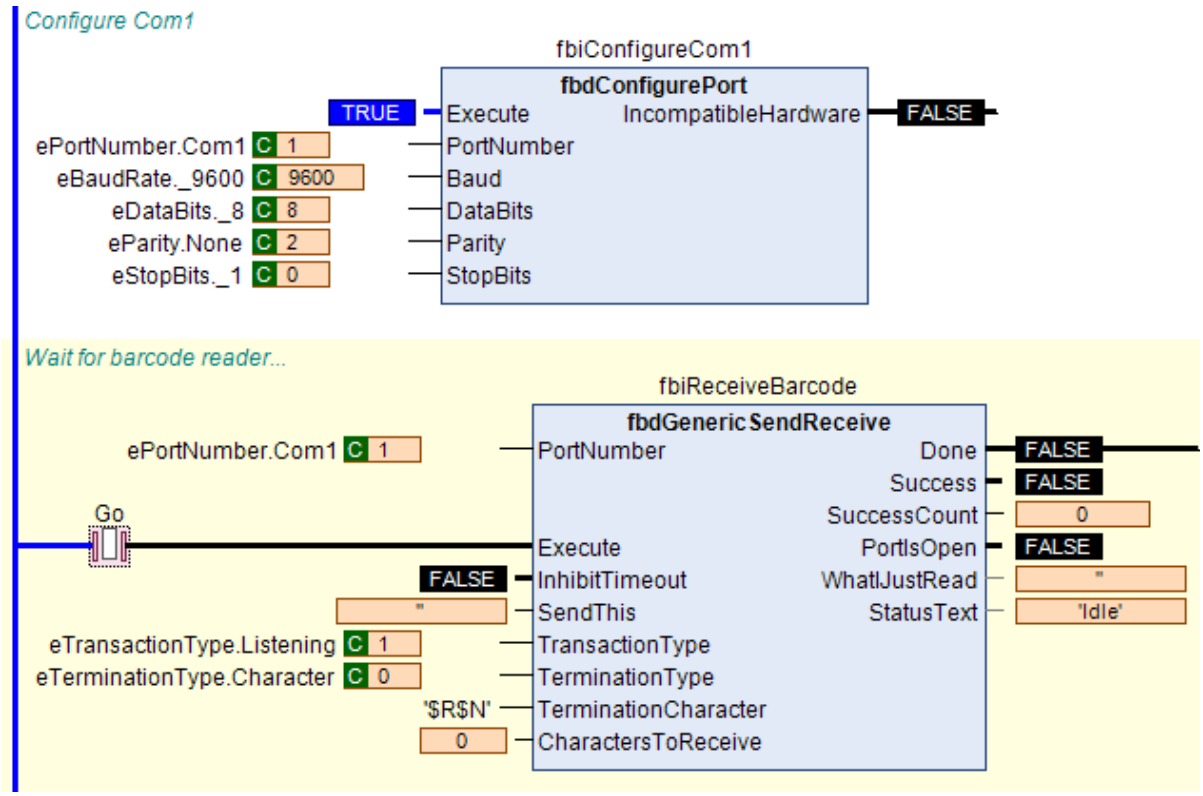
Most parameters are set with ENUMs to reduce errors.

The FB can act as a manager for send/receive (for devices that need a prompt) or just cause the port to listen (for unsolicited messages).

Message termination can be by termination character or receiving a pre-defined number of characters.

All three serial ports of an X2 / BoX2 control can be used (COM1, COM2 and COM3).

The library file (*.compiled-library) can be installed to the CODESYS software on your PC and the FBs be accessed as any blocks, please follow guidelines and description.



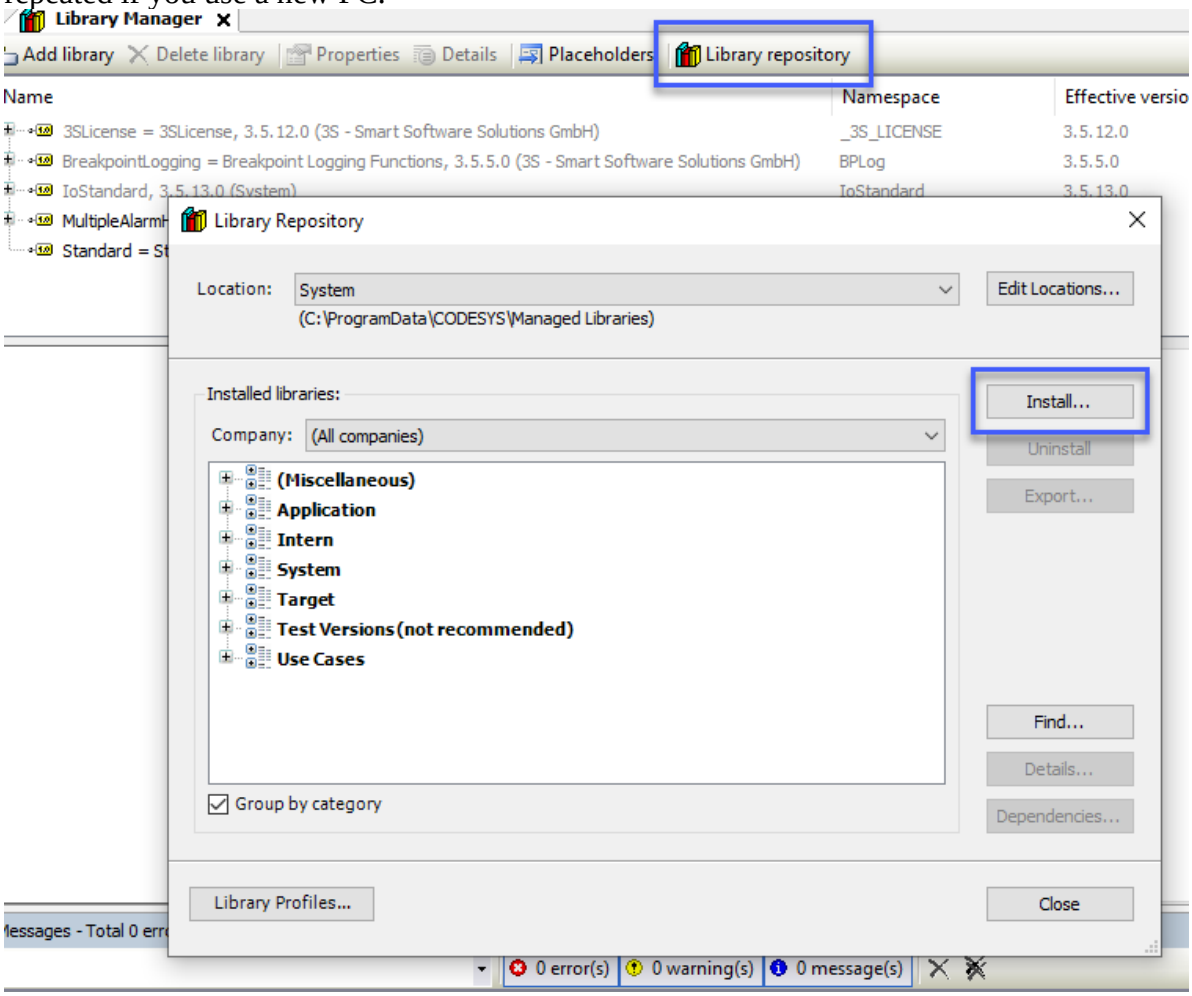
5 Preparing your editor

The following chapter describes important procedures and settings needed for a well functioning system.

5.1 Installation of the library to your editor

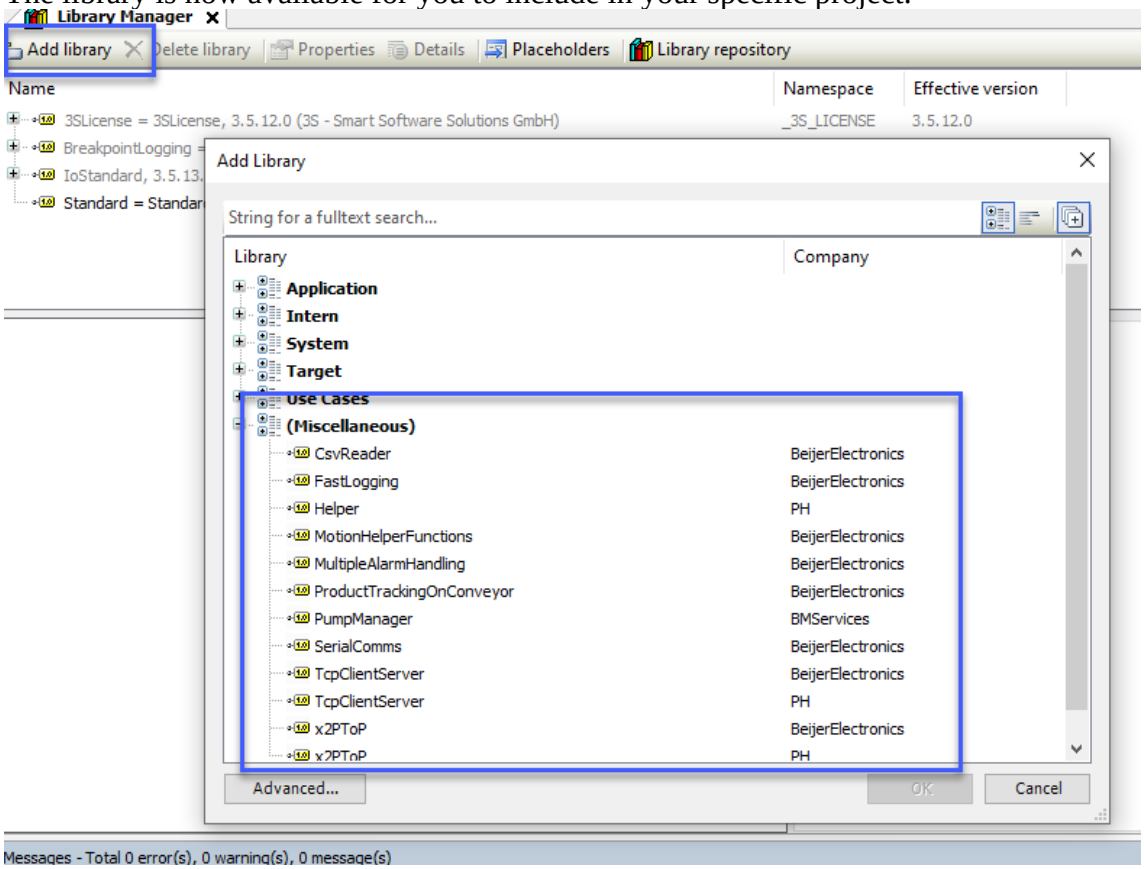
The *.compiled-library needs to be made available in your system so it can be included in projects. This is done by accessing the 'Library Manager'→'Library Repository' then 'Install'

Navigate to the folder where you have put the *.compiled-library. This procedure will need to be repeated if you use a new PC.



5.2 Add the library into your project

The library is now available for you to include in your specific project.



The selected library is now visible in the Library Manager. Its public objects and supplementary help is available here.

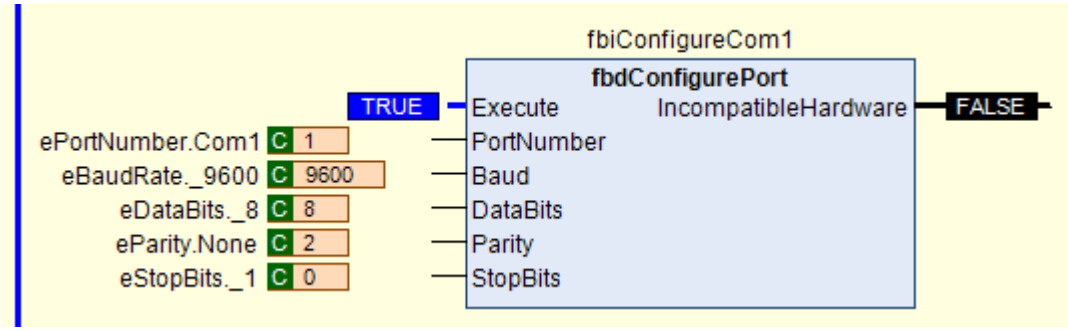
6 Description of function blocks

6.1 fbdConfigurePort

The FB **fbdConfigurePort** is required to set the port's parameters.

Match the port settings to the device you're talking to. Simply invoke and enter the appropriate port, baud, data bits, parity and stop bits.

All parameters are ENUMs.



Name	Scope	Type	Comment
Execute	VAR_IN	BOOL	Configures the port's parameters on the rising edge
PortNumber	VAR_IN	ePortNumber	Choose the serial port
Baud	VAR_IN	eBaudRate	
DataBits	VAR_IN	eDatabits	
Parity	VAR_IN	eParity	
StopBits	VAR_IN	eStopBits	
IncompatibleHardware	VAR_OUT	BOOL	Target is not an X2Control or BoX2Control device

6.2 fbdGenericSendReceive

This FB provides functionality to interact via a com port to a device.

The type can be 'Polled' or 'Listening'. Polled is used to send a request to a device and wait for a response (typically a weighscale). Listening simply waits for an unsolicited incoming message (typically a barcode reader).

The incoming message can be terminated one of two ways:

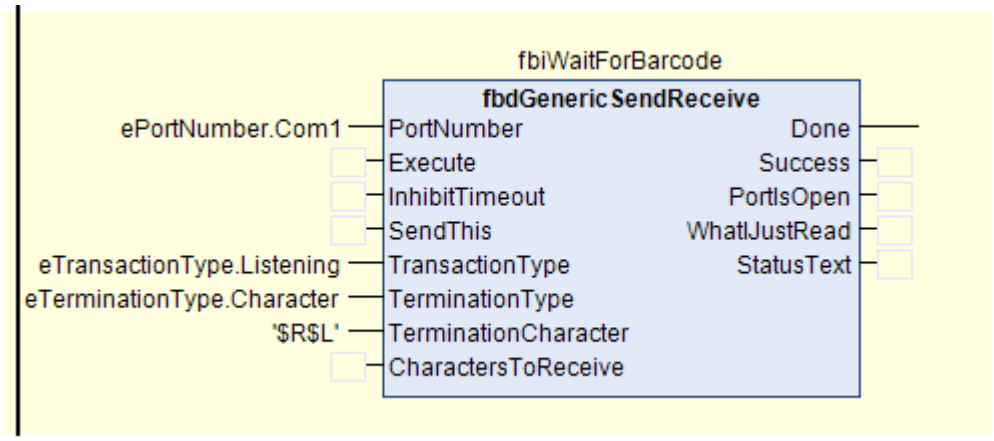
- Receiving a termination character (for example CRLF)
- After receiving a pre-defined number of characters.

BothTransactionTypes can be used with either TerminationTypes.

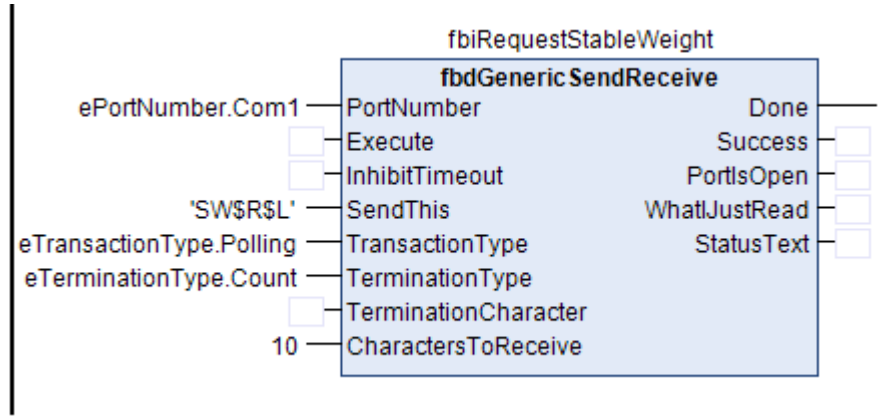
It won't execute until that port's parameters have been set.

Examples:

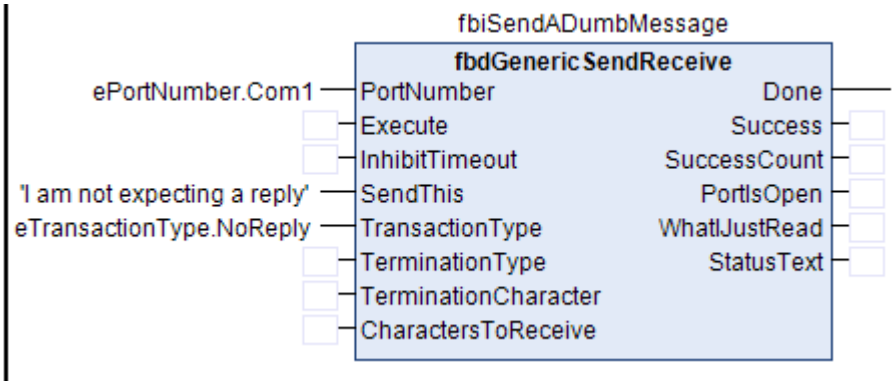
a) This configuration will wait (as long as Execute is high) for an un-prompted frame which is terminated with special characters: <CR><LF>



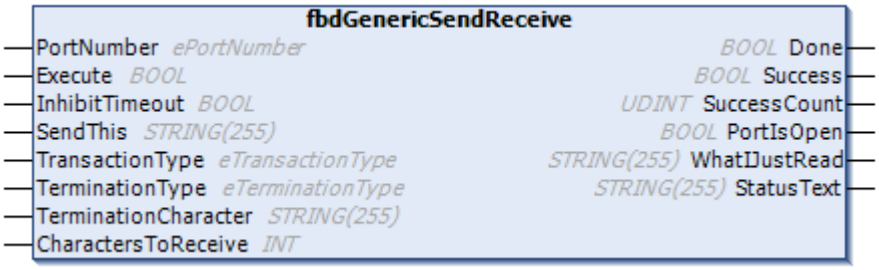
b) This configuration will (on the Execute rising edge) make a request and wait for a response, which is always padded to 10 characters.



c) This configuration will (on the Execute rising edge) send a message and not wait for a reply.



6.2.1 fbdGenericSendReceive (datatypes)



Input	Type	Initial	Comment
PortNumber	ePortNumber		Choose the serial port
Execute	BOOL		If transactionType is 'Polling' this rising edge initiates the send/receive. In 'Listening' mode, the port listens for as long as the flag is high
InhibitTimeout	BOOL		For debugging comms only. Normally FALSE
SendThis	STRING(255)		In 'Polling' mode, this is the request sent to the device
TransactionType	eTransactionType	eTransactionType.Polling	Used to choose the transaction type.
TerminationType	eTerminationType	eTerminationType.Character	Used to choose the termination type
TerminationCharacter	STRING(255)	'RN'	Valid if TerminationType is eTerminationType.Character
CharactersToReceive	INT		Valid if TerminationType is eTerminationType.Count

Output	Type	Initial	Comment
Done	BOOL		Indicates completion
Success	BOOL		Indicate a successful completion i.e. the termination character has been received
SuccessCount	UDINT		
PortIsOpen	BOOL		
WhatIJustRead	STRING(255)		The received string available for subsequent processing
StatusText	STRING(255)		See below for possibilities

Status Text	Meaning
Idle	Waiting for instruction
Opening port	Opening the port. This would indicate the port is already in use by another application
Clearing buffer	Removing old characters from the buffer
Sending	Sending the 'SendThis' string
Looking for termination character	When the TerminationType is 'Character'
Waiting for 10 characters	When the TransactionType is 'Count'
Done, request high	For TransactionTypes 'Polling' or 'NoReply' this indicates the sequence is finished and waiting a new rising edge
Invalid parameters	In TerminationMode 'Character', no termination character has been specified. In TerminationMode 'Count', the count is 0 or greater than 255

7 Special characters

Codesys identifies special characters (non-printable) with escape sequences.

This is a snippet from the Codesys Help Online.

Hexadecimal code

String with $ code	Interpretation
`'$<8-bit code>'`	8-bit code: Two-digit hexadecimal number that is interpreted according to ISO/IEC 8859-1.
`'$41'`	`A`
`'$A9'`	`©`
`'$40'`	`@`
`'$0D'`	Control character: Line break (corresponds to '$R')
`'$0A'`	Control character: New line (corresponds to '$L' and '$N')

Special cases

String with $ code	Interpretation
`'$L'`, `'$l'`	Control character: Line feed (corresponds to `'$0A'`)
`'$N'`, `'$n'`	Control character: New line (corresponds to `'$0A'`)
`'$P'`, `'$p'`	Control character: Form feed
`'$R'`, `'$r'`	Control character: Line break (corresponds to `'$0D'`)
`'$T'`, `'$t'`	Control character: Tab
`'$$'`	Dollar sign: `§`
`'$''`	Single straight quotation mark: `'`

8 About Beijer Electronics

Beijer Electronics is a multinational, cross-industry innovator that connects people and technologies to optimize processes for business-critical applications. Our offer includes operator communication, automation solutions, digitalization, display solutions and support. As experts in user-friendly software, hardware

and services for the Industrial Internet of Things, we empower you to meet your challenges through leading-edge solutions.

Beijer Electronics is a Beijer Group company. Beijer Group has a sale over 1.6 billion SEK in 2019 and is listed on the NASDAQ OMX Nordic Stockholm Small Cap list under the ticker BELE. **www.beijergroup.com**

China	**NORWAY**	**TAIWAN**
Shanghai	Drammen	Taipei
DENMARK	**SOUTH KOREA**	**TURKEY**
Roskilde	Seoul	Istanbul
FRANCE	**SWEDEN**	**UNITED KINGDOM**
Paris	Göteborg Malmö Stockholm	Nottingham
GERMANY		**USA**
Nürtingen		Salt Lake City

8.1 Contact us

Global offices and distributors